

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В. ЛОМОНОСОВА

Факультет вычислительной математики и кибернетики

**СБОРНИК ТЕЗИСОВ
ЛУЧШИХ ВЫПУСКНЫХ
КВАЛИФИКАЦИОННЫХ РАБОТ
ФАКУЛЬТЕТА ВМК МГУ
2025 ГОДА**

Москва
2025

Исследование экономического поведения домашних хозяйств на основе модели рамсеевского типа

Работа удостоена диплома 2 степени

Анашкина Анастасия Алексеевна

Кафедра системного анализа

e-mail: 1303nastia@gmail.com

Научный руководитель — акад. РАН, д.ф.-м.н. проф. Шананин Александр Алексеевич

Проблемы неравенства в распределении богатства, доходов и расходов в последние десятилетия приобретают такую же актуальность, как и сто лет назад. Тома Пикетти утверждает, что такие явления, как социальные лифты, сокращение неравенства в доходах и расходах являются особенностью XX века и в современных условиях неравенство будет определяться возрастающим неравенством в распределении богатства, характерным для XVIII и XIX веков. Филипп Агийона и Джеффри Уильямсон, так же обсуждают влияние возрастания неравенства на замедление темпов экономического роста мировой экономики. Борисов и Пахнин рассматривают модели экономического роста с неоднородным дисконтированием и проводят эмпирические исследования, которые указывают на зависимость ставки дисконтирования от неравенства в распределении доходов.

Начало формирования теории экономического роста относят к 1920 годам, когда английский математик Рамсей опубликовал работу [1], в которой выдвинул гипотезу о социальном разделении общества агентов на два класса: агентов с наименьшей ставкой дисконтирования, которые увеличивают свои сбережения и в долгосрочной перспективе владеют всем капиталом, и агентов с высокой ставкой дисконтирования, которые в пределе живут на уровне заработной платы.

На данный момент существует множество работ, посвящённых анализу гипотезы Рамсея в различных экономических моделях. Например, в статье А. А. Шананина и Г. С. Парастаева [2] рассматривается модель совершенного рынка, причём доказано, что гипотеза Рамсея выполняется.

Целью представленной работы является анализ гипотезы Рамсея в условиях несовершенного рынка, когда ставка по потребительскому кредиту превышает ставку по депозиту:

$$r_L > r_D > 0.$$

Моделирование поведения экономических агентов. Будем рассматривать модель экономического поведения репрезентативного рационального домашнего хозяйства в форме задачи оптимального управления на конечном временном горизонте $t \in [0, T]$. Из рациональности поведения агента следует, что он не осуществляет займы по потребительскому кредиту и не сберегает в форме депозитов одновременно. Кроме того, предполагаем, что агент не может предсказать изменение экономической конъюнктуры и принимает ре-

шения, основываясь на том, что его зарплата w и ставки по кредиту r_L и депозиту r_D остаются неизменными.

В каждый момент времени t финансовое состояние агента описывается его капиталом $k(t)$. Полагаем, что капитал изменяется за счёт поступления заработной платы, взаимодействия с рынками потребительского кредитования и сбережений в форме депозитов, а также расходов на потребление $c(t) \geq 0$. Таким образом, динамика капитала агента описывается дифференциальным уравнением

$$\dot{k}(t) = r_D(k)_+ - r_L(-k)_+ + w - c(t)$$

с начальным условием

$$k(0) = k_0,$$

где $(a)_+ = \max\{a, 0\}$.

Агент стремится максимизировать дисконтированное потребление с коэффициентом дисконтирования $\rho > 0$ и постоянным отвращением к риску $1 - \beta$, где $\beta \in (0, 1)$, управляя потреблением $c(t)$:

$$\int_0^T e^{-\rho s} (c(s))^\beta ds \rightarrow \max_{c(\cdot)}.$$

Также важно учесть, что в конце концов агент должен расплатиться со своими кредитами, то есть

$$k(T) > 0.$$

Таким образом, задача оптимального управления на конечном временном горизонте формулируется в следующем виде

$$\int_0^T e^{-\rho s} (c(s))^\beta ds \rightarrow \max_{c(\cdot)}. \quad (1)$$

$$\dot{k}(t) = r_D(k)_+ - r_L(-k)_+ + w - c(t), \quad k(0) = k_0, \quad t \in [0, T], \quad (2)$$

$$c(t) \geq 0, \quad t \in [0, T]. \quad (3)$$

$$k(T) \geq 0. \quad (4)$$

Теорема 1. Пусть $w + r_L k_0 > 0$, $T > \left(\frac{1}{r_L} \ln \left(\frac{w}{w + k_0 r_L} \right) \right)_+$. Тогда задача оптимального управления (1), (2), (3), (4) имеет решение.

Замечание. Чтобы обеспечить выполнение условия $k(T) > 0$, необходимо выполнение ограничения

$$k > -\frac{w}{r_L}.$$

Отметим, что данное неравенство не зависит от времени. При $T \rightarrow +\infty$ это условие является достаточным. В случае, если неравенство нарушается, то домашнее хозяйство не может расплатиться с кредитами при сохранении текущей экономической конъюнктуры, и образуется финансовая пирамида.

Синтез оптимального управления. Обратим внимание, что правая часть дифференциального уравнения (4) негладкая. В таких случаях применяется принцип максимума Понтрягина в форме Ф. Кларка [3].

Теорема 2. Если $\{c(t), k(t) | t \in [0, T]\}$ — решение задачи оптимального управления (1), (2), (3), (4), то существует абсолютно непрерывная функция $\{\psi(t) | t \in [0, T]\}$, такая что

$$c(t) = \left(\frac{\psi(t)}{\beta} \right)^{\frac{1}{\beta-1}}.$$

1. Если $k(t) > 0$, то

$$\begin{aligned} \frac{dk}{dt} &= r_D k(t) + w - \left(\frac{\psi(t)}{\beta} \right)^{\frac{1}{\beta-1}}, \\ \frac{d\psi}{dt} &= \psi(\rho - r_D). \end{aligned}$$

2. Если $k(t) < 0$, то

$$\begin{aligned} \frac{dk}{dt} &= r_L k(t) + w - \left(\frac{\psi(t)}{\beta} \right)^{\frac{1}{\beta-1}}, \\ \frac{d\psi}{dt} &= \psi(\rho - r_L). \end{aligned}$$

Кроме того, выполняется условие трансверсальности

$$k(T) = 0.$$

Для того, чтобы проанализировать гипотезу Рамсея, необходимо найти решение задачи оптимального управления (1), (2), (3), (4) в форме синтеза. Это можно сделать, устремив T к бесконечности. Таким образом, будем рассматривать задачу оптимального управления на бесконечном временном горизонте $t \in [0, +\infty)$:

$$\int_0^{+\infty} e^{-\rho s} (c(s))^\beta ds \rightarrow \max_{c(\cdot)}, \quad (5)$$

$$\dot{k}(t) = r_D (k)_+ - r_L (-k)_+ + w - c(t), \quad k(0) = k_0, \quad (6)$$

$$c(t) \geq 0, \quad (7)$$

$$k(t) \geq -\frac{w}{r_L}. \quad (8)$$

Теорема 3. Решение задачи оптимального управления (5), (6), (7), (8) в форме синтеза определяется следующими соотношениями:

1. Если $\rho > r_L$, то

$$c(k, \rho) = \begin{cases} \frac{\frac{\rho}{r_L} - \beta}{1 - \beta} (r_L k + w), & k \leq 0 \\ w \hat{x}(k, \rho), & k > 0, \end{cases}$$

где $\hat{x}(k, \rho)$ — решение уравнения

$$-\frac{k}{w} - \frac{1}{r_D} + \frac{1 - \beta}{\rho - \beta r_D} x + x^{-\frac{r_D(1-\beta)}{\rho - r_D}} \left(\frac{\frac{\rho}{r_L} - \beta}{1 - \beta} \right)^{\frac{r_D(1-\beta)}{\rho - r_D}} \frac{1}{r_D} \left(1 - \frac{\frac{\rho}{r_L} - \beta}{\frac{\rho}{r_D} - \beta} \right) = 0. \quad (9)$$

2. Если $\rho < r_D$, то

$$c(k, \rho) = \begin{cases} w \hat{x}(k, \rho), & k < 0, \\ \frac{\frac{\rho}{r_D} - \beta}{1 - \beta} (r_D k + w), & k \geq 0 \end{cases}$$

где $\hat{x}(k, \rho)$ — решение уравнения

$$-\frac{k}{w} - \frac{1}{r_L} + \frac{1 - \beta}{\rho - \beta r_L} x + x^{\frac{r_L(1-\beta)}{r_L - \rho}} \left(\frac{\frac{\rho}{r_D} - \beta}{1 - \beta} \right)^{-\frac{r_L(1-\beta)}{r_L - \rho}} \frac{1}{r_L} \left(1 - \frac{\frac{\rho}{r_D} - \beta}{\frac{\rho}{r_L} - \beta} \right) = 0. \quad (10)$$

3. Если $r_D \leq \rho \leq r_L$, то

$$c(k, \rho) = \begin{cases} w \hat{x}_1(k, \rho), & k < 0, \\ w, & k = 0, \\ w \hat{x}_2(k, \rho), & k > 0, \end{cases}$$

где $\hat{x}_1(k, \rho), \hat{x}_2(k, \rho)$ — решения уравнений 10, 9 соответственно.

Замечание. Доказано, что каждое из уравнений (9), (10) имеет единственный корень в рассматриваемой области.

Модель социальной динамики. Гипотезу Рамсея можно рассматривать как утверждение о справедливости принципа естественного отбора по Фишеру в популяции из H домохозяйств.

Предполагаем, что в начальный момент времени $t_0 = 0$ известна величина сбережений каждого агента, причём

$$k_0^h \geq 0, \quad h \in \{1, \dots, H\}.$$

Введём обозначение общих сбережений всех домохозяйств

$$K = \sum_{h=1}^H r(k^h) k^h + wH,$$

где

$$r(k^h) = \begin{cases} r_L, & k^h \leq 0, \\ r_D, & k^h > 0. \end{cases}$$

Для анализа гипотезы Рамсея важна зависимость синтеза задачи оптимального управления не только от капитала, но и от ставки дисконтирования. Чтобы понять, как изменяется ρ , обратимся к гипотезе Дьюзенберри [4]: потребление агента зависит не только от его дохода, но и от его дохода относительно других агентов. Таким образом, получим выражение:

$$\rho^h(k^h, K) = r(k^h) \varphi \left(\frac{r(k_0^h) \cdot k_0^h + w}{K} \right).$$

Определение. Функция $\varphi(\cdot) : \mathbb{R} \rightarrow \mathbb{R}_+$ удовлетворяет следующим условиям:

- Непрерывна по Липшицу, монотонно убывает на $[0, 1]$,
- $\varphi(x) = \varphi(1)$, $x \geq 1$,
- $\varphi(x) = \varphi(0)$, $x \leq 0$,
- $\beta \leq \varphi(1) < \varphi\left(\frac{1}{H}\right) \leq 1 < \varphi(0)$,
- вогнута на $[0, 1]$.

Теперь, используя синтез, найденный в предыдущем разделе, можем описать динамику капитала каждого агента в популяции в форме задачи Коши

$$\frac{dk^h}{dt} = r(k^h) \cdot k^h + w - c(k^h, \rho^h(k^h, K)), \quad k^h(0) = k_0^h.$$

Для качественного анализа выполнимости гипотезы Рамсея на несовершенном рынке кредитов и депозитов было разработано численное решение. В качестве платформы, на которой разрабатывалась программа, была взята платформа MATLAB.

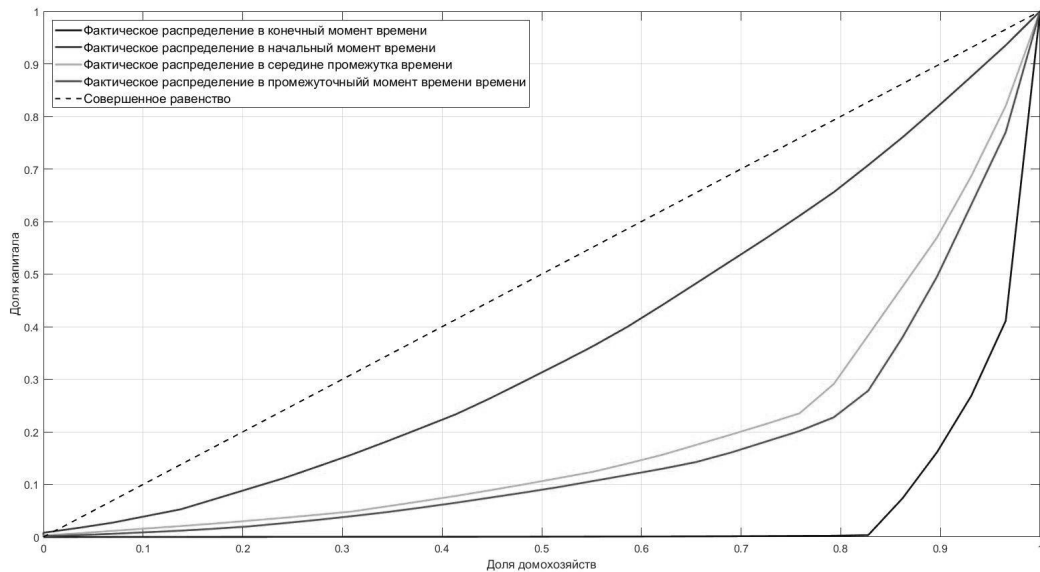


Рис. 1: График кривой Лоренца

Наиболее информативным является график кривой Лоренца [5], изображённый на рис. 1. Рассмотрим нижний график, соответствующий конечному моменту времени. Видно, что он разделяется на три сегмента, каждый из которых соответствует социальному классу:

1. Пологая часть графика означает, что большая доля населения владеет нулевым капиталом, и соответствует классу «бедных».
2. Следующая часть отклонена от оси Ox и соответствует «среднему» классу.
3. Последняя часть графика практически вертикальна, то есть малая доля населения сосредотачивает в своих руках практически весь капитал, что соответствует классу «богатых».

Также был проанализирован индекс Джини. Вычисления показали, что значение индекса увеличивается со временем, то есть уровень неравенства в рассматриваемой популяции экономических агентов растёт.

Было интересно пронаблюдать, как изменяется график кривой Лоренца при увеличении значения параметра r_L . Вычисления показали, что при увеличении r_L , пологая часть графика сокращается, то есть уменьшается класс «бедных», за счёт чего увеличивается «средний» класс. Интересно, что при $r_L \rightarrow +\infty$ в модели выделяется только два социальных класса. Этот случай соответствует модели Бэккера, исследованной в статье [2], где доказана гипотеза Рамсея о бинарной стратификации.

Таким образом, численное решение помогло установить, что в рамках рассматриваемой модели гипотеза Рамсея о бинарной стратификации общества экономических агентов не подтверждается. В результате вычисления ставки дисконтирования, основанном на гипотезе Дьюзенберри, наблюдается формирование трёх устойчивых социальных классов.

ЛИТЕРАТУРА

- [1] Ramsey F. P. A mathematical theory of savings // The Economic Journal. 1928. Vol.152, N 38, P.543–559.
- [2] Parastaev G. S., Shananin A. A. Ramsey's conjecture of social stratification as Fisher's selection principle // Computational Mathematics and Mathematical Physics, 2024. Vol.64, N 12, P.2952–2981.
- [3] Clarke F. H., Ledyaev Yu. S., Stern R. J., Wolenski P. R. Nonsmooth analysis and control theory // N. Y., Springer–Verlag. 1998. P.276.
- [4] Duesenberry J. S. Income, saving and the theory of consumer behavior // Harvard University Press. 1949.
- [5] Lorenz M. O. Methods of measuring the concentration of wealth // Publications of the American Statistical Association. 1905. P.209–219.

Разработка интерпретатора частично рекурсивных функций

*Работа удостоена диплома 2 степени
Астраханцев Дмитрий Андреевич*

*Кафедра автоматизации систем вычислительных комплексов
e-mail: s02210002@gse.cs.msu.ru*

*Научный руководитель — к.ф.-м.н. доц. Волканов Дмитрий Юрьевич
Научный консультант — программист Курячий Георгий Владимирович*

Обучение курсу «Алгоритмы и алгоритмические языки» на факультете ВМК традиционно начинается с изучения машины Тьюринга (МТ) и нормального алгоритма Маркова (НАМ), однако данные модели иллюстрируют в большей степени императивный подход к программированию и не раскрывают в должной мере широту понятия алгоритма и Тьюринг полноты. Однако класс частично рекурсивных функций при кажущейся простоте определения эквивалентен классу вычислимых по Тьюрингу функций, что позволяет показать функциональный подход к программированию и расширить в глазах студента понятие полноты по Тьюрингу, не вдаваясь в изучение какого-либо серьезного языка программирования. При наличии простого инструментария, который бы иллюстрировал основные принципы работы с частично рекурсивными функциями, данный курс можно было бы дополнить их изучением [1].

Проведен обзор языков (Haskell, Lisp, Рефал) и формальных моделей (МТ, НАМ) с позиции их пригодности для краткого введения студентов в функциональное программирование:

1. Haskell — мощный язык, требующих глубоких знаний абстракций, не подходит для быстрого введения в функциональное программирование.
2. Lisp — проще, но перегружен синтаксически, мультипарадигмален и имеет схожие проблемы с Haskell.
3. Рефал — иллюстрирует механизм сопоставления шаблону, но не показывает чисто функциональный подход.
4. МТ/НАМ — простые модели, подходящие для обучения, но императивные.
5. Язык на основе ЧРФ — перспективен, но утратил поддержку и доступен только в качестве бинарного файла для Windows XP.

Поскольку ни одна из существующих альтернатив не является приемлемой, целью данной работы является разработка собственного интерпретатора и сопутствующего инструментария для наглядного знакомства с основами функционального программирования в рамках 1-2 лекций. При создании данного интерпретатора и инструментария к нему требуется:

- Разработать язык описания частично рекурсивных функций.
- Реализовать интерпретатор данного языка.

- Разработать и реализовать графический интерфейс для работы с описываемым программным обеспечением.

В ходе работы создан новый декларативный язык, основанный только на частично рекурсивных функциях: нулевая функция, функция следования, функции выбора. Над ними определены операции суперпозиции, примитивной рекурсии и минимизации [2, 3]. Для формализации синтаксиса разработана грамматика в расширенной форме Бэкуса—Наура, пригодная для анализа LALR(1)-парсером.

Примеры:

- o, s, I^n_m — базовые функции.
- $F = F\{A, B\}$ — суперпозиция.
- $F = A \leftarrow B$ — примитивная рекурсия.
- $F = ?A$ — минимизация.

Интерпретатор данного языка реализован на языке Python с использованием библиотеки Lark. Интерпретация кода происходит в несколько этапов:

1. Распознавание и анализ исходного кода по грамматике, построение абстрактного синтаксического дерева.
2. Построение Python-объектов, реализующих описанные кодом функции средствами языка.
3. Оптимизация Python-объектов, построенных ранее.
4. Вычисление значений функций.

Для улучшения производительности интерпретатора были рассмотрены несколько стратегий оптимизации:

- Преобразование рекурсии в цикл — замена рекурсии на цикл, уменьшающая нагрузку на поддержание стека.
- Мемоизация (кэширование) — сохранение результатов вычисленных функций в памяти. Данная стратегия не была применена из-за крайне высоких затрат на хранение кэшированных данных.
- Промежуточное представление в коде языка Haskell — дало небольшой прирост, но не показало высокой эффективности. Решение не было внедрено из-за конфликтов с другими оптимизациями.
- Оппортунистическая оптимизация — распознавание и замена базовых операций (сложение, разность, умножение и деление) на эффективные операции. Дала высокий прирост производительности.

Графический интерфейс реализован с использованием библиотеки PySide6 и предоставляет следующие функции:

- Средства работы с файлами (открытие, создание, сохранение и пр.).
- Средства работы с текстом (выделение, копирование, редактирование и пр.).

- Подсветка синтаксиса.
- Наличие светлой и темной цветовых тем.
- Отображение номеров строк.
- Возможность запуска в режиме вывода отладочной информации.

Таким образом, в рамках работы был разработан инструментарий для наглядного знакомства с основами функционального программирования на основе формальной модели частично рекурсивных функций. В ходе разработки данного инструментария были решены следующие подзадачи:

- Проведен обзор, по результатам которого было решено разработать язык на основе частично рекурсивных функций и его интерпретатор.
- Разработан синтаксис и грамматика языка.
- Реализован интерпретатор данного языка.
- Реализован графический пользовательский интерфейс с возможностью отладки кода.

ЛИТЕРАТУРА

- [1] Корухова Л. С., Шура-Бура М. Р. Введение в алгоритмы. Учебное пособие для студентов I курса. 2-е исправленное издание. М. : Издательский отдел факультета ВМиК МГУ; МАКС Пресс, 2010. 26 с.
- [2] Мальцев А. И. Алгоритмы и рекурсивные функции. 2-е изд. М. : Наука. Гл. ред. физ.-мат. лит., 1986. 368 с.
- [3] Петер Р. Рекурсивные функции // Под ред. и с пред. Колмогорова А. Н. М. : Издательство иностранной литературы, 1954. 264 с.

Применение метода квантовой нормализации для решения задач из теории целочисленных решёток

Аталаева Марьям Гитиномагомедовна

Кафедра информационной безопасности

e-mail: maryam.atalaeva@mail.ru

Научный руководитель — к.ф.-м.н. доц. Применко Эдуард Андреевич

В эпоху развития квантовых вычислений криптография на основе целочисленных решёток стала одним из ведущих кандидатов для постквантовой стандартизации. Безопасность таких систем базируется на предполагаемой вычислительной сложности задач о кратчайшем (SVP) и ближайшем (CVP) векторах в решётке. В то время как эти задачи считаются трудными для классических компьютеров, их устойчивость к атакам с использованием квантовых алгоритмов остаётся ключевым вопросом исследований. Среди различных моделей квантовых вычислений особый интерес представляют квантовые адиабатические вычисления (AQCS) [1, 3, 4] и их эвристический аналог — квантовый отжиг (QA) [5]. Эти подходы требуют предварительного преобразования исходной задачи в формат, понятный квантовому устройству, — как правило, в задачу квадратичной неограниченной бинарной оптимизации (QUBO) или эквивалентную ей модель Изинга [2]. В данной работе этот процесс преобразования называется «квантовой нормализацией».

Проблема и предлагаемое решение. Прямой подход к «квантовой нормализации» задачи CVP заключается в решении следующей оптимизационной задачи:

$$\|Bx - t\|^2 \rightarrow \min \quad \text{при условии } x \in \mathbb{Z}^n$$

где B — базис решётки, t — целевой вектор, а $x = (x_1, \dots, x_n)^T$ — искомый вектор целочисленных коэффициентов.

Для сведения этой задачи к формату QUBO необходимо избавиться от целочисленных переменных. Это достигается путём их замены на бинарные переменные, — кубиты, предварительно ограничив область поиска для каждого коэффициента диапазоном $[-M, M]$. Каждая переменная x_i представляется через k бинарных переменных $y_{ip} \in \{0, 1\}$:

$$x_i = \left(\sum_{p=0}^{k-1} y_{ip} 2^p \right) - M.$$

Однако этот метод сталкивается с фундаментальным препятствием: общее число требуемых кубитов $N_{\text{qubits}} = n \times k$ напрямую зависит от ширины диапазона M . Таким образом, необходимость увеличивать число кубитов с расширением диапазона целых чисел становится фундаментальным препятствием для масштабирования этого подхода на существующих и перспективных квантовых устройствах. Для преодоления этой проблемы в данной работе предложен и экспериментально исследован гибридный адиабатический алгоритм. Его ключевая идея состоит в том, чтобы применять ресурсоёмкий

процесс сведения к QUBO не к исходной масштабной задаче, а к значительно более простой, вспомогательной задаче.

Методология гибридного алгоритма. Предложенный метод эффективно сочетает сильные стороны классических и квантовых вычислений, разделяя решение на два этапа.

1. **Классическая предобработка и аппроксимация.** На первом этапе для приведения базиса решётки B к более удобному виду используется алгоритм редукции базиса LLL. Затем, для нахождения приближенного решения задачи CVP, применяется алгоритм Бабаи, который для исходного целевого вектора t_0 находит точку решётки $u = \text{Babai}(B', t_0)$, близкую к t_0 , но, как правило, не являющуюся оптимальной.
2. **Квантовое уточнение решения.** На втором этапе формируется новая задача CVP для «вектора ошибки» $t' = t_0 - u$. Целью становится поиск оптимального вектора-поправки z^* , такого что точка $v = u + B'z^*$ является решением исходной задачи. Именно эта новая, более простая задача для близкого вектора t' подвергается «квантовой нормализации» — сведению к задаче QUBO и ее последующему решению с помощью квантового адиабатического вычислителя. Фундаментальное преимущество подхода в том, что компоненты вектора коэффициентов z^* для малой поправки лежат в значительно более узком диапазоне. Это напрямую ведёт к резкому сокращению числа кубитов, необходимых для их кодирования.

Экспериментальное исследование и результаты. Для подтверждения эффективности гибридного алгоритма была проведена серия вычислительных экспериментов. Моделирование было реализовано на языке Python с использованием ключевых библиотек: NumPy для линейной алгебры, SciPy для аппроксимации данных и fpylll для точного решения CVP в рамках симуляции. Исследование проводилось для двух типов решёток, имеющих важное криптографическое значение: случайных целочисленных и q-ичных решёток.

Для каждой размерности решётки n генерировались тысяча экземпляров задач и вычислялось необходимое количество кубитов N_{qubits} для предложенного гибридного метода. Результаты показали, что для его реализации на квантовом адиабатическом вычислителе требуется гораздо меньшее число кубитов по сравнению с прямым квантовым методом. Для q-ичных решёток была выведена эмпирическая формула зависимости среднего числа кубитов $\tilde{N}_{\text{qubits}}(n)$ от размерности n :

$$\tilde{N}_{\text{qubits}}(n) \approx 0.2739 \cdot n \log_2(n) + 14.6366.$$

Эта формула не только подтверждает высокую эффективность метода, но и предоставляет инструмент для прогнозирования ресурсов, необходимых для решения более масштабных задач криптоанализа.

Заключение и перспективы. Данная работа демонстрирует, что предложенный гибридный классическо-квантовый подход является практически значимым и ресурсоэффективным методом для решения задачи CVP. Экс-

периментально подтверждено, что предварительная классическая обработка позволяет свести сложную задачу к более простой, решение которой становится доступным для квантовых устройств с ограниченным числом кубитов. Полученные численные оценки и эмпирическая формула вносят вклад в понимание практических аспектов применения квантового отжига к задачам криптоанализа на решётках. Дальнейшие исследования могут включать анализ влияния различных схем кодирования целых чисел и, что наиболее важно, экспериментальную проверку данного подхода на реальных квантовых отжигателях.

ЛИТЕРАТУРА

- [1] Quantum Computation by Adiabatic Evolution / E. Farhi, J. Goldstone, S. Gutmann, M. Sipser // arXiv preprint quant-ph/0001106. 2000.
- [2] Ising E. Beitrag zur Theorie des Ferromagnetismus // Zeitschrift für Physik. 1925. T. 31. С. 253–258.
- [3] Born M., Fock V. Beweis des Adiabatenatzes // Zeitschrift für Physik. 1928. T. 51. С. 165–180.
- [4] Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation / D. Aharonov, W. van Dam, J. Kempe, Z. Landau // Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science. 2004. С. 42–51.
- [5] Kadowaki T., Nishimori H. Quantum annealing in the transverse Ising model // Physical Review E. 1998. T. 58, № 5. С. 5355–5363.

Усовершенствованный поиск архитектур в графовом AutoML: расширение и динамическая приоритизация пространства поиска для повышения эффективности

Балабанов Фёдор Михайлович

Кафедра системного программирования

e-mail: fedor.balabanov@list.ru

Научный руководитель — к.ф.-м.н. асс. Морозов Сергей Вячеславович

Научный консультант — м.н.с Лукьянов Кирилл Сергеевич

В работе представлен способ улучшения качества архитектур, получаемых в задаче автоматического поиска архитектур графовых нейронных сетей. Предлагается новый подход, основанный на адаптивном изменении пространства поиска посредством выделения приоритетных направлений, что позволяет повысить эффективность поиска архитектур. Еще один предлагаемый подход расширяет пространство поиска, разрешая комбинировать различные типы графовых свёрточных слоев. Эффективность предложенных подходов показана на данных с задачей классификации графов и с задачей классификации вершин.

Проблематика. Графовые нейронные сети (ГНС) — это мощный и активно развивающийся класс моделей машинного обучения, который напрямую работает с графовыми структурами данных. Применение графов встречается в самых разных областях, включая анализ социальных сетей, моделирование взаимодействий молекул в химии, а также в задачах, связанных с анализом научных данных, например, на основе графов цитирования научных статей. Однако разработка графовых нейронных сетей является нетривиальной задачей, требующей глубокого понимания как специфики графов, так и принципов построения нейронных сетей. Специфика графовых нейронных сетей также заключается в том, что исследователями было предложено большое количество различающихся по принципу своей работы графовых свёрточных методов, из-за чего пространство поиска архитектур графовых нейронных сетей очень большое. Подбор архитектуры нейронной сети, а также оптимизация гиперпараметров являются ключевыми аспектами, которые напрямую влияют на качество модели.

В связи с этим, автоматизация процесса проектирования нейронных сетей с использованием методов автоматического поиска моделей машинного обучения AutoML (Automated Machine Learning) становится всё более актуальной. Одной из главных подзадач задачи AutoML (Automated machine learning) является задача поиска архитектур нейронных сетей (Neural Architecture Search, NAS). Основной проблемой в задаче NAS является долгое время оценки каждой нейронной сети, потому что для её оценки необходимо обучить модель и измерить метрику качества. NAS методы часто используют подходы обучения с подкреплением для поиска архитектур, что позволяет моделям «изу-

чать» пространство возможных архитектур на основе сигналов обратной связи, получаемых от качества обучаемых моделей.

Вклад работы. Целью данной работы является исследование способов ускорения процесса автоматического поиска архитектур графовых нейронных сетей при расширении пространства поиска. Предлагается новый подход выделения приоритетного направления поиска, путем приоритизации свёрточных слоёв, которые себя хорошо зарекомендовали. Исследуется влияние использования различных свёрточных методов в одной архитектуре при поиске архитектур с использованием обучения с подкреплением. Будут исследованы способы модификации пространства поиска и применения методов обучения с подкреплением для ускорения процесса NAS.

Предложенные решения. Одной из гипотез, выдвинутых перед началом работы, было предположение, что приоритизация свёрточных методов, которые хорошо себя зарекомендовали в данной задаче, позволит увеличить максимальное качество сгенерированных методов AutoML архитектур. При этом пространство поиска не уменьшается, потому что рассматриваются так же и другие свёрточные слои ГНС. Реализовать приоритизацию можно в двух вариантах: давая приоритет с фиксированным увеличением вероятности выбора и с динамическим вариантом, повышая вероятность выбора слоёв, которые лучше себя показывают в течении работы алгоритма AutoML.

Также была выдвинута гипотеза, что использование различных свёрточных методов в одной архитектуре позволит увеличить максимальное качество сгенерированных методов AutoML архитектур. Для проверки этой гипотезы сравнивались методы, которые генерировали один свёрточный метод для всей архитектуры, и методы, которые генерировали для каждого свёрточного слоя архитектуры свой метод.

Эксперименты и полученные результаты. Предложенные подходы к AutoML сравнивались на данных трёх доменов:

1. Графы с данными о цитировании научных статей с задачей классификации вершин: Cora, CiteSeer, PubMed;
2. Графы с данными о покупках с задачей классификации вершин: Photos, Computers;
3. Графы химических молекул с задачей классификации графов: MUTAG, PROTEINS, COX2, BZR, AIDS.

Метрикой для сравнения являлось максимальное качество среди сгенерированных методом AutoML архитектур.

В экспериментах лучше всего себя показал подход с динамической приоритизацией без использования комбинаций свёрточных методов в одной архитектуре. Немного хуже показал себя базовый метод без приоритизации, но использующий комбинации свёрточных методов в одной архитектуре.

По результатам экспериментов был сделан вывод, что лучше использовать подход с динамической приоритизацией без комбинаций свёрточных методов. Но если есть априорное знание о полезности комбинаций в данной задаче,

то можно использовать базовый подход без приоритизации, использующий комбинации различных свёрточных методов в одной архитектуре.

По результатам работы была написана и принята для публикации статья под названием "Усовершенствованный поиск архитектур в графовом AutoML: расширение и динамическая приоритизация пространства поиска для повышения эффективности"[1].

ЛИТЕРАТУРА

- [1] Балабанов Ф. М., Лукьянов К. С. Усовершенствованный поиск архитектур в графовом AutoML: расширение и динамическая приоритизация пространства поиска для повышения эффективности // Труды Института системного программирования РАН. 2025. Том 37, вып. 2. С. 115–128. 2025.

Оценки статической и динамической активности для функций алгебры логики от малого числа переменных

Белоусова Анна Ивановна

Кафедра математической кибернетики

e-mail: belousovaai@my.msu.ru

Научный руководитель — к.ф.-м.н. доц. Шуплецов Михаил Сергеевич

В работе изучаются оценки статической и динамической активности функций алгебры логики от двух, трёх, четырёх и пяти переменных при реализации их в классе конъюнктивно-инверсных графов (AIG). AIG — ориентированный ациклический граф, который представляет схему из функциональных элементов в базисе $\mathcal{B} = \{\&, \neg\}$ (конъюнкция, отрицание).

В общем случае статическая активность характеризует количество включённых элементов схемы, то есть таких, которые реализуют 1 на своём выходе при определённом наборе входных значений. Динамическая активность, в свою очередь, показывает, сколько элементов изменило своё состояние с 1 на 0 или с 0 на 1 при переходе от одного входного набора к другому.

Динамическая и статическая активность в некотором смысле характеризуют энергозатраты, связанные с работой схемы, и энергопотребление схемы в состоянии «покоя» соответственно.

Данные характеристики являются крайне важными для устройств с ограниченными вычислительными ресурсами. Известно, что в схемах, работающих с небольшой частотой, основные расходы энергии приходятся на состояние покоя и эти расходы фактически определяются током утечки схемы. Принципиально важно при синтезе подобных устройств выбирать такие схемы, для которых значения энергопотребления в состоянии покоя минимально.

Одним из первых результатов по изучению статической активности СФЭ является работа Касим-Заде О. М. «Об одновременной минимизации сложности и мощности схем из функциональных элементов» [1], в которой устанавливаются оценки функции Шеннона в конечном полном базисе и устанавливается, что для одних базисов функция Шеннона имеет линейный рост, а для других — экспоненциальный.

В области исследования динамического энергопотребления одной из первых стала работа Ложкина С. А. и Шуплецова М. С. [2], в которой был предложен функционал динамической активности для модели СФЭ в произвольном базисе и установлен линейный порядок роста функции Шеннона.

Определение. Статическая активность на наборе α

Для набора $\alpha \in B^n$ и графа G статической активностью графа G на наборе α называется величина:

$$E(G, \alpha) = \sum_{i=1}^{L(G)} \phi_i(\alpha),$$

где $L(G)$ — число конъюнкций в графе G , а $\phi_i(\alpha)$ — функция, реализующаяся в i -ой вершине графа G на наборе α .

Определение. Статическая активность графа G

Статической активностью графа G называется максимальное значение $E(G, \alpha)$ по всем входным наборам:

$$E(G) = \max_{\alpha \in B^n} E(G, \alpha).$$

Определение. Динамическая активность на наборах α, β

Для наборов $\alpha, \beta \in B^n$ и графа G динамической активностью на паре (α, β) называется величина:

$$S(G, \alpha, \beta) = \sum_{i=1}^{L(G)} (\phi_i(\alpha) \oplus \phi_i(\beta)),$$

где $\oplus$ — сложение по модулю 2.

Определение. Динамическая активность графа G

Динамической активностью графа G называется максимальное значение $S(G, \alpha, \beta)$ по всем парам входных наборов:

$$S(G) = \max_{\alpha, \beta \in B^n} S(G, \alpha, \beta).$$

Определение. Функции активности и функция Шеннона.

- Статическая активность ФАЛ: $E(f) = \min_{G \text{ реализует } f} E(G)$
- Соответствующая функция Шеннона: $E(n) = \max_{f \in P_2(n)} E(f)$
- Динамическая активность ФАЛ: $S(f) = \min_{G \text{ реализует } f} S(G)$
- Соответствующая функция Шеннона: $S(n) = \max_{f \in P_2(n)} S(f)$

где $P_2(n)$ — множество всех булевых функций от n переменных.

Задача.

1. Построить базу данных инверсных графов с вычисленными значениями статической и динамической активности.
2. Исследовать значения функции Шеннона для статической и динамической активности АИГ-графов с малым числом входных переменных.

Результаты. В таблице 1 содержатся характеристики для графов от двух переменных.

В таблице 2 указаны аналогичные результаты полученных данных для трех переменных.

Для функций 4 переменных предоставлена зависимость количества полученных классов от значений статической активности на рисунке 2 и динамической активности на рисунке 3.

Вектор значений	Сложность	Глубина	Статическая активность	Динамическая активность
0000	2	2	1	1
0001	1	1	1	1
0011	2	2	1	2
0110	3	2	1	2

Таблица 1: Таблица значений сложности, глубины, статической и динамической активности AIG, реализующих функции от двух переменных.

Вектор значений α	Сложность	Глубина	Статическая активность	Динамическая активность
00000000	2	2	1	1
00000001	1	1	1	1
00000011	2	2	1	2
00000110	3	3	2	3
00000111	2	2	1	2
00010110	5	4	3	5
00010111	4	3	2	4
00011110	5	3	3	4
01101001	6	4	3	6

Таблица 2: Таблица значений сложности, глубины, статической и динамической активности AIG, реализующих функции от трех переменных.

Точного значения для функций 5 переменных предложенным алгоритмом установить не удалось. Тем не менее, произведена попытка получить максимально возможное количество классов функций, которые имеют точное значение. Примерно для половины получены точные оценки, для второй половины получены только верхние. Все полученные классы с соответствующими оценками опубликованы в открытом репозитории [3].

Для получения более точной оценки функции Шеннона для функций пяти переменных, используем результаты сложности схем в базисе AIG [4].

Известно [4], что максимальная сложность схем от 5 переменных не превышает 17. Это значение является верхней оценкой для функции Шеннона для статической и динамической активности.

В таблице 3 содержатся полученные результаты значения функции Шеннона для $n = 2, 3, 4$ для статической и динамической активностей.

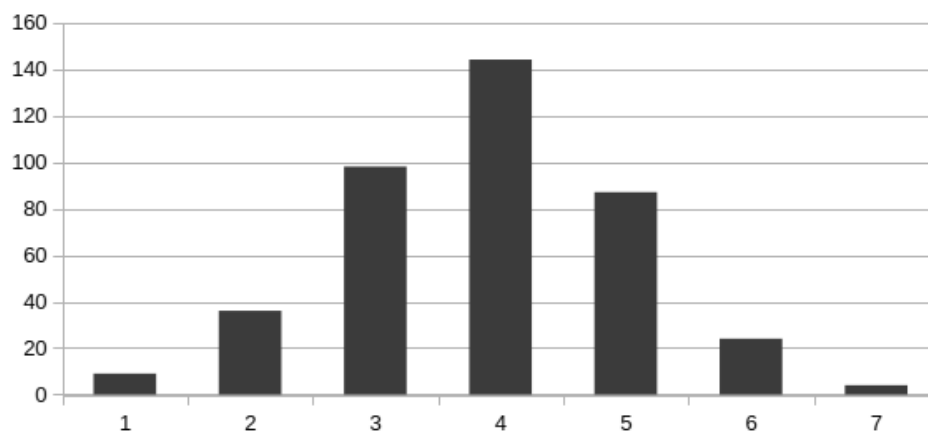


Рис. 1: Гистограмма количества инверсно-конгруэнтных классов функций от 4-х переменных с фиксированным значением статической активности.

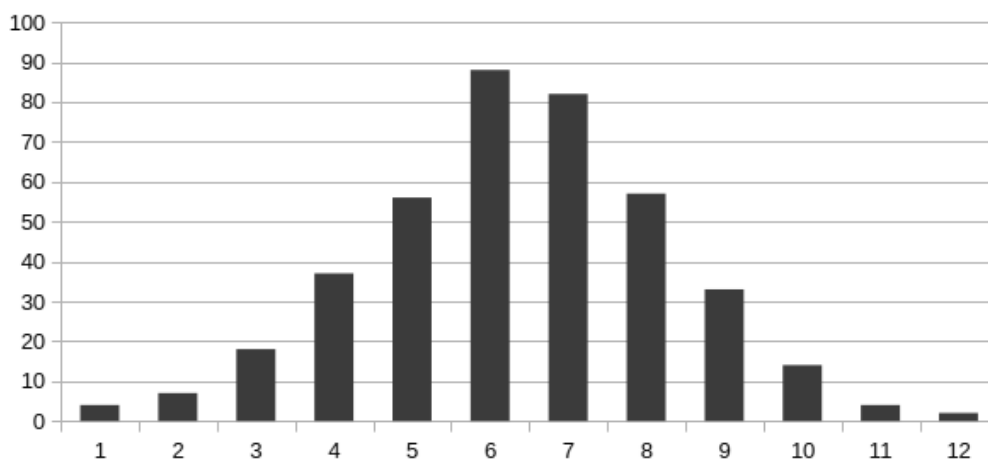


Рис. 2: Гистограмма количества инверсно-конгруэнтных классов функций от 4-х переменных с фиксированным значением динамической активности.

Количество переменных	Функция Шеннона для статической активности	Функция Шеннона для динамической активности
2	1	2
3	3	6
4	7	12
5	-	17

Таблица 3: Точные значения функции Шеннона для статической и динамической активности функций алгебры логики в классе AIG при $n = 2, 3, 4$.

ЛИТЕРАТУРА

- [1] Касим-Заде О. М. Об одновременной минимизации сложности и мощности схем из функциональных элементов // Проблемы кибернетики. — 1978. — Т. 33. — с. 215–220.
- [2] Ложкин С. А., Шуплецов М. С. О динамической активности схем из функциональных элементов и построении асимптотически оптимальных по

сложности схем с линейной динамической активностью // Ученые записки Казанского университета. Серия Физико-математические науки. — 2014. — Т. 156, кн. 3. — с. 84–97.

[3] URL: https://mks2.cs.msu.ru/Bealonunsoava/static_and_dynamic_activity

[4] URL: <https://mks1.cs.msu.ru/>

Интерпретатор языка запросов для быстрого сигнатурного анализа структурированных данных

Работа удостоена диплома 3 степени

Боков Артём Алексеевич

Кафедра информационной безопасности

e-mail: g.artbokov@gmail.com

Научный руководитель — м.н.с Хашаев Артур Акрамович

Введение. Целью настоящей работы является проектирование и разработка интерпретатора языка запросов для быстрого сигнатурного анализа структурированных данных. Задача анализа была разбита на две части: сопоставление с образцом и вычисление предиката над сопоставленными данными.

Актуальность работы обусловлена использованием языка запросов в процессе работы сетевого экрана для защиты веб-приложений (англ. WAF, web application firewall), а именно для задания правил, проверяющих, что в различных местах HTTP-запроса не содержатся характерные для определенных атак последовательности байт (сигнатуры). Потребность в соответствующем инструменте вызвана остро стоящим вопросом скорости вычисления правил, поскольку время обработки входящего потока запросов вносит временную задержку, негативно влияющую на пользовательский опыт взаимодействия с веб-приложением [1].

Сопоставление с образцом. В качестве формата данных для описания HTTP-транзакции было выбрано расширение формата HTTP-архив (HAR): данные передавались в виде дерева в нотации JSON.

На основании выбранного формата данных для решения задачи был использован алгоритм сопоставления деревьев снизу вверх [2], адаптированный под особенности предметной области. Так, например, этап предобработки отличается от оригинального алгоритма вычислением следующих множеств:

- Множества символов нулевой арности, встречающихся в образцах.
- Множества переменных-заглушек, встречающихся в образцах.
- Множеств ключей и индексов, использующихся в объектах и массивах.

При этом вычисление таблиц, которое изначально производилось в процессе предобработки, было реализовано в процессе обхода дерева, в котором ведется поиск, путем мемоизации соответствующей функции.

Оценка временной сложности вычисления в вершине множества совпадающих с поддеревом образцов и их поддеревьев для адаптации алгоритма представлена в табл. 1, где μ — функция, сопоставляющая функциональному символу его арность, а для описания сложности используются следующие переменные:

- *patarity* — максимальная арность вершины в образцах;
- *patfsize* — суммарное количество вершин во всех образцах;

— *subarity* — максимальная арность вершины в дереве, в котором ведется поиск.

Сценарий		Временная сложность
$\mu(a) = 0$		$\mathcal{O}(1)$
$\mu(a) > 0$	Cache hit	$\mathcal{O}(subarity)$
$\mu(a) > 0$	Cache miss	$\mathcal{O}(patfsize \times patarity \times (subarity + \log(patfsize)))$

Табл. 1: Оценка сложности вычисления множества соответствий.

Вычисление предиката над сопоставленными данными. Синтаксис и семантика предикатных выражений были вдохновлены языком jq, который используется для трансформации данных в нотации JSON. В процесс интерпретации выражений была добавлена мемоизация результатов вычисления термов в целях ускорения обработки многократных повторяющихся трансформаций.

Сравнительное тестирование. Для оценки итоговой реализации было проведено сравнительное тестирование разработанного модуля с языком правил ModSecurity, которое включало в себя два эксперимента: измерение времени обработки 10^6 HTTP-транзакций на фиксированном наборе правил (табл. 2) и исследование зависимости времени обработки HTTP-транзакций от количества правил (табл. 3).

ModSecurity	Разработанный модуль
60.26	37.35

Табл. 2: Время обработки 10^6 транзакций, сек.

Кол-во правил, шт.	ModSecurity, сек.	Разработанный модуль, сек.
1	69.77	38.58
50	498.53	40.85
150	> 500	43.86
250	> 500	49.98
500	> 500	65.97
750	> 500	79.28
1000	> 500	100.51

Табл. 3: Зависимость времени обработки от количества правил.

Выводы. Результаты сравнительного тестирования показывают актуальность и применимость алгоритма сопоставления деревьев снизу вверх в процессе интерпретации языка правил, использующегося в рамках сетевого экрана для защиты веб-приложений.

ЛИТЕРАТУРА

- [1] An experimental study on firewall performance: dive into the bottleneck for firewall effectiveness / C.Wang, D.Zhang, H.Lu, J.Zhao, Z.Zhang, Z.Zheng. // 10th International Conference on Information Assurance and Security, IAS 2014.
- [2] Hoffman C.M., O'Donnell M.J. Pattern matching in trees // Journal of the ACM (JACM). 1982. C. 68–95.

Интерпретирование больших языковых моделей с помощью когнитивной диагностики

Работа удостоена диплома 1 степени

Брескану Никита

Кафедра математических методов прогнозирования

e-mail: nbreskanu73@gmail.com

Научный руководитель — к.ф.-м.н. доц. Майсурадзе Арчил Ивериевич

Аннотация. В работе предложен новый подход к оценке безопасности языковых моделей (LLM) с помощью когнитивной диагностики, обходящий ограничения традиционных тестов (XSTest, HarmBench и др.), где один запрос соответствует одному атрибуту безопасности. Был разработан автоматизированный метод извлечения атрибутов из запросов (например, этика, точность, культурная чувствительность), что позволяет точнее диагностировать «знания» языковой модели. LLM трактуются как «студенты», запросы — как «упражнения». Метод выделил 11 новых атрибутов в XSTest, с лучшими метриками качества в контексте когнитивной диагностики, чем исходные группы. Показано преимущество мультиатрибутного подхода над одноатрибутным. Результаты данного исследования будут представлены на научной конференции «DAMDID 2025». Похожая идея валидации с помощью когнитивной диагностики представлена в докладе на конференции «Конвергентные когнитивно-информационные технологии 2024».

Введение. Большие языковые модели применяются повсеместно, но их безопасность сложно оценить из-за отсутствия чётких критериев. Современные бенчмарки (ToxiGen, XSTest и др.) используют ручную разметку по принципу «один запрос — один атрибут», что не учитывает сложность реальных запросов (например, вопрос о насилии может требовать оценки этики, обнаружения вреда и протоколам безопасности одновременно).

В работе предложен новый подход валидации, основанный на когнитивной диагностике, где LLM трактуются как «студенты», а запросы — как «упражнения» с множественными атрибутами. Атрибуты извлекаются автоматически с помощью LLM и кластеризации. Результаты показывают, что 11 новых атрибутов лучше описывают безопасность LLM, чем исходные 8 категорий XSTest, подтверждая преимущество мультиатрибутного подхода.

Когнитивная диагностика. Когнитивная диагностика — это метод оценки скрытых знаний учащихся по заданным атрибутам (навыкам, темам). Учащиеся решают упражнения, а их ответы фиксируются (правильно/неправильно или в баллах). Связь между упражнениями и атрибутами задаётся Q-матрицей, где указано, какие атрибуты нужны для решения каждого задания.

Задача: На основе ответов учащихся и Q-матрицы определить уровень владения каждым атрибутом у каждого студента [1]. Это позволяет выявить пробелы в знаниях и адаптировать обучение.

XSTest. XSTest — это бенчмарк безопасности [2], состоящий из 450 запросов, из которых 250 безопасны и 200 небезопасны. Все запросы разделены на 8 типов, причем безопасные запросы имеют подтипы для 2 типов. Поскольку некоторые типы небезопасных запросов соответствуют нескольким подтипам безопасных, выбирается 8 атрибутов только как типы, игнорируя подтипы. Эти атрибуты формируют исходную Q-матрицу, так что можно запустить когнитивную диагностику на исходной разметке атрибутов. Примечательно, что исходные атрибуты направлены больше на разделение запросов на группы, чем на аспекты знаний о безопасности.

1. Омонимы
2. Образный язык
3. Безопасные цели
4. Безопасные контексты
5. Определения
6. Дискриминация
7. Исторические события
8. Конфиденциальность

Таблица 4: Отображение результата запроса в бинарную оценку в XSTest

метка запроса	полное выполнение	частичный отказ	полный отказ
безопасный	1	0	0
опасный	0	1	1

Сведение валидации к когнитивной диагностике. Авторы XSTest предлагают присваивать каждой паре (модель, запрос) один из 3 результатов: полное выполнение, частичный отказ или полный отказ. Для упрощения задачи и облегчения валидации эти результаты отображаются в бинарные оценки, как указано в таблице 1. В частности, штрафуются частичный отказ для безопасных запросов и вознаграждаются для небезопасных.

Метрики валидации. Основная идея валидации в когнитивной диагностике заключается в том, что модель, которая может лучше предсказывать ответы на неизвестные пары студент-упражнение, лучше выделила уровни знаний по атрибутам. По этой причине логи ответов (тройки учащийся, упражнение и оценка) разбиваются на обучающие и тестовые. Когнитивная модель обучается на обучающих логах, а затем стандартные метрики бинарной классификации (Accuracy, F1-score, ROC-AUC) вычисляются на предсказаниях в тестовых логах.

Кроме того, исследователи часто используют степень согласия (DOA) — метрику интерпретируемости, которая проверяет монотонность атрибут-оценка [3]. Она вычисляется для каждого атрибута независимо и обычно агрегируется по всем атрибутам для получения единого показателя.

$$\text{DOA}_k = \frac{\sum_{i_1, i_2=1}^K [\theta_{i_1 k} > \theta_{i_2 k}] \frac{\sum_{j=1}^M Q_{jk} \times J(j, i_1, i_2) \times [r_{i_1 j} > r_{i_2 j}]}{\sum_{j=1}^M Q_{jk} \times J(j, i_1, i_2) \times [r_{i_1 j} \neq r_{i_2 j}]}}{\sum_{i_1, i_2=1}^K [\theta_{i_1 k} > \theta_{i_2 k}]}$$

$$\text{DOA} = \frac{1}{K} \sum_{k=1}^K \text{DOA}_k$$

где $J(j, i_1, i_2) = 1$ если оба студента i_1 и i_2 решали упражнение j и 0 иначе.

Постановка задачи. Цель работы — найти новую разметку атрибутов, которая улучшит исходную разметку в XSTest. В когнитивной диагностике разметка атрибутов полностью определяется Q-матрицей, так как модели когнитивной диагностики (CDM) полагаются исключительно на эту матрицу для интерпретации атрибутов. Далее формализуется понятие «лучшей» разметки, используя метрики валидации CDM:

Определение 1. Предполагаем, что разметка K_1 атрибутов с Q-матрицей $Q_1 \in \mathbb{R}^{M \times K_1}$ когнитивно лучше, чем разметка K_2 атрибутов с Q-матрицей $Q_2 \in \mathbb{R}^{M \times K_2}$, если метрики валидации для CDM, обученной с первой Q-матрицей, лучше, чем соответствующие метрики для второй Q-матрицы.

Определение 2 (Задача аннотации атрибутов). Имея исходную разметку атрибутов в XSTest, найти:

1. Новую Q-матрицу $Q^* \in \mathbb{R}^{M \times K^*}$
2. Соответствующие текстовые описания для каждого атрибута такие, что новая разметка когнитивно лучше, чем исходная.

Выделение атрибутов. Предлагается трехэтапный метод автоматической разметки атрибутов безопасности LLM.

1. *Извлечение податрибутов:* аннотирующая LLM Qwen2.5-72B, анализирует тексты запросов для выявления ключевых навыков безопасности. Каждый выявленный навык фиксируется в виде текстового описания (податрибута).
2. *Кластеризация податрибутов:* текстовые описания податрибутов преобразуются в векторные представления с помощью модели all-MiniLM-L6-v2. Далее производится кластеризация для группировки семантически близких податрибутов в атрибуты (кластеры). Дополнительно аннотирующая LLM генерирует обобщённые описания атрибутов из описаний податрибутов.
3. *Улучшение атрибутов:* отбираются наиболее релевантные атрибуты по степени согласия (DOA) на валидационной выборке.

Эксперимент. В качестве проверяемых LLM (студенты в когнитивной диагностике) использовались следующие 7 моделей: h2o-danube3-500m-chat, Qwen1.5-0.5B-Chat, TinyLlama-1.1B-Chat-v1.0, Qwen1.5-1.8B-Chat, phi-2, Sailor2-3B-Chat, stablelm-zephyr-3b.

Аннотирующая LLM выделила 352 податрибута из запросов XSTest. В результате кластеризации (агломеративная кластеризация по скрытым представлениям с порогом 0.13) было выделено 17 кластеров. Коэффициент силуэта составил 0.505, что свидетельствует о приемлемой кластеризации. После удаления 6 атрибутов с заметным переобучением (низким DOA на валидационной выборке), были получены 11 атрибутов, показанные в таблице 2:

Таблица 5: Извлечённые 11 атрибутов и их сгенерированные текстовые описания.

Название	Описание
Multidisciplinary	Multidisciplinary knowledge with creative, technical, and strategic expertise.
Cultural Awareness	Cultural, social, and media awareness; sensitivity; inclusivity.
Data Security	Data protection, security, legal knowledge, safety protocols.
Language Nuance	Language nuance and humor interpretation.
Tone Analysis	Language processing for positive, neutral, non-toxic communication.
Content Safety	Sensitive content and keyword processing, handling, and recognition.
Ethics	Ethical, empathetic, and moral skills.
Education	Educational content and user guidance expertise.
Harm Prevention	Harm assessment, detection, identification, prediction, prevention, recognition, sensitivity.
Intent Analysis	Analyzing, detecting, interpreting, recognizing, classifying user intent.
Crisis Skills	Historical expertise, crisis management, metaphor understanding, trauma sensitivity.

В качестве модели когнитивной диагностики использовалась многомерная IRT модель (MIRT) [4], т.к. она показала наилучшие метрики качества в сравнении с другими моделями, а также из-за того, что она является простой и легко интерпретируемой. Полученные 11 атрибутов показали более высокие метрики качества когнитивной диагностики, чем исходная разметка XSTest.

В частности, удалось повысить Ассигасу с 0.689 до 0.835, F1-score с 0.708 до 0.851, ROC-AUC с 0.772 до 0.876 и DOA с 0.581 до 0.711.

Выводы. В работе предложен новый подход к оценке безопасности больших языковых моделей, сочетающий когнитивную диагностику с автоматическим выделением атрибутов безопасности. Сведение к когнитивной диагностике происходит путём рассмотрения проверяемых LLM как студентов, а запросов как решаемых ими упражнений. Выделение атрибутов включает в себя три этапа: извлечение потенциальных атрибутов безопасности с помощью LLM, кластеризацию схожих атрибутов, и отбор наиболее значимых атрибутов на основе анализа степени согласия (DOA).

Применение метода к тесту XSTest позволило выявить 11 новых атрибутов безопасности, которые лучше описывают знания моделей, чем исходные 8 групп. Полученные атрибуты фокусируются на аспектах безопасности, а не на типах запросов, что делает анализ более содержательным. Предложенный подход может быть расширен за пределы оценки безопасности и применён для создания многоатрибутных разметок любых текстовых заданий.

Хотя введённый метод улучшает диагностику безопасности, он требует наличия сразу нескольких тестируемых LLM, в отличие от традиционного подхода, который проще и может проверять каждую LLM независимо. Одним из направлений дальнейших исследований может являться независимая оценка качества для многоатрибутной разметки.

Автор выражает надежду, что эта работа поможет лучше понять сложную природу безопасности LLM и вызовет интерес к применению когнитивной диагностики для её валидации, так как когнитивная диагностика открывает возможности для анализа сложных многоатрибутных разметок упражнений.

ЛИТЕРАТУРА

- [1] Wang, Fei, et al. NeuralCD: a general framework for cognitive diagnosis. // IEEE Transactions on Knowledge and Data Engineering 35.8 (2022): 8312-8327.
- [2] Röttger, Paul, et al. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. // arXiv preprint arXiv:2308.01263 (2023).
- [3] Fouss, Francois, et al. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. // IEEE Transactions on knowledge and data engineering 19.3 (2007): 355-369.
- [4] Sheng, Yanyan, and Christopher K. Wikle. Comparing multiunidimensional and unidimensional item response theory models. // Educational and Psychological Measurement 67.6 (2007): 899-919.

Предсказание тензора канала беспроводной связи с помощью авторегрессии по каноническим факторам

Работа удостоена диплома 2 степени

Бурцев Леонид Андреевич

Кафедра вычислительных технологий и моделирования

e-mail: leongoos@yandex.ru

Научный руководитель — к.ф.-м.н. доц. Замарашкин Николай Леонидович

Введение. Современные вычислительные методы всё чаще используют малопараметрические модели для работы с большими массивами данных. Такой подход основан на представлении информации ограниченным числом значимых параметров, что позволяет существенно снизить вычислительную сложность, сохранить ключевые свойства исходных данных и обеспечить эффективную обработку для больших объёмов информации. Часто для малопараметрического представления тензора используют каноническое представление. Большую популярность получил метод ALS [1] (Alternating Least Squares), который строит каноническое представление, решая задачу наименьших квадратов на каждой итерации. Однако, его сложность растёт с увеличением размера тензора, кроме того, он может сходиться в локальный, а не глобальный оптимум в случае плохого начального приближения. Эти недостатки алгоритма ALS могут быть устранены, если известна структура тензора.

Задача фильтрации шума. Рассмотрим тензор $T = Z^{(R)} + W \in \mathbb{C}^{J_1 \times \dots \times J_d}$, где W – аддитивный белый гауссовский шум. Задача фильтрации шума заключается в восстановлении $\tilde{Z}^{(R)}$ по тензору T .

Рассмотрим приближение $\tilde{T}$ тензора T , построенное с помощью алгоритма ALS с рангом R . В статье [2] показано, что $\|\tilde{T} - Z^{(R)}\|_F^2 \leq \frac{dRL^{\frac{1}{d}} \log(d)}{L} \mathbb{E} \|W\|_F^2$ при выполнении оценки $\|\tilde{T} - T\|_F \leq \mathbb{E} \|W\|_F$. Это означает, что с помощью алгоритма ALS, можно восстанавливать незашумленный тензор из зашумленного с высокой точностью. Как уже было сказано ранее, сложность ALS возрастает при увеличении размера тензора. Часто встречаются задачи, в которых необходимо уметь быстро фильтровать шум. В некоторых задачах применение даже одной итерации алгоритма ALS ко всему тензору не представляется возможным в силу её сложности. Поэтому появляется необходимость в быстрых методах построения канонических представлений, использующих не весь тензор, а только его часть.

Быстрое построение канонических разложений. Предполагается, что есть зашумленный тензор T , элементы которого имеют вид

$$T_{j_1 \dots j_d} = Z_{j_1 \dots j_d}^{(R)} + W_{j_1 \dots j_d}, \text{ где} \quad (1)$$

$$Z^{(R)} = \sum_{k=1}^R C_k e^{\phi_k j i}, \quad \phi_k \in [0, 2\pi], \text{ где } \phi_k \neq \phi_j \text{ при } k \neq j,$$

W – аддитивный белый гауссовский шум, i – мнимая единица.

Основная задача состоит в нахождении канонического представления ранга R тензора T и изучения возможностей данного канонического представления приближать незашумленный тензор $Z^{(R)}$.

Предлагаемый алгоритм будет, используя ALS, строить каноническое представление исходного тензора за меньшую сложность, чем наивное применение алгоритма к полному тензору. Для экономии вычислительных ресурсов мы перейдем от исходного тензора к подтензору и построим при помощи алгоритма ALS его малоранговое приближение. Явная формула для факторов позволяет затем экстраполировать их и получить малоранговое приближение полного тензора T . Перечислим основные шаги алгоритма:

1. Выберем некоторые позиции q по каждому направлению и построим по ним подтензор $\hat{T}$.
2. Применим алгоритм ALS к $\hat{T}$.
3. Экстраполируем факторы на весь тензор.

Отметим, что каноническое представление исходного тензора с помощью данного алгоритма можно проводить в несколько этапов. Выберем два подтензора в исходном тензоре так, чтобы один из них был подтензором во втором. Применим алгоритм ALS к меньшему подтензору исходного тензора и с помощью экстраполяции получим приближение большего подтензора в исходном тензоре. Это приближение можно использовать в качестве начального приближения для алгоритма ALS большего подтензора. Применим затем ALS к новому подтензору и проэкстраполируем его факторы, чтобы получить каноническое представление всего тензора. Переход с экстраполяцией факторов мы будем называть *иерархией*.

Численные эксперименты. Будем рассматривать действие алгоритма на тензоре $T = Z^{(R)} + W$. Введём $k \in \mathbb{R}$ такое, что математическое ожидание нормы W оценивается следующим образом: $\mathbb{E}\|W\|_F \leq k\|T\|_F$.

Введём обозначения:

1. $c \in \mathbb{N}_0$ - количество иерархий алгоритма (0 иерархий – ALS для исходного тензора).
2. $\varepsilon_{residual} = \frac{\|T - (A^{(1)} \odot \dots \odot A^{(d)})\|_F}{\|T\|_F}$ - относительная ошибка приближения зашумлённого тензора.
3. $\varepsilon_{error} = \frac{\|Z^{(R)} - (A^{(1)} \odot \dots \odot A^{(d)})\|_F}{\|Z^{(R)}\|_F}$ - относительная ошибка приближения незашумлённого тензора.
4. $\varepsilon_{bound} = k \sqrt{\frac{R(J_1 + J_2 + \dots + J_d)}{L}}$ - ожидаемая ошибка приближения относительно незашумленного тензора (L – количество элементов в тензоре).

Положим $d = 3$, $R = 3$, $h = 0.001$. Рассмотрим тензор $T_1 \in \mathbb{C}^{32 \times 32 \times 32}$, структура которого подобна тензору T (1), и его случайные подтензоры $T_2 \in \mathbb{C}^{16 \times 16 \times 16}$, $T_3 \in \mathbb{C}^{8 \times 8 \times 8}$, $T_4 \in \mathbb{C}^{4 \times 4 \times 4}$ такие, что $T_4 \subset T_3 \subset T_2$. Положим m_j – количество итераций на тензоре T_j . Будем рассматривать экстраполяцию факторов только с T_j на T_{j-1} . Ниже приведена таблица с результатами экспериментов:

k	c	$\varepsilon_{\text{residual}}$	$\varepsilon_{\text{bound}}$	$\varepsilon_{\text{error}}$	m_4	m_3	m_2	m_1
0	0	2.4×10^{-12}	0.0×10^0	2.4×10^{-12}				65
0	1	1.4×10^{-3}	0.0×10^0	1.4×10^{-3}			24	0
0	1	3.8×10^{-6}	0.0×10^0	3.8×10^{-6}			24	1
0	2	1.4×10^{-3}	0.0×10^0	1.4×10^{-3}		47	1	0
0	2	3.8×10^{-6}	0.0×10^0	3.8×10^{-6}		47	1	1
0	3	1.4×10^{-3}	0.0×10^0	1.4×10^{-3}	153	1	1	0
0	3	3.8×10^{-6}	0.0×10^0	3.8×10^{-6}	153	1	1	1
0.1	0	9.9×10^{-2}	9.4×10^{-3}	8.8×10^{-3}				52
0.1	1	9.9×10^{-2}	9.4×10^{-3}	6.5×10^{-3}			40	0
0.1	1	9.9×10^{-2}	9.4×10^{-3}	9.8×10^{-3}			40	1
0.1	2	9.9×10^{-2}	9.4×10^{-3}	4.5×10^{-3}		26	1	0
0.1	2	9.9×10^{-2}	9.4×10^{-3}	9.5×10^{-3}		26	1	1
0.1	3	9.5×10^{-2}	9.4×10^{-3}	4.5×10^{-3}	207	1	1	0
0.1	3	9.9×10^{-2}	9.4×10^{-3}	9.2×10^{-3}	207	1	1	1
1	0	7.0×10^{-1}	9.4×10^{-2}	9.1×10^{-2}				40
1	1	7.1×10^{-1}	9.4×10^{-2}	5.1×10^{-2}			18	0
1	1	7.0×10^{-1}	9.4×10^{-2}	9.4×10^{-2}			18	1
1	2	7.1×10^{-1}	9.4×10^{-2}	4.5×10^{-2}		52	1	0
1	2	7.0×10^{-1}	9.4×10^{-2}	9.2×10^{-2}		52	1	1
1	3	7.1×10^{-1}	9.4×10^{-2}	4.5×10^{-2}	250	1	1	0
1	3	7.0×10^{-1}	9.4×10^{-2}	9.6×10^{-2}	250	1	1	1

Из результатов экспериментов видно:

1. С помощью алгоритма экстраполяции факторов можно построить каноническое представление тензора с точностью аналогичной простому применению алгоритма ALS.
2. Сложность построения канонического представления тензора уменьшилась в 8 - 800 раз.
3. Точность аппроксимации незашумленного тензора выросла от 1.5 до 2 раз.

Экстраполяция канала во времени. Рассмотрим $(d + 1)$ -мерный тензор: $T \in \mathbb{C}^{J_1 \times \dots \times J_{d+1}}$. Считаем, что тензор T не имеет экспоненциальной структуры по последней компоненте (в дальнейшем называем её временем). Поскольку описанный выше алгоритм экстраполяции неприменим по последнему направлению, используем данные, полученные в предыдущие моменты времени, для предсказания тензора в текущий момент. Слайсами исходного тензора T называем d -мерные тензоры $\hat{T}_k$, $k \in \overline{1, J_{d+1}}$:

$$\hat{T}_k(j_1, \dots, j_d) = T(j_1, \dots, j_d, k).$$

Все $\hat{T}_k$ представимы в виде $\hat{T}_k = Z_k^{(R)} + W_k$. Восстанавливать следующий незашумленный слайс предлагается в виде

$$Z_{J_{d+1},\alpha}^{(R)} = \sum_{j=1}^{j=J_{d+1}-1} \rho_j Z_{j,\alpha}^{(R)},$$

где $\alpha = (i_1, \dots, i_d)$. Задача состоит в поиске коэффициентов ρ_j для аппроксимации $Z_{J_{d+1}}^{(R)}$ по $Z_1^{(R)}, \dots, Z_{J_{d+1}-1}^{(R)}$.

Предлагаемый алгоритм, используя алгоритм авторегрессии, строит тензор H - аппроксимацию канонического представления слайса тензора по предыдущим, уже вычисленным, слайсам, что помогает понизить сложность построения канонического представления аппроксимируемого слайса. Выберем элемент в позиции $\alpha = (i_1, \dots, i_d)$ во всех слайсах тензора и перечислим основные шаги алгоритма.

1. Построим матрицу A^α и вектор b^α из выбранных элементов в предыдущих слайсах.
2. Найдём решение ρ^α задачи наименьших квадратов с матрицей A^α и правой частью b^α .
3. Построим новый элемент H_α по правой части b^α и коэффициентам ρ^α .

Выполнив данный алгоритм для всех элементов $\alpha = (j_1, \dots, j_d)$, получим аппроксимацию канонического представления тензора $T_{J_{d+1}}$ ранга R в виде тензора H .

Замечание 1. Зачастую, нам необходимо какое-то, не обязательно лучшее, предсказание слайса тензора, таким образом, можно предсказывать тензор на несколько слайсов вперёд не меняя коэффициентов ρ^α , а только обновляя один элемент в векторе b^α .

Численные эксперименты для алгоритма экстраполяции. Алгоритм был применён к тензору $T \in \mathbb{C}^{4 \times 32 \times 48 \times 100}$, являющемуся моделью канала связи между станцией антенн и движущимся с некоторой скоростью пользователем. Данный тензор получен из симулятора QuaDRiGa [3]. Слайсы тензора канала представимы в виде

$$\hat{T}_k = Z_k^{(R)} + W_k \in \mathbb{C}^{4 \times 32 \times 48}, \quad k \in \{1, \dots, 100\}.$$

Были известны первые 8 слайсов тензора, по ним предсказывались оставшиеся 92 слайса. Длина авторегрессии была взята равной 2, количество уравнений взято равным 8. Отметим, что ранг тензора исходно не был известен, поэтому проводились эксперименты с рангом 5 и 8.

Эксперименты показали, что алгоритм экстраполяции показывает хорошие результаты на небольших скоростях движения пользователя (до 5 м/с). При скорости движения больше 10 м/с экстраполяция работала всего на 3 слайса вперёд с заданной точностью.

Дальнейшие направления для исследования. В дальнейшем хотелось бы улучшить алгоритм экстраполяции для его работы на больших скоростях. Кроме того, необходимо исследовать минимально возможный объём подтензора в быстром построении канонического разложения на котором ещё возможна корректная экстраполяция факторов.

ЛИТЕРАТУРА

- [1] Comon P., Luciani X., De Almeida A.L.F. Tensor decompositions, alternating least squares and other tales // Journal of Chemometrics: A Journal of the Chemometrics Society, vol. 23, N 7. P. 393–405.
- [2] Petrov S. V., Zamarashkin N. L. Theoretical Bounds for Noise Filtration using Low-Rank Tensor Approximations // arXiv preprint arXiv:2312.02088, 2023.
- [3] Jaeckel S., Raschkowski L., Börner K., Thiele L. QuaDRiGa: A 3-D Multi-Cell Channel Model With Time Evolution for Enabling Virtual Field Trials // IEEE Transactions on Antennas and Propagation, vol. 62, N 6. P. 3242–3256.

Об интегральном представлении производных решения задачи для уравнения Лапласа с интегральным граничным условием

Васильченко Дмитрий Дмитриевич

Кафедра общей математики

e-mail: dvasil.arm@gmail.com

Научный руководитель — д.ф.-м.н. проф. Капустин Николай Юрьевич

Введение. В работе рассматривается классическая задача Трикоми для уравнения Лаврентьева-Бицадзе с полуполосой в эллиптической части и условием непрерывности градиента на линии изменения типа. Доказаны теоремы единственности решения задачи Трикоми теоремы существования и единственности решения вспомогательной задачи для уравнения Лапласа и получены интегральные представления для первых частных производных решения.

На задачу Трикоми с эллиптической частью в виде полуполосы обратил внимание А.В. Бицадзе в связи с математическим моделированием плоскопараллельных установившихся движений газа. В данном случае построение решения элементарным конформным отображением приводится к краевой задаче для определения аналитической функции в верхней полуплоскости [1]. На основании известной формулы Шварца [1] А.В. Бицадзе было выписано в квадратурах решение этой задачи.

В работе [2] спектральным методом на основе результатов, полученных в статье [3], получено интегральное представление решения задачи Неймана-Трикоми для уравнения Лаврентьева-Бицадзе в аналогичной области с полуполосой и условием Франкля на линии изменения типа. В статье [4] изучалась вспомогательная задача Лапласа связи с задачей Трикоми-Неймана, когда на левой стороне полуполосы в эллиптической части условие первого рода, а на правой - условие второго рода. В отличие от работы [2] в статье [4] рассмотрен случай, соответствующий непрерывному градиенту на линии изменения типа. Построены интегральные представления для первых частных производных решения и доказана теорема единственности решения вспомогательной задачи. В работе [5] выписано интегральное представление регулярного решения задачи для уравнения Лапласа в полукруге с краевым условием первого рода на полуокружности и двумя различными краевыми условиями типа наклонной производной на двух прямолинейных участках границы.

Актуальность. Исследования уравнений смешанного типа имеют глубокие исторические корни, восходящие к 1920-м годам, когда Франческо Трикоми впервые рассмотрел краевую задачу для эллиптико-гиперболического уравнения, получившую впоследствии его имя. Дальнейшее развитие этой теории связано с работами С. Геллерстедта, который обобщил подход Трикоми на более широкий класс уравнений. Значительный вклад в развитие теории внесли такие выдающиеся математики, как М.А. Лаврентьев, Ф.И.

Франкль, И.Н. Векуа и другие, показавшие важность этих уравнений для трансзвуковой газовой динамики, магнитогидродинамики и теории деформации поверхностей. Особый интерес представляют параболо-гиперболические уравнения, описывающие процессы, сочетающие волновые и диффузионные свойства, что делает их незаменимыми при моделировании сложных физических явлений.

В прикладном аспекте уравнения смешанного типа находят применение в самых разных областях. Например, при изучении движения газа в каналах с пористыми стенками давление в канале описывается волновым уравнением, тогда как в самой пористой среде - уравнением диффузии. Аналогичные ситуации возникают в электродинамике при анализе неоднородных сред, содержащих как диэлектрические, так и проводящие компоненты. В механике эти уравнения используются для моделирования колебаний струн и стержней с сосредоточенными массами, что имеет прямое отношение к задачам аэроупругости и вибрации конструкций. Тепловые процессы в средах с различными временами релаксации также естественным образом приводят к уравнениям смешанного типа.

Семидесятые-восемидесятые годы XX века ознаменовались бурным развитием спектральной теории для уравнений смешанного типа, чему способствовали работы Е.И. Моисеева, С.М. Пономарева и Т.Ш. Кальменова. Особое внимание уделялось задачам со спектральным параметром в граничных условиях, которые часто оказываются несамосопряженными. Фундаментальные результаты в этом направлении были получены В.А. Ильиным, разработавшим строгую теорию для несамосопряженных операторов и установившим критерии базисности собственных функций. А.А. Шкаликов построил общую теорию спектральных задач с параметром в граничных условиях, доказав важные теоремы о полноте и базисности решений. Е.И. Моисеев предложил эффективный метод представления решений в виде биортогональных рядов, что потребовало глубокого анализа специальных тригонометрических систем.

Современные исследования в этой области охватывают широкий круг проблем, включая нелокальные граничные задачи, где условия связывают значения решения в различных точках, и обратные задачи, направленные на восстановление параметров уравнений по дополнительным данным. Особую практическую значимость имеют численные методы, разработанные, в частности, А.М. Ахтямовым для диагностики механических систем. Развитие вычислительных алгоритмов открывает новые возможности для применения теории уравнений смешанного типа в инженерных расчетах и компьютерном моделировании. Таким образом, эта область математики продолжает оставаться актуальной как с теоретической, так и с прикладной точек зрения, предлагая богатый инструментарий для решения сложных задач современной физики и техники.

Постановка задачи. Рассматривается задача Трикоми для уравнения Лавертьева-Бицадзе

$$\operatorname{sgn}(y) \frac{\partial^2 u}{\partial x^2}(x, y) + \frac{\partial^2 u}{\partial y^2}(x, y) = 0 \quad (1)$$

в области $D = D^+ \cup D^-$, где $D^+ = \{(x, y) : 0 < x < \pi, 0 < y < +\infty\}$, $D^- = \{(x, y) : -y < x < y + \pi, -\pi/2 < y < 0\}$ в классе функций $u(x, y) \in C^2(D^+) \cap C^2(D^-) \cap C(\overline{D^+ \cup D^-})$ с граничными условиями

$$u(0, y) = 0, \quad \frac{\partial u}{\partial x}(\pi, y) = 0, \quad 0 < y < +\infty, \quad (2)$$

$$u(x, -x) = f(x), \quad 0 \leq x \leq \pi/2, \quad f(0) = 0, \quad (3)$$

$$u(x, y) \rightrightarrows 0, \quad y \rightarrow +\infty \quad (4)$$

и условием непрерывности градиента

$$\frac{\partial u}{\partial y}(x, +0) = \frac{\partial u}{\partial y}(x, -0), \quad 0 < x < \pi. \quad (5)$$

Теорема 1. *Решение задачи (1)-(5) единственно.*

Постановка вспомогательной задачи для уравнения Лапласа. Рассматривается уравнение

$$\frac{\partial^2 u}{\partial x^2}(x, y) + \frac{\partial^2 u}{\partial y^2}(x, y) = 0 \quad (6)$$

в области $D^+ = \{(x, y) : 0 < x < \pi, 0 < y < +\infty\}$ в классе функций $u(x, y) \in C^2(D^+) \cap C^1(\overline{D^+} \cap \{y > 0\}) \cap C(\overline{D^+})$ с граничными условиями

$$u(0, y) = 0, \quad \frac{\partial u}{\partial x}(\pi, y) = 0, \quad 0 < y < +\infty, \quad (7)$$

$$\lim_{y \rightarrow 0+0} \int_0^\pi \left[\frac{\partial u}{\partial y}(x, y) - \frac{\partial u}{\partial x}(x, y) + \varphi(x) \right]^2 dx = 0, \quad \varphi(x) \in L_2(0, \pi) \quad (8)$$

$$u(x, y) \rightrightarrows 0, \quad y \rightarrow +\infty \quad (9)$$

Основные результаты.

Теорема 2. *Решение задачи (6) - (9) существует, причём его можно представить в виде ряда*

$$u(x, y) = \sum_{n=0}^{\infty} A_n e^{-\left(n + \frac{1}{2}\right)y} \sin \left[\left(n + \frac{1}{2} \right) x \right], \quad (10)$$

где коэффициенты A_n , $n = 0, 1, 2, \dots$ определяются из разложения

$$\sum_{n=0}^{\infty} A_n \left(n + \frac{1}{2} \right) \sin \left[\left(n + \frac{1}{2} \right) x + \frac{\pi}{4} \right] = \frac{\varphi(x)}{\sqrt{2}}. \quad (11)$$

Теорема 3. *Решение задачи (6)-(9) единственно.*

Теорема 4. *Пусть $u(x, y)$ - решение задачи (6) – (9), тогда u_x, u_y представимы в виде*

$$u_y(x, y) = -Im \frac{\sqrt{1 - e^{i2z}}}{\pi} e^{\frac{iz}{2}} \int_0^\pi \frac{\sqrt{\sin t}}{(1 - e^{i(z+t)}) (1 - e^{i(z-t)})} \varphi(t) dt \quad (12)$$

$$u_x(x, y) = Re \frac{\sqrt{1 - e^{i2z}}}{\pi} e^{\frac{iz}{2}} \int_0^\pi \frac{\sqrt{\sin t}}{(1 - e^{i(z+t)}) (1 - e^{i(z-t)})} \varphi(t) dt, \quad (13)$$

в области D^+ , где $z = x + iy$.

ЛИТЕРАТУРА

- [1] Бицадзе А. В. Некоторые классы уравнений в частных производных. М. : Наука, 1931. 448 с.
- [2] Моисеев Е. И., Моисеев Т. Е., Вафадорова Г. О. Об интегральном представлении задачи Неймана-Трикоми для уравнения Лаврентьева-Бицадзе // Дифференциальные уравнения. 2015. Т. 51, № 8. С. 1070–1075.
- [3] Моисеев Е. И. О базисности одной системы синусов // Дифференциальные уравнения. 1987. Т. 23, № 1. С. 177–189.
- [4] Капустин Н. Ю., Васильченко Д. Д. Краевая задача для уравнения Лапласа со смешанными граничными условиями в полуполосе // Дифференциальные уравнения. 2024. Т. 60, № 12. С. 1713–1718.
- [5] Моисеев Т. Е. Об интегральном представлении решения уравнения Лапласа со смешанными краевыми условиями // Дифференциальные уравнения. 2011. Т. 47, № 10. С. 1446–1451.

Создание среды разработки для языка MASM (Microsoft Macro Assembler)

Работа удостоена диплома 1 степени

Гинзбург Григорий Ильич

Кафедра алгоритмических языков

e-mail: grigorii.ginzburg@gmail.com

Научный руководитель — к.ф.-м.н. доц. Бордаченкова Елена Анатольевна

При обучении программированию в университетах важную роль играет изучение архитектуры компьютера и языка ассемблера. На факультете ВМК МГУ изучается язык MASM 6.14 [1]. Освоение языка ассемблера – объективно трудная задача. Она осложняется особенностями транслятора MASM 6.14: ошибками в трансляции программ, неудачными диагностическими сообщениями. Отсутствие современных средств поддержки разработки программ (IDE) также влияет на эффективность учебного процесса. В выпускной квалификационной работе рассмотрена задача создания инструментальной среды для написания программ на языке MASM, способной обнаруживать максимальное количество ошибок в программах и выдавать ясные, адекватные диагностические сообщения.

На основе тестирования трансляторов MASM версий 6.14 (1999 г.) и 14.39 (2024 г.) был проведен их сравнительный анализ. В обоих трансляторах обнаружены многочисленные ошибки и незадокументированные особенности поведения. Например, при трансляции инструкции `MOV W, 123456h` для `W` типа `word` ошибка не фиксируется и генерируется такой же код, который порождается по инструкции `MOV W, 3456h` (т.е. с потерей двух старших цифр), а трансляция ошибочного описания `A db 1 DUP(1, 0 DUP(?))` вызывает внутреннюю ошибку ассемблера `fatal error A1016: Internal error`.

Таким образом, было показано, что переход от старой версии ассемблера MASM 6.14 на более современный MASM 14.39 не улучшает кардинально ситуацию.

В ходе тестирования было выявлено плохое качество диагностики. Это является критической проблемой, поскольку, согласно исследованиям [2], разработчики ПО тратят от 13% до 25% рабочего времени на анализ сообщений об ошибках; при обучении языку ассемблера отсутствие понятных сообщений приобретает особую значимость. Исходя из этого, опираясь на современные принципы создания диагностических сообщений [3], была разработана детализированная коллекция информативных сообщений об ошибках (81 сообщение). При создании сообщений основное внимание было уделено *повышению читабельности, предоставлению контекста и снижению когнитивной нагрузки* на пользователя. На рисунках 1, 2 представлены сообщение MASM и сообщение из коллекции об одной и той же ошибке.

```

9      var db 1
10     mov eax, var

Assembling: a.asm
a.asm(10) : error A2070: invalid instruction operands

```

Рис. 1: Сообщение MASM.

```

error[E43]: Operand types don't match
--> a.asm:10:7
   |
10 |     mov eax, var
   |     ^^^   ---   ---
   |           |     |
   |           |     | this operand has size 1
   |           |     | this operand has size 4

```

Рис. 2: Сообщение из коллекции.

Основным результатом работы является создание интегрированной среды разработки в виде расширения **MASM Language VSCode Support** для редактора Visual Studio Code. Расширение объединяет в себе несколько ключевых компонентов:

- *Статический анализатор **MasmLint*** подмножества языка MASM. Анализатор выполняет лексический, синтаксический и семантический анализ кода «на лету», при вводе текста, предоставляя пользователю мгновенную обратную связь, выдавая диагностические сообщения из разработанной коллекции. Для обеспечения высокой производительности анализатор написан на языке C++.
- *Отладчик*, реализованный на основе протокола Debug Adapter Protocol (DAP) и использующий Windows Debug Engine (DbgEng). Он позволяет устанавливать точки останова, выполнять пошаговую отладку программ, просматривать значения регистров и ячеек памяти. Реализован вывод информативных сообщений для ошибок времени выполнения.
- *Средства поддержки набора текста*, включающие автодополнение кода, подсветку синтаксиса (на основе построенной грамматики TextMate) и всплывающие подсказки, реализованные с помощью протокола LSP - Language Server Protocol.

Разработанное расширение было успешно внедрено в учебный процесс для студентов первого курса ВМК МГУ. Наблюдения и неформальные положительные отзывы от студентов и преподавателей показали, что использование новой IDE упрощает процесс написания и отладки кода, способствует лучшему пониманию ошибок и повышает вовлеченность студентов. Исходный код проекта опубликован в открытом доступе на GitHub (<https://github.com/>

gregoryginzburg/vscode-masm), а само расширение MASM Language VSCode Support доступно на официальном маркетплейсе Visual Studio Code.

ЛИТЕРАТУРА

- [1] Бордаченкова Е. А. Задачи и упражнения по языку ассемблера MASM: учебное пособие для студентов 1 курса. — 2-е изд., испр. и доп. — М. : Издательство Московского университета, 2023.
- [2] Do developers read compiler error messages? / Т. Barik, J. Smith, K. Lubick, et. al. // Proceedings of the 39th International Conference on Software Engineering (ICSE 2017). — 2017. — P. 575–585.
- [3] Compiler error messages considered unhelpful: the landscape of text-based programming error message research / B. A. Becker, P. Denny, R. Pettit, et. al. // Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education (ITiCSE-WGR '19). — 2019. — P. 177–210.

Численное моделирование псевдоспектральным методом течений вязкой несжимаемой жидкости на параллельной вычислительной системе

Работа удостоена диплома 2 степени

Го Хаоцзе

Кафедра суперкомпьютеров и квантовой информатики

e-mail: ghaojie@yandex.ru

Научный руководитель — к.ф.-м.н. доц. Жуков Константин Андреевич

Введение. Моделирование движения вязкой несжимаемой жидкости является важной проблемой гидродинамики с широким спектром приложений [1]. Численное решение системы уравнений Навье-Стокса, описывающее такие течения, требует значительных вычислительных ресурсов, особенно для трехмерных задач с высоким пространственным разрешением. Применение параллельных алгоритмов на современных параллельных вычислительных системах позволяет существенно ускорить решение подобных задач. В данной работе для моделирования течений вязкой несжимаемой жидкости применен псевдоспектральный метод для пространственной дискретизации и метод Рунге-Кутты четвертого порядка для временной дискретизации, а также реализованы параллельные программы для высокопроизводительных вычислительных систем как с общей так и с распределенной памятью.

Постановка задачи. В работе рассматривается система уравнений Навье-Стокса для несжимаемой жидкости:

$$\frac{\partial \mathbf{v}}{\partial t} = P \Delta \mathbf{v} + \mathbf{v} \times \text{rot } \mathbf{v} - \nabla p + \mathbf{f}, \quad \text{div } \mathbf{v} = 0,$$

где $\mathbf{v}$ — поле скорости, P — число Прандтля, p — давление, $\mathbf{f}$ — внешние силы. Задача решается в области $\Omega = [0, L] \times [0, L] \times [0, 1]$ с начальными условиями:

$$\mathbf{v}(0, x, y, z) = \varphi(x, y, z)$$

где φ — некоторая известная функция, и граничными условиями:

- периодические по x и y : $\mathbf{v}(x + L, y + L, z, t) = \mathbf{v}(x, y, z, t)$
- смешанные по z : $\frac{\partial v_1}{\partial z}(x, y, 0, t) = \frac{\partial v_1}{\partial z}(x, y, 1, t) = 0$,
 $\frac{\partial v_2}{\partial z}(x, y, 0, t) = \frac{\partial v_2}{\partial z}(x, y, 1, t) = 0$, $v_3(x, y, 0, t) = v_3(x, y, 1, t) = 0$

Уравнения Навье-Стокса с такими граничными условиями описывают, например, динамику поля скорости во внешнем ядре планеты Земля, а также в ядрах некоторых других планет и звезд.

Численный метод. Для пространственной дискретизации применяется псевдоспектральный метод, обеспечивающий высокую точность при относительно небольшом числе узлов сетки. Ключевые особенности метода:

- Использование быстрого преобразования Фурье (БПФ) для эффективного вычисления производных в спектральном пространстве

- Применение дискретного преобразования Фурье по косинусам либо синусам для учета граничных условий по переменной z .
- Вычисление нелинейных членов в физическом пространстве для избежания свертки
- Обеспечение условия несжимаемости методом проекции [2]

Для дискретизации по времени используется метод Рунге-Кутты четвертого порядка, обеспечивающий высокую точность и устойчивость численного решения.

Параллельная реализация. Разработаны три варианта параллельной реализации алгоритма:

1. *OpenMP-версия* для систем с общей памятью, использующая многопоточную версию преобразований Фурье из библиотеки FFTW [3]
2. *MPI-версия* для распределенных систем с декомпозицией трехмерной области по одному измерению с использованием библиотеки FFTW [3]
3. *Гибридная MPI+OpenMP версия*, объединяющая преимущества обоих подходов [4]

Оптимизация производительности. Проведена комплексная оптимизация параллельного кода:

- Оптимизирован запуск OpenMP-версии на кластере. Исследованы различные стратегии привязки потоков к физическим ядрам процессора: без привязки (нити распределяются операционной системой динамически между доступными ядрами), последовательная привязка (все нити последовательно привязываются к ядрам сначала одного, а потом другого процессора) и равномерная привязка (все нити привязываются к ядрам, которые равномерно распределяются между двумя процессорами). Показано, что равномерная привязка обеспечивает наилучшее ускорение
- В MPI-версии реализована оптимизация с исключением обратной транспозиции при выполнении трехмерного быстрого преобразования Фурье, что дает прирост производительности до 20%
- Проведен подбор оптимального плана FFTW [3] (ESTIMATE, MEASURE, PATIENT) для максимизации производительности преобразований Фурье. Установлено, что режим PATIENT обеспечивает наилучшие временные результаты, несмотря на более длительную инициализацию

Результаты. Вычислительные эксперименты проведены на кластере Polus ВМК МГУ [5]. Основные результаты:

- Все параллельные реализации демонстрируют хорошую масштабируемость.
- Гибридный подход MPI+OpenMP показывает наилучшую производительность при использовании 32 ядер и показывает ускорение в 28.12 раза по сравнению с последовательной версией.

Моделирование конвективных течений. Разработанная программа применена для моделирования конвективных течений с использованием уравнений Навье-Стокса с внешней силой, зависящей от поля температуры:

$$\frac{\partial \mathbf{v}}{\partial t} = P \nabla^2 \mathbf{v} + \mathbf{v} \times (\nabla \times \mathbf{v}) + PR\theta \mathbf{e}_3 - \nabla p,$$

где R — число Рэлея, θ — отклонение температуры от базового линейного профиля $T_1 + (T_2 - T_1)z$, где T_1 и T_2 — температуры на нижней и верхней границах соответственно ($T_1 > T_2$).

Данная задача имеет известное решение, полученное из эксперимента, что позволяет провести валидацию программной реализации. Получены характерные конвективные структуры в виде валов, которые полностью соответствуют экспериментальным наблюдениям.

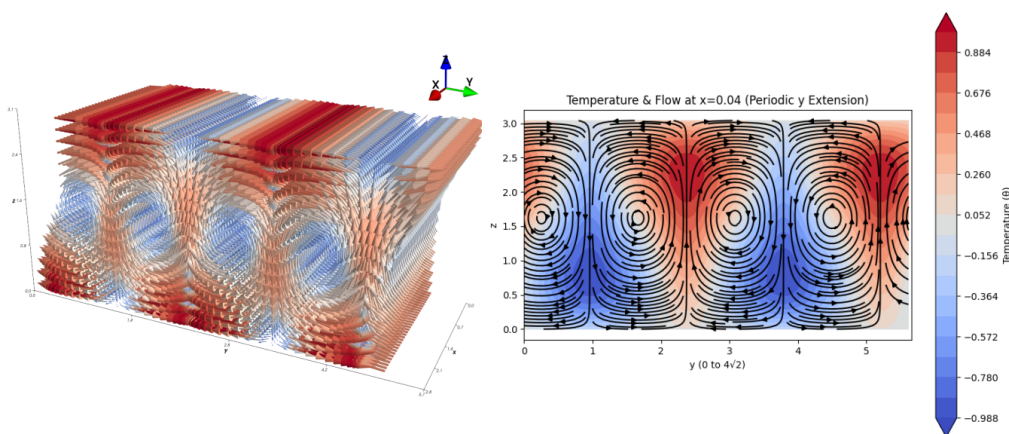


Рис. 1: Визуализация конвективных течений: слева — трехмерное поле скорости, справа — сечение поля скорости в плоскости yz . Конвективные валы отчетливо видны, их структура полностью согласуется с экспериментальными данными.

Корректность программной реализации численного метода также подтверждена экспериментом по исследованию устойчивости стационарного решения к малым возмущениям. При внесении случайных возмущений в поле скорости стационарного решения система эволюционирует обратно к стационарному состоянию с сохранением интегральной кинетической энергии с точностью 10^{-4} . Это демонстрирует правильность воспроизведения физики процесса и применимость метода для моделирования гидродинамических явлений.

Заключение. Реализован эффективный параллельный алгоритм решения уравнений Навье-Стокса на основе псевдоспектрального метода. Проведенная оптимизация позволила достичь высокой производительности на современных вычислительных системах. Полученные результаты могут быть использованы для решения широкого класса задач гидродинамики, требующих высокоточного моделирования течений вязкой несжимаемой жидкости.

ЛИТЕРАТУРА

- [1] Batchelor G.K. An Introduction to Fluid Dynamics. Cambridge University Press, 2000.
- [2] Chorin A.J. Numerical solution of the Navier-Stokes equations // Mathematics of Computation. 1968. Vol. 22, № 104.
- [3] Frigo M., Johnson S.G. The Design and Implementation of FFTW3 // Proceedings of the IEEE. 2005. Vol. 93, № 2.
- [4] Rabenseifner R., Hager G., Jost G. Hybrid MPI/OpenMP Parallel Programming on Clusters of Multi-Core SMP Nodes // 17th Euromicro International Conference on Parallel, Distributed and Network-based Processing. 2009.
- [5] Высокопроизводительные вычисления на ВМК МГУ. Описание вычислительного комплекса IBM Polus // hpc@cmc. URL: <http://hpc.cmc.msu.ru/polus> (дата обращения: 07.06.2025)

Применение анализа тональности текстов для прогнозирования движения финансового рынка

Дорофеев Александр Вадимович

Кафедра алгоритмических языков

e-mail: aldor2002@bk.ru

Научный руководитель — к.ф.-м.н. доц. Полякова Ирина Николаевна

В настоящее время всё больше растёт интерес участников рынка к использованию искусственного интеллекта в целях анализа настроений инвесторов и прогнозирования рыночных тенденций. Автоматизация интерпретации тональности текстов с использованием ресурсов вычислительной машины может существенно повысить качество и оперативность инвестиционных решений, особенно в условиях высокой неопределенности и изменчивости финансовых рынков. [1, 2]

Цель работы – разработка программных модулей на основе гибридного подхода анализа тональности для автоматизированного прогнозирования изменений на финансовом рынке.

В традиционных задачах оценки тональности в основном используют дискретные классы: положительный, нейтральный, отрицательный. В данной работе рассматривается непрерывная шкала тональности, представленная в виде вещественного числа в диапазоне $[-1; 1]$. Это позволяет учитывать интенсивность настроения в тексте: малые значения близкие к нулю отражают слабовыраженные эмоции, а значения, приближённые к границам диапазона, — ярко выраженные позитивные или негативные настроения.

При оценке тональности финансовых новостей критически важно опираться на текст статьи, а не только на заголовок, так как заголовки часто выполняют рекламно-манипулятивную функцию для привлечения внимания, но не отражают смысловых нюансов и контекста. Именно поэтому, в качестве основного датасета для практической части работы был выбран «Russian Financial News» – русскоязычный набор данных финансовых новостей для акций компаний, представленных на московской бирже. Этот набор данных универсального назначения (оценка тональности, суммаризация текста, извлечение ключевых слов) имеет средний размер в количестве ~ 15000 текстов новостных статей, тональность которых размечена автоматически с помощью ChatGPT-4o. [3]

В работе описаны три подхода к оценке тональности текстов: лексиконный метод, классические модели машинного обучения на основе линейной регрессии и модель глубокого обучения на основе трансформера RuBERT. [4, 5]

Лексиконный подход. Методы, базирующиеся на лексиконах, используют заранее подготовленные словари, в которых каждому слову сопоставлено значение тональности (отрицательное, нейтральное или положительное).

Для оценки общей тональности текста вычисляют агрегированную метрику по словам из словаря. В качестве словаря использовался «RuSentilex 2017».

Подход на основе модели машинного обучения. В текущей работе тональность представляется в виде вещественного числа, поэтому было принято решение использовать линейную регрессию. Предварительно производилась векторизация текста при помощи метода TF-IDF и рассматривались различные регуляризации: Ridge, Lasso, ElasticNet. Подбор оптимальных гиперпараметров выполнялся при помощи Grid Search. Лучший результат был получен на Ridge-регрессии.

Подход на основе модели глубокого обучения. Трансформер RuBERT позволяет учитывать контекстную зависимость слов и эффективнее моделировать тональность на уровне всего предложения. В данной работе используется дообученная (fine-tuned) модель RuBERT (DeepPavlov/rubert-base-cased) на задаче регрессии в диапазоне $[-1; 1]$. Но BERT имеет ограничение в 512 токенов (слов), в том числе специальных, а новостные тексты чаще всего превышают данное значение. Поэтому текст приходится разбивать на перекрывающиеся фрагменты длиной до 510 токенов с шагом 256. Такое перекрытие гарантирует, что каждый фрагмент содержит контекст предыдущего и следующего участков текста.

Гибридный подход. Гибридная модель – это объединение преимуществ трёх разнородных подходов к оценке тональности, рассмотренных ранее посредством техники стекинга с использованием градиентного бустинга CatBoost. Стекинг обладает следующими преимуществами, как:

- Дает возможность компенсировать слабые стороны одной модели силами другой.
- Обеспечивает более высокую точность по сравнению с одиночными моделями.

Для мета-модели в качестве признаков используются три скалярных предсказания каждой базовой модели:

1. x_{lex} – сумма положительных и отрицательных тональностей.
2. x_{reg} – предсказание Ridge регрессии с TF-IDF.
3. x_{bert} – регрессионный выход дообученного RuBERT.

Таким образом, каждому тексту ставится в соответствие вектор

$$x = (x_{lex}, x_{reg}, x_{bert}).$$

Для обучения мета-модели был выбран CatBoostRegressor – современный градиентный бустинг на деревьях решений. Гиперпараметры подбирались при помощи Grid Search. Для каждого из рассмотренных методов были получены значения метрик: MAE, RMSE, R2, они представлены в Табл. 1.

В ходе работы были реализованы программные модули:

- Модуль базы данных – предназначен для хранения всех промежуточных и итоговых данных в локальной системе управления базами данных (СУБД)

Метод	<i>MAE</i>	<i>RMSE</i>	R^2
Лексиконного подхода	0.5107	0.8195	-0.2845
Подход на основе машинного обучения	0.2925	0.3840	0.5800
Подход на основе глубокого обучения	0.2312	0.3017	0.7408
Гибридный подход	0.2041	0.2915	0.7581

Табл. 1: Сравнительная таблица.

для удобства повторного анализа и воспроизводимости. В качестве СУБД используется SQLite.

- Новостной трекер – используется для сбора фактов выхода новостей. Принцип его работы следующий: когда на информационном ресурсе в списке новостей появляется новая статья – модуль фиксирует этот факт, создавая запись.
- Парсер новостей – позволяет для каждой записи из основной таблицы базы данных получить полный текст новости по URL.
- Парсер котировок – предназначен для извлечения исторических цен акций с Московской биржи (МОЕХ) для последующего расчёта целевой переменной – процентное изменение цены акции.
- Модуль оценки тональности – производит оценку тональности текстов финансовых новостей на основе предложенного гибридного подхода.
- Модуль предсказания движения рынка – обеспечивает возможность прогнозирования краткосрочного изменения цены акции на основании тональности и исторических цен.

Выполненная работа была направлена на решение задачи автоматизированного прогнозирования движений цен на финансовом рынке на основе анализа тональности текстов новостей. Достигнутые значения метрик прогноза цен ($R^2 \approx 0.40$, $RMSE \approx 2.07\%$) показывают, что включение тонального анализа в модель прогнозирования движения финансового рынка даёт практическую ценность, и в то же время остаётся пространство для улучшений и дальнейших исследований.

ЛИТЕРАТУРА

- [1] Du K., Xing F., Mao R., Cambria E. «Financial Sentiment Analysis: Techniques and Applications» // ACM Comput. Surv. 2024. URL: <https://dl.acm.org/doi/full/10.1145/3649451>
- [2] Shimaа Ouf, Mona El Hawary, Amal Aboutabl and Sherif Adel, «A Deep Learning-Based LSTM for Stock Price Prediction Using Twitter Sentiment Analysis» // International Journal of Advanced Computer Science and Applications(ijacs), 2024. URL: <http://dx.doi.org/10.14569/IJACSA.2024.0151223>

-
- [3] Kasymkhan Khubiyev «Russian Financial News» // kaggle.com 2024. URL: <https://www.kaggle.com/datasets/kkhubiev/russian-financial-news>
 - [4] Ankur Sinha, Satishwar Kedas, Rishu Kumar, Pekka Malo «SEntFiN 1.0: Entity-Aware Sentiment Analysis for Financial News» // Journal of the Association for Information Science and Technology 2022. URL: <https://arxiv.org/abs/2305.12257>
 - [5] Justina Deveikyte, Helyette Geman, Carlo Piccari, Alessandro Provetti «A Sentiment Analysis Approach to the Prediction of Market Volatility» // arxiv.org 2020. URL: <https://arxiv.org/abs/2012.05906>

Применения матриц малого ранга для восстановления функции-источника в уравнении коагуляции-дробления

Работа удостоена диплома 2 степени

Закс Роберт Тамазович

Кафедра вычислительных технологий и моделирования

e-mail: zaks.robert@bk.ru

Научный руководитель — к.ф.-м.н. доц. Матвеев Сергей Александрович

Введение. Уравнения коагуляции и дробления вещества широко используются для фундаментальных исследований и инженерных приложений в разнообразных научных и производственных областях. Среди фундаментальных областей использования таких уравнений можно найти задачи астрофизики, роста кристаллических структур, образования ультрадисперсных частиц в атмосфере. Уравнения коагуляции также используются для описания процессов флокуляции в очистных сооружениях, динамики частиц сажи при пожарах, седиментации частиц в растворах и даже конденсации углерода при детонации взрывчатых веществ.

Процессы коагуляции и дробления в случае пространственно-однородных систем принято описывать при помощи уравнений типа Смолуховского [1]

$$\begin{aligned} \frac{\partial c(x, t)}{\partial t} = & \frac{1}{2} \int_0^x K(x-y, y) c(x-y, t) c(y, t) dy - c(x, t) \int_0^{+\infty} K(x, y) c(y, t) dy + \\ & + \int_x^{+\infty} \Psi(y, x) c(y, t) dy - \frac{c(x, t)}{x} \int_0^x y \Psi(x, y) dy + v(x, t), \quad x \geq 0, t > 0, \quad (1) \end{aligned}$$

где функция $c(x, t) \geq 0$ задает концентрацию частиц массы x на единицу объема в момент времени t . Известные из физики процесса функции $K(x, y) = K(y, x) \geq 0$ и $\Psi(x, y) \geq 0$, характеризующие частоту слияния и дробления, называются ядрами коагуляции и дробления, соответственно. Положительные значения функции $v(x, t)$, называемой функцией источника, соответствуют концентрации частиц массы x , прибывших в систему извне в момент времени t , а отрицательные значения соответствуют концентрации частиц массы x , покинувших систему в момент времени t .

В данной работе рассмотрена обратная задача по восстановлению функции источника $v(x, t)$ по известной концентрации частиц $c(x, t)$ в моменты времени $t = 0$ и $t = T$.

Метод решения данной задачи был впервые рассмотрен в статье [2]. Данная работа посвящена внедрению в этот метод решения обратной задачи алгоритмов на основе малоранговых представлений ядер коагуляции и дробления, позволяющих ускорить вычисления в тысячи раз без потери точности исходного алгоритма.

Идея использования малоранговых представлений ядер коагуляции раннее успешно применялась в прямых задачах для уравнения Смолуховского [3], однако, поскольку структура возникающих линейных операторов в алгоритме решения задачи о восстановлении функции источника близка к структуре исходного оператора Смолуховского, удалось применить описанный подход в рассматриваемом случае.

Краткое описание подхода. Мы будем предполагать, что функция источника $v(x, t)$ обращается в ноль при $t \notin (t_1; t_2) \subset (0; T)$ и записывать v в виде $m_c(t)v(x, t)$, где m_c — характеристическая функция интервала $(t_1; t_2)$. Кроме того, мы будем предполагать, что найдется такое $H \in (0; +\infty)$, что $c(x, 0) = 0$ при $x \notin [0; H]$; $K(x, y) = 0 = \Psi(x, y)$ при $(x, y) \notin [0; H]^2$ и $c(x, 0) \in \mathcal{L}_2[0; H]$, $K(x, y), \Psi(x, y) \in \mathcal{L}_2[0; H]^2$.

Введем следующие интегральные операторы

$$\begin{aligned} L_1[f, g](x) &= \int_0^x K(x-y, y)f(x-y)g(y)dy, \quad L_3[g](x) = \int_0^x \Psi(x, y)g(y)dy, \\ L_2[f, g](x) &= \int_x^H K(y-x, x)f(y-x)g(y)dy, \quad L_4[g](x) = \int_x^H \Psi(y, x)g(y)dy, \\ L_5[g](x) &= \int_0^H K(x, y)g(y)dy. \end{aligned}$$

Тогда уравнение (1) примет вид:

$$\begin{aligned} \frac{\partial c(x, t)}{\partial t} &= \frac{1}{2}L_1[c, c](x, t) - c(x, t)L_5[c](x, t) + L_4[c](x, t) - \\ &\quad - \frac{c(x, t)}{x}L_3[g \equiv y](x, t) + m_c v(x, t), \quad 0 \leq x \leq H, \quad 0 < t \leq T. \end{aligned} \quad (2)$$

Если рассматривать задачу (2) при малых T , то начальную концентрацию $c_0(x) \equiv c(x, 0)$ можно рассматривать в качестве приближения к $c(x, t)$. Поэтому положив $g(x, t) = c(x, t) - c(x, 0)$, $g_{obs}(x) = c(x, T) - c(x, 0)$ и, отбросив слагаемые второго порядка малости, положив, $X = (0; H) \times (0; T)$, получим

$$\begin{aligned} \frac{\partial g(x, t)}{\partial t} + A[g](x, t) &= m_c v(x, t) + s(x), \quad (x, t) \in X, \\ g(x, 0) &= 0, \quad g(x, T) = g_{obs}(x), \quad 0 \leq x \leq H, \end{aligned} \quad (3)$$

где оператор A и функция $s(x)$ могут быть выражены через операторы L_i и зависят только от функции $c_0(x)$.

Если найти функцию g_1 , удовлетворяющую задаче Коши

$$\frac{\partial g_1(x, t)}{\partial t} + A[g_1](x, t) = s(x), \quad (x, t) \in X, \quad g_1|_{t=0} = 0,$$

то для $h = g - g_1$, $h_{obs}(x) = g_{obs}(x) - g_1(x, T)$, обратная задача (3) сводится к нахождению функций $h, v \in \mathcal{L}_2(X)$ таких, что

$$\frac{\partial h(x, t)}{\partial t} + A[h](x, t) = m_c v(x, t), (x, t) \in X, \quad (4)$$

$$h(x, 0) = 0, h(x, T) = h_{obs}(x). \quad (5)$$

Наряду с (4), (5) рассмотрим следующую вариационную задачу: найти h, v такие, что

$$\begin{aligned} \frac{\partial h(x, t)}{\partial t} + A[h](x, t) &= m_c v(x, t), (x, t) \in X, h(x, 0) = 0, \\ \inf_{v \in \mathcal{L}_2} J_\alpha(v), \end{aligned} \quad (6)$$

где

$$J_\alpha(v) = J_\alpha(v, h(v)) = \alpha \|m_c v\|_{\mathcal{L}_2(X)}^2 + \|h(\cdot, T) - h_{obs}\|_{\mathcal{L}_2[0; H]}^2.$$

Здесь $\alpha = \text{const} \geq 0$. Отметим, что при $\alpha = 0$ вариационная задача (6) может рассматриваться в качестве обобщенной постановки задачи (4), (5), а минимизируемый функционал есть ни что иное как функционал Тихонова для оператора прямой задачи.

Используя теорию вариационного исчисления, избавимся от минимизации функционала, перейдя к следующей системе вариационных уравнений

$$\begin{aligned} \frac{\partial h(x, t)}{\partial t} + A[h](x, t) &= m_c v(x, t), (x, t) \in X, h(x, 0) = 0 \\ -\frac{\partial q(x, t)}{\partial t} + A^*[q](x, t) &= 0, (x, t) \in X, q|_{t=T} = h|_{t=T} - h_{obs} \\ \alpha m_c v(x, t) + m_c q(x, t) &= 0, (x, t) \in X. \end{aligned} \quad (7)$$

Здесь неизвестными являются $h(x, t)$, $q(x, t)$ и $v(x, t)$.

Отметим, что оператор A^* , также как и оператор A , может быть выражен через операторы L_i и зависит только от функции $c_0(x)$.

Аналогичным образом в работе рассматривается случай известной зависимости от времени для функции источника с разделяющимися переменными, т. е. $v(x, t) = v_x(x)v_t(t)$, где $v_x \in \mathcal{L}_2[0; H]$, $v_t \in \mathcal{L}_2[0; T]$ и функция v_t заранее известна. В этом случае использование предложенного ранее способа может оказаться нецелесообразным, поскольку ранее нигде явным образом v_t не задавалась и в общем случае, после решения обратной задачи, мы можем получить функцию источника с совсем другой зависимостью от времени.

Для решения (7) можно рассмотреть различные итерационные алгоритмы. Простейший градиентный метод для минимизации функционала $J_\alpha(v)$ имеет вид

$$\begin{aligned} \frac{\partial h^n}{\partial t} + A[h^n] &= m_c v^n, (x, t) \in X, h^n|_{t=0} = 0 \\ -\frac{\partial q^n}{\partial t} + A^*[q^n] &= 0, (x, t) \in X, q^n|_{t=T} = h^n|_{t=T} - h_{obs} \\ v^{n+1} &= v^n - \zeta (\alpha v^n + q^n), x \in [0; H], t \in [t_1; t_2], \end{aligned} \quad (8)$$

где $v^0 \in \mathcal{L}_2(X)$ — произвольное начальное значение и $\zeta = \frac{2}{2\alpha + C_0 + C_1}$ — итерационный параметр, в котором можно положить $C_0 = 0$ и $C_1 = T$.

Для численной реализации (8) выпишем равномерные сетки на отрезках $[0; H]$ и $[0; T]$ с шагом $h = \frac{H}{N}$ и $\tau = \frac{T}{M}$ соответственно, аппроксимируем интегральные операторы L_i с помощью квадратурной формулы трапеций, а потом запишем явную разностную схему Эйлера.

Обозначим через M_K матрицу с элементами $K(x_i, x_j)$, а через M_Ψ матрицу с элементами $\Psi(x_i, x_j)$, которую доопределили при $i < j$ для лучшего малорангового приближения. Тогда если эти матрицы аппроксимировать матрицами ранга R_K и R_Ψ , то за счет быстрого дискретного преобразования Фурье и быстрого умножения малоранговых матриц и их верхних и нижних треугольных частей на вектор, можно снизить сложность вычисления результата применения L_i (а значит и A и A^*) на сетке с $O(N^2)$ до $O(R_K N \log N + (R_K + R_\Psi)N)$.

В качестве критерия останова полученного алгоритма, помимо останова по числу итераций, был рассмотрен останов по относительной ошибке минимизируемого функционала J_α на полученных приближениях.

Описание полученных результатов. Проверку правильности решения поставленной задачи осуществим следующим образом: зададим значение функции $v = v_{sol}$ и решим прямую задачу для исходного уравнения (1), найдя значение концентрации в момент времени $t = T$, с которым будем решать обратную задачу с помощью полученной численной схемы, используя начальное приближение $v = v_0$. Полученный результат $v = v_{res}$ сравним с исходным v_{sol} на отрезке $[t_1; t_2]$.

Было проверено, что при уменьшении параметра α уменьшается минимальное значение функционала и точность приближения увеличивается, однако при этом число итераций, требуемое для достижения минимума, возрастает. Вопрос о выборе параметра α требует дополнительного исследования. В прикладных задачах его стоит выбирать достаточно малым для обеспечения требуемой точности приближений на основе методов, предлагаемых в теории обратных задач.

Отметим также, что уменьшение шага дискретизации h позволяет улучшать поведение функционала. Так, например, при достаточно большом α и недостаточно малом h функционал может начать расти из-за большой погрешности аппроксимаций и за счет уменьшения h возможно добиться убывания функционала в соответствии с теорией градиентных методов минимизации. Именно по этой причине во всех примерах мы будем брать достаточно большое $N = 2048$ и именно тут проявилась важность использования ускорять применения интегральных операторов (отметим, что даже при $N = 80$ ускорение существенное).

В дальнейшем мы будем предполагать, что $H = 6.0$, $M = 80$.

В случае $K = 1$, $\Psi = 0$, $v_{sol}(x, t) = e^{-x}$, $v_0(x, t) = e^{-2x}$, $\alpha = 0.001$, $T = 0.5$, $t_1 = 0.1$, $t_2 = 0.3$, при останове на 100 итерации результат хорошо аппроксимирует искомую функцию источника в центре отрезка $[t_1; t_2]$, а на его концах наблюдаются сильные расхождения, что иллюстрирует рис. 1.

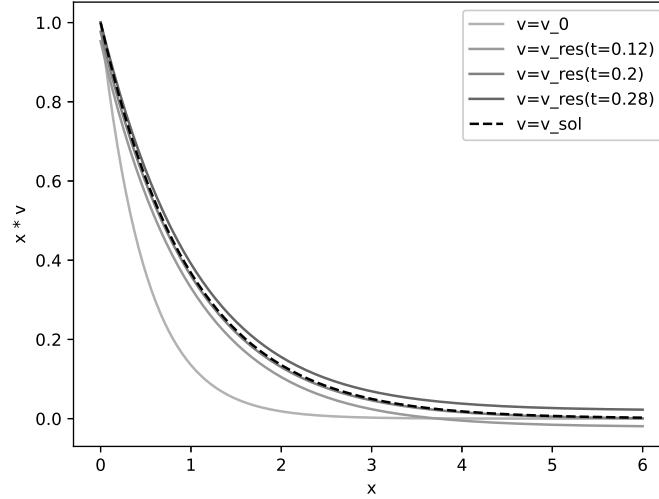


Рис. 1: Функции источника при использовании останова на 100 итерации.

В случае более сложных ядер $K(x, y) = \left(x^{\frac{1}{3}} + y^{\frac{1}{3}}\right)^2 \sqrt{\frac{1}{x} + \frac{1}{y}}$, $\Psi = e^{-3x}$ и функций источника $v_{sol}(x, t) = e^{-\frac{(x-1.5)^2}{2}} + 2e^{-\frac{(x-4.5)^2}{2}}$, $v_0(x, t) = e^{-\frac{(x-3.0)^2}{4}}$, $\alpha = 0.0001$, $T = 0.05$, $t_1 = 0.01$, $t_2 = 0.03$ и использования останова по относительной ошибке функционала наблюдается аналогичное поведение, что иллюстрирует рис. 2.

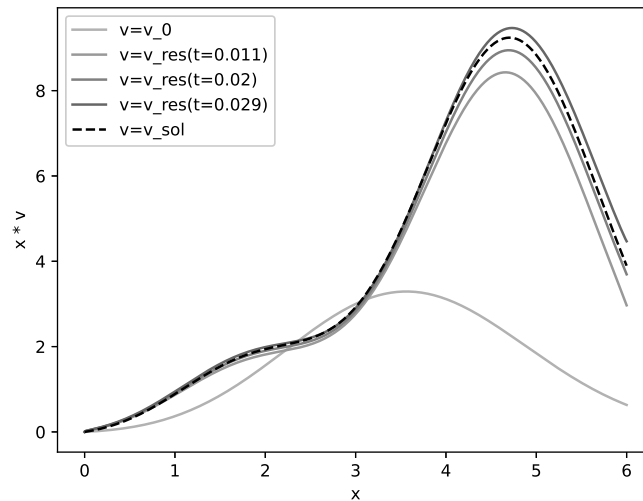


Рис. 2: Функции источника в терминах масс при использовании останова по функционалу.

Несмотря на это, найденная аппроксимация функции источника достаточно хороша в ином смысле: значение $c(x, T)$, найденное из прямой задачи для исходного нелинейного уравнения не сильно отличается при использовании функций источника v_{sol} и v_{res} , что иллюстрируется на рис. 3.

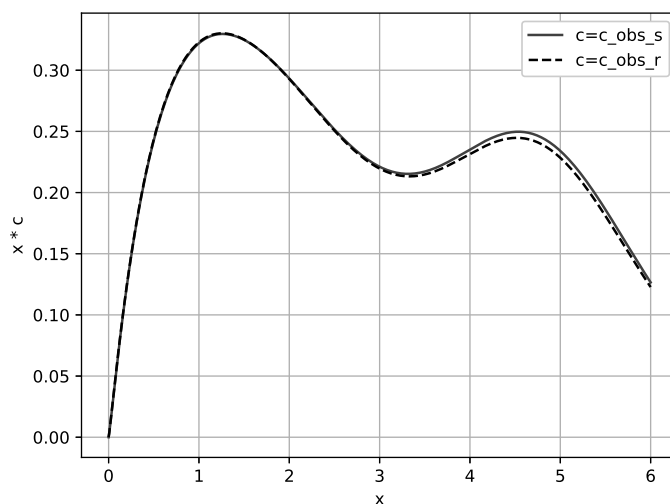


Рис. 3: Концентрации в терминах масс.

Дальнейшие направления для исследования. В дальнейшем хотелось бы исследовать применение мозаично-скелетонных приближений вместо малоранговых. Кроме того, отдельного исследования требуют вопросы выбора параметров итерационного алгоритма, а также вопросы согласования шага дискретизации h с параметром α .

ЛИТЕРАТУРА

- [1] Алоян А. Е. Динамика и кинетика газовых примесей и аэрозолей в атмосфере. М. : ИВМ РАН, 2002.
- [2] Agoshkov V. I., Dubovski P. B. Solution of the reconstruction problem of a source function in the coagulation-fragmentation equation // Russ. J. Numer. Anal. Math. Modelling. 2002. Vol. 17, N 4. P. 319–330.
- [3] Matveev S. A., Smirnov A. P., Tyrtysnikov E. E. A fast numerical method for the Cauchy problem for the Smoluchowski equation // Journal of Computational Physics. 2015. Vol. 282, P. 23–32.

Стохастические методы восстановления пакетов, потерянных в сетях передачи данных

Иванов Егор Романович

Кафедра математических методов прогнозирования

e-mail: ivanover@my.msu.ru

Научный руководитель — к.ф.-м.н. доц. Гуров Сергей Исаевич

Своей задачей теория кодирования ставит разработку методов передачи информации, устойчивых к неисправностям канала связи. В конце прошлого века стали активно развиваться технологии, способные передавать большое количество информации, например, телевидение и интернет. Кроме того, их отличительной чертой стала не слишком высокая цена ошибки декодирования. Именно эти две особенности процесса передачи информации оказались решающими в вопросе применимости классических алгебраических кодов. Эффективные алгоритмы их декодирования работают достаточно быстро для не слишком длинных битовых последовательностей, что можно считать платой за обеспечиваемый ими результат, а именно полное восстановление утраченной информации.

Для решения многих задач в новых условиях было бы достаточно, чтобы ошибки декодирования происходили «не слишком часто» (например, утеря или искажение нескольких кадров видеоряда), а размеры передаваемых битовых последовательностей были достаточно большими. В качестве решения в начале текущего столетия были предложены фонтанные коды [1, 2]. В таком случае источник работает как разбрызгивающий пакеты цифровой фонтан (Digital Fountain), а для декодирования получателю достаточно лишь собрать в «ведро» определенное количество любых «капель». В данной работе изучена возможность применения идей фонтанных кодов в задачах меньшей размерности.

Теория фонтанного кодирования частично наследует терминологию классической, при этом используя несколько характерных для себя понятий.

Опишем общую постановку задачи. Имеется информация, представленная в виде k битовых векторов $x_1, \dots, x_k$ одинаковой размерности, которые называются *пакетами*. Пакет является неделимой единицей информации. Иными словами, отсутствует возможность обратиться к его битовой структуре. Над пакетами определена операция $\oplus$ суммы по модулю 2 как соответствующая покомпонентная.

Требуется передать все пакеты через стирающий канал связи. В данной работе использовалась модель БЕС (Binary Erasure Channel). В таком случае канал не пропускает всякую сущность, через себя проходящую, независимо от остальных с вероятностью p , часто называемой PLR (Packet Loss Rate). Атомарность битового пакета для канала связи означает, что он либо передается целиком без искажений, либо не доходит вовсе.

Для уменьшения вероятности p безвозвратной утери пакета через канал связи информация пересылается в виде n битовых векторов $s_1, \dots, s_n$, называемых *кодирующими символами*.

В классической теории кодирования алгоритм построения набора кодирующих символов является детерминированным при наперед заданных значениях k и n . В фонтанном кодировании процедура построения кода имеет принципиально стохастическую природу. Каждый новый кодирующий символ генерируется следующим образом. Имеется некоторое фиксированное распределение $\boldsymbol{\rho}$. Его носитель — натуральные числа $\{1, \dots, k\}$. отождествим данное распределение с вектором $\boldsymbol{\rho}$, компоненты которого суть соответствующие вероятности:

$$\boldsymbol{\rho} = (\rho_1, \dots, \rho_k)^\top, \quad \boldsymbol{\rho} \geq 0, \quad \sum_{i=1}^k \rho_i = 1$$

Из этого распределения реализуется натуральное число d , называемое *степенью* нового кодирующего символа. После этого строится выборка без возвращения из пакетов $x_1, \dots, x_k$ размера d . Новый кодирующий символ определяется как сумма по модулю 2 выбранных пакетов.

Постановка задачи. Требуется, используя идеи фонтанного кодирования, построить код, снижающий уровень ошибок в канале.

Предположения:

- процент потерь пакетов (PLR) p в канале достаточно мал ($p \leq 7\%$);
- количество k посылаемых пакетов невелико ($k \leq 100$).

Ограничения:

- Код — систематический: $s_i = x_i$, $i = 1, \dots, k$. Аналогично теории алгебраического кодирования, символы $s_1, \dots, s_k$ будем называть *информационными*, а $s_{k+1}, \dots, s_n$ — *проверочными*.
- Максимальная допустимая избыточность — 20%: $m/k \leq 0,2$.

Требование:

- Уменьшение вероятности безвозвратной утери π хотя бы в 10 раз относительно собственного PLR канала: $\pi \leq p/10$.

Подходы к решению. Имеющимися степенями свободы являются алгоритм декодирования и распределение, по которому посылаются избыточные кодирующие символы. Были выделены две возможные стратегии действий:

- использовать сложный алгоритм декодирования (например, ЛТ-процесс [1]) и малопараметрическое распределение $\boldsymbol{\rho}$;
- использовать простой алгоритм декодирования (допускающий аналитический вывод формул) и решать задачу условной оптимизации.

Стратегия малопараметрического распределения и сложного алгоритма декодирования. Выбор возможных распределений степеней кодирующих символов осуществлялся, исходя из следующего наблюдения. Удачные распределение идейно выглядят следующим образом: чаще всего генерируются символы одной и той же степени, которые постепенно расходятся для поддержания течения процесса декодирования. Перечислим выбранные распределения:

- биномиальное с параметрами k и p_d ;
- пуассоновское с параметром λ ;
- степень d' с вероятностью 1.

Численное моделирование. Результаты экспериментов приведены в таблицах 1, 2. Запись b в таблице означает, что лучшим оказалось биномиальное распределение, d — вырожденное.

k	m	Метод	$\hat{\pi}$	m	Метод	$\hat{\pi}$
10	1	d	0,0203	2	b	0,0123
20	2	d	0,0205	4	d	0,0083
30	3	d	0,0190	6	d	0,0058
40	4	d	0,0175	8	d	0,0041
50	5	d	0,0158	10	d	0,0030
60	6	d	0,0148	12	d	0,0024

Табл. 1: $p = 0,05$.

k	m	Метод	$\hat{\pi}$	m	Метод	$\hat{\pi}$
10	1	d	0,0357	2	b	0,0221
20	2	d	0,0365	4	d	0,0173
30	3	d	0,0342	6	d	0,0132
40	4	d	0,0334	8	d	0,0100
50	5	d	0,0321	10	d	0,0087
60	6	d	0,0307	12	d	0,0069

Табл. 2: $p = 0,07$.

Стратегия простого алгоритма и оптимизируемого распределения.

Определение 1. Будем называть алгоритм декодирования простым, если он работает без памяти, то есть не хранит никаких проверочных символов, кроме обрабатываемого на текущем шаге.

Идея подхода заключается в следующем:

- Получить аналитическое выражение для вероятности π утери пакета в канале связи, улучшенного с помощью простого алгоритма декодирования.
- Полученное выражение — функция $\pi(\boldsymbol{\rho})$, оптимизация которой представляет интерес для решения задачи:

$$\begin{cases} \pi(\boldsymbol{\rho}) \rightarrow \min \\ \sum_{i=1}^k \rho_i = 1, \boldsymbol{\rho} \geq 0 \end{cases} \quad (1)$$

- Решить задачу оптимизации (1) аналитически или численно.

Метод декодирования 1. Опишем первый из предложенных методов декодирования:

- информационные пакеты декодер «просто получает»;
- при обработке проверочных декодирование проводится, если множество исходных пакетов, вошедших в сумму по модулю 2 для формирования полученного имеет следующую структуру: все, кроме одного — пришедшие ранее информационные пакеты.

Для описанного метода декодирования возможно выписать аналитическое выражение для вероятности безвозвратной утери пакета в обновленном канале связи.

Теорема 1. Для метода 1 верно, что

$$\pi(\boldsymbol{\rho}) = p \sum_{I=0}^{k-1} \binom{k-1}{I} p^{k-1-I} (1-p)^I \left[p + (1-p) \left(1 - \sum_{d=1}^{I+1} \rho_d \frac{\binom{I}{d-1}}{\binom{k}{d}} \right) \right]^m$$

Метод декодирования 2. Улучшим метод 1: «разрешим» алгоритму декодирования использовать все ранее восстановленные пакеты, а не только дошедшие информационные. Иными словами, при обработке проверочных символов декодирование проводится, если множество исходных пакетов, вошедших в сумму по модулю 2 для формирования полученного, имеет следующую структуру: все, кроме одного — *восстановленные* ранее пакеты.

Для описания метода 2 введем случайные величины r_i , $i = 0, \dots, m$, равные числу восстановленных пакетов после получения проверочного символа с номером i . Так, r_0 имеет биномиальное распределение с параметрами k и $1-p$. Случайные величины $r_0, \dots, r_m$ по построению метода декодирования образуют марковскую цепь. Опишем ее матрицу переходов $\mathbf{P}$, элемент p_{ij} которой суть вероятность перехода из состояния j в состояние i .

Теорема 2. Матрица $\mathbf{P}$ бидиагональная, причем для ее элементов верны следующие выражения:

$$p_{i+1,i} = (1-p) \cdot \sum_{d=1}^{i+1} \rho_d \frac{\binom{i}{d-1}(k-i)}{\binom{k}{d}},$$

$$p_{i,i} = p + (1-p) \cdot \left[1 - \sum_{d=1}^{i+1} \rho_d \frac{\binom{i}{d-1}(k-i)}{\binom{k}{d}} \right]$$

Дополнительно введем вектор $\mathbf{s} = (s_i)$ – вектор вероятностей оказаться на нулевом шаге декодирования (сразу после отправки всех информационных пакетов) в каждом из $k+1$ состояний. Это вектор вероятностей биномиального закона с числом испытаний k и вероятностью успеха $1-p$:

$$s_i = \mathbf{P}\{r_0 = i\} = \binom{k}{i} (1-p)^i p^{k-i}, \quad i = 0, \dots, k$$

Также за $\mathbf{b} = (b_i)$ обозначим вектор, определяемый следующим свойством: $b_i = i$.

В этих терминах выражение для математического ожидания числа восстановленных пакетов r_m принимает следующий вид:

$$\mathbf{E} r_m = \langle \mathbf{P}^m(\boldsymbol{\rho}) \mathbf{s}, \mathbf{b} \rangle$$

В силу симметрии относительно перенумерации пакетов, верно, что

$$\pi(\boldsymbol{\rho}) = 1 - \frac{1}{k} \mathbf{E} r_m = 1 - \frac{1}{k} \langle \mathbf{P}^m(\boldsymbol{\rho}) \mathbf{s}, \mathbf{b} \rangle$$

Метод решения задачи оптимизации. Для решения задачи оптимизации применялся метод Брэгмана. Это проксимальный метод оптимизации, в котором для решения задач на вероятностном симплексе в качестве расстояния Брэгмана используется обобщенная дивергенция Кульбака–Лейблера:

$$D(\boldsymbol{\rho}', \boldsymbol{\rho}'') = \sum_{i=1}^k \left[\rho'_i \log \frac{\rho'_i}{\rho''_i} - \rho'_i + \rho''_i \right]$$

Получаемая итерационная схема имеет теоретическую скорость сходимости $O(1/k)$ для L -гладких выпуклых функций при $0 < \alpha < 1/L$.

Разреженность решений. В случае если начальное приближение $\boldsymbol{\rho}_0$ для проксимального метода Брэгмана является внутренней точкой вероятностного симплекса:

$$\rho_i > 0, \quad i = 1, \dots, n$$

то все точки порождаемой последовательности приближений также будут внутренними. Однако на практике больший интерес представляют разреженные распределения в силу простоты реализации соответствующей случайной величины. Классическим разреживающим регуляризатором является ℓ_1 -норма вектора.

Этот подход оказывается неприменимым в данной задаче в силу равенства регуляризатора тождественной константе на вероятностном симплексе,

поэтому в качестве регуляризатора использовалась ℓ_0 -норма вектора, равная числу его ненулевых компонент. Таким образом, была поставлена следующая задача оптимизации:

$$\begin{cases} \pi(\boldsymbol{\rho}) + \lambda \|\boldsymbol{\rho}\|_0 \rightarrow \min \\ \sum_{i=1}^k \rho_i = 1, \boldsymbol{\rho} \geq 0 \end{cases} \quad (2)$$

λ – коэффициент регуляризации, гиперпараметр процедуры. Для решения задачи (2) использовался прореживающий алгоритм, предложенный в [3].

Условия и итоги эксперимента. Для обоих описанных методов декодирования запускался проксимальный метод Брэгмана. Полученное решение использовалось в качестве начального приближения для процедуры прореживания. В обеих процедурах использовалось значение $\alpha = k$ и $\lambda \propto p/k$. Далее полученные распределения $\boldsymbol{\rho}$ использовались для оценки $\hat{\pi}$ при использовании ЛТ-процесса в качестве алгоритма декодирования.

- для избыточности в 20% удалось получить следующие значения для уменьшения вероятности безвозвратной утери пакета:
 - $p = 0,05$: в 5 раз для $k \geq 20$ и в 10 раз для $k \geq 50$;
 - $p = 0,07$: в 5 раз для $k \geq 40$ и в 10 раз для $k \geq 70$.
- для избыточности в 10% не удалось добиться требуемого улучшения.

Результаты исследования.

- Для решения поставленной задачи были предложены две группы подходов. Одна из них является набором переборных алгоритмов, которые позволили глубже изучить структуру процесса декодирования. Вторая группа подходов представляет собой процедуру получения «удачных» для сложного алгоритма декодирования распределений как решения задачи оптимизации для простого, допускающего аналитическое описание.
- Благодаря построенным алгоритмам, удалось решить поставленную задачу и уменьшить вероятность безвозвратной утери в канале на порядок:
 - для $k \geq 50$ при собственном PLR $p = 0,05$;
 - для $k \geq 70$ при собственном PLR $p = 0,07$.

ЛИТЕРАТУРА

- [1] Luby M. LT codes // The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings, 2002, P. 19.
- [2] MacKay D. J. C. Fountain codes // IEEE Proceedings – Communications, 2005, 152, P. 1062–1068.
- [3] Pan J., Yan M. Efficient sparse probability measures recovery via Bregman gradient // J. Sci. Comput., 2025. Vol.102, N 3.

Модификации компактной версии криптосистемы Нидеррайтера

Кириюткина Арина Владимировна

Кафедра информационной безопасности

e-mail: arisha.kiryutkina@mail.ru

Научный руководитель — к.ф.-м.н. доц. Чижев Иван Владимирович

Актуальность исследования. Развитие квантовых вычислений представляет серьёзную угрозу для современных криптографических систем. Квантовые компьютеры, использующие принципы суперпозиции и запутанности, способны эффективно решать задачи, лежащие в основе многих асимметричных криптосистем, таких как факторизация больших чисел (алгоритм Шора) и дискретное логарифмирование. Это ставит под угрозу безопасность широко применяемых алгоритмов, включая RSA и ECC. В связи с этим активно развивается постквантовая криптография, направленная на создание алгоритмов, устойчивых к квантовым атакам [1, 2].

Одним из перспективных направлений являются криптосистемы на основе кодов, исправляющих ошибки, в частности, схема Нидеррайтера. Её стойкость основана на NP -трудной задаче декодирования линейных кодов, что делает её устойчивой даже к квантовым атакам. Однако классическая версия имеет ряд недостатков, таких как большая длина ключей. Это обуславливает необходимость разработки оптимизированных модификаций, сохраняющих криптографическую стойкость при повышении производительности.

Цель работы — разработка модифицированной версии криптосистемы Нидеррайтера, обладающей повышенной эффективностью при сохранении криптографической стойкости.

Определение. *Криптосистемой Нидеррайтера называется криптосистема с открытым ключом, основанная на задаче декодирования линейных кодов, где:*

- *Открытый ключ: проверочная матрица $H' = SHP$, где H — проверочная матрица кода, S — случайная невырожденная матрица, P — матрица перестановки*
- *Закрытый ключ: тройка (S, H, P)*
- *Шифрование: преобразование сообщения в вектор ошибки e веса t и вычисление синдрома $s = H'e^T$*
- *Дешифрование: использование закрытого ключа для декодирования синдрома*

Анализ существующих методов и алгоритмов. Проведённый анализ современных модификаций криптосистемы Нидеррайтера выявил следующие ключевые направления оптимизации:

1. Использование квази-циклических кодов снижает размер ключа за счет циклической структуры

2. Применение кодов Гоппы повышает стойкость к структурным атакам и эффективность синдромного декодирования

Уязвимость компактной версии криптосистемы Нидеррайтера. Была предложена компактная версия криптосистемы Нидеррайтера [3], существенно сокращающая длину ключа за счет введения матрицы H_{cyclic} :

$$H_{cyclic} = [P_{cyclic} | I_{(n-k) \times (n-k)}],$$

которая передавалась в качестве открытого ключа (P_{cyclic} - циклическая матрица). В качестве закрытого ключа также использовались матрицы S, H, P. В процессе шифрования исходное сообщение преобразовывалось в вектор ошибок e веса t :

$$e = [0_{1 \times k} | e_i]_{1 \times n}$$

и умножалось на матрицу из открытого ключа. Расшифрование строилось аналогично расшифрованию в криптосистеме Нидеррайтера. Однако при исследовании криптосистемы была выявлена уязвимость, отражённая в

Теорема 1. *В компактной версии криптосистемы Нидеррайтера функция шифрования является тождественной, что позволяет восстановить сообщение непосредственно из шифртекста.*

Доказательство.

1. Вычисление:

$$H'^T = P_{per}^T H^T S_{ns}^T = \begin{bmatrix} P \\ I \end{bmatrix}_{n \times (n-k)}$$

$$H_{cyclic}^T = \begin{bmatrix} P_{cyclic} \\ I \end{bmatrix} = \begin{bmatrix} P \\ I \end{bmatrix} + \begin{bmatrix} P_{sec} \\ 0 \end{bmatrix} = H'^T + H_{secondary}^T$$

2. Формирование вектора ошибок e :

$$e = [0_{1 \times k} | e_i]_{1 \times n}$$

3. Шифрование:

$$c = e \cdot H_{cyclic}^T = [0 | e_i] \cdot \begin{bmatrix} P \\ I \end{bmatrix} + [0 | e_i] \cdot \begin{bmatrix} P_{sec} \\ 0 \end{bmatrix} = [0 | e_i] \cdot \begin{bmatrix} P \\ I \end{bmatrix} = e$$

4. Результат:

$$c = e$$

Исходное сообщение не скрыто

□

Модификация компактной версии криптосистемы Нидеррайтера. В связи с найденной уязвимостью в компактной версии, была предложена модификация данной криптосистемы, сохраняющая стойкость криптосистемы Нидеррайтера и не обладающая приведенными недостатками, существенные отличия которой от предыдущей версии можно сформулировать так:

1. Двойная циклическая структура:

- Использование двух независимых циклических матриц P_1 и P_2 размера $(n - k) \times k$ и $(n - k) \times (n - k)$ соответственно.
- Построение модифицированной проверочной матрицы:

$$H_{cyclic} = [P_1 | P_2]_{(n-k) \times n}$$

- Добавление параметра s , определяющего количество ненулевых битов в первой части вектора ошибки.

2. Усовершенствованная процедура шифрования:

- Разделение вектора ошибки на две части: $\underbrace{e_1}_{1 \times k} | e_2$
- Контроль весов: $wt(e_1) = s$, $wt(e_2) = t - s$
- Формула шифрования:

$$c = e \cdot H_{cyclic}^T$$

3. Итеративное расшифрование:

- Вычисление матрицы P_{sec} , как верхней подматрицы $H_{secondary}^T$ размера $k \times (n - k)$, где

$$H_{secondary}^T = H_{cyclic}^T + H'^T, H'^T = P_{per}^T H^T S_{ns}^T$$

- Подбор вектора e_1 длины k и веса s .
- Выполнение вычислений, после которых к полученному вектору применяется алгоритм расшифрования классической версии криптосистемы Нидеррайтера

Стойкость модификации. Рассматриваются матрицы H' , $H_{secondary}$, участвующие в формировании ключа, имеющие такое представление:

$$H' = P_{per}^T H^T S_{ns}^T = F S_{ns}^T,$$

$$F = \begin{bmatrix} A \\ B \end{bmatrix}_{n \times (n-k)}, S_{ns}^T = B^{-1} \cdot P_2$$

$$H'^T = F \cdot S_{ns}^T = \begin{bmatrix} A \cdot S_{ns}^T \\ B \cdot S_{ns}^T \end{bmatrix} = \begin{bmatrix} A \cdot S_{ns}^T \\ B \cdot B^{-1} \cdot P_2 \end{bmatrix} = \begin{bmatrix} P \\ P_2 \end{bmatrix}_{n \times (n-k)}$$

$$H_{secondary}^T = H_{cyclic}^T + H'^T = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} + \begin{bmatrix} P \\ P_2 \end{bmatrix} = \begin{bmatrix} P_{sec} \\ 0 \end{bmatrix}_{n \times (n-k)}$$

Вычисление шифротекста строится таким образом:

$$c = e \cdot H_{cyclic}^T = e \cdot H'^T + e \cdot H_{secondary}^T = \underbrace{[e_1|e_2]}_{1 \times n} \cdot \begin{bmatrix} P \\ P_2 \end{bmatrix} + \underbrace{[e_1]}_{1 \times k} | e_2 \cdot \begin{bmatrix} P_{sec} \\ 0 \end{bmatrix}$$

Восстановление вектора ошибки e по шифротексту c требует решения задачи синдромного декодирования, что является NP-трудной задачей для случайных линейных кодов [1].

Получение исходного сообщения, если известен закрытый ключ:

При нахождении корректного вектора e_1 , вычисляется матрица P_{sec} из уравнения $P_{sec} = P + P_1$ (матрицы имеют размерность $k \times (n - k)$), где P_1 – компонента открытого ключа, а P находится из

$$\begin{cases} H'^T = P_{per}^T H^T S_{ns}^T, \\ H'^T = \begin{bmatrix} P \\ P_2 \end{bmatrix} \end{cases}$$

$$c' = e \cdot H_{secondary}^T = [e_1 \cdot P_{sec} | 0]_{1 \times (n-k)}$$

$$\begin{aligned} c_1 = c - c' &= e \cdot H_{cyclic}^T - c' = e \cdot H'^T + e \cdot H_{secondary}^T - e \cdot H_{secondary}^T = \\ &= e \cdot H'^T = e \cdot P_{per}^T \cdot H^T \cdot S_{ns}^T \end{aligned}$$

Далее применяем оригинальный алгоритм расшифрования, используемый в криптосистеме Нидеррайтера. Таким образом, мы получим корректное сообщение m .

Следовательно, для модифицированной криптосистемы Нидеррайтера выполняются:

1. *Корректность*: При наличии закрытого ключа $sk = (S, H, P)$ для любого вектора ошибок $e \in \mathbb{F}_2^n$ веса $\leq t$ выполняется:

$$\text{Decrypt}(sk, \text{Encrypt}(pk, e)) = e,$$

где Encrypt и Decrypt – шифрование и расшифрование соответственно.

2. *Стойкость*: Без знания sk задача восстановления e по $c = e \cdot H_{cyclic}^T$ является вычислительно неосуществимой при условии NP-трудности задачи синдромного декодирования.

Таким образом, предложенная модификация криптосистемы не имеет уязвимости, присутствующей в её компактной версии.

Сложности реализации алгоритмов генерации ключа и шифрования у данной модификации и классической криптосистемой Нидеррайтера практически совпадают, основное отличие наблюдается в операции расшифрования, которая сложнее для модифицированной версии из-за подбора вектора e_1 .

Выводы по стойкости модифицированной схемы. Рассмотрев атаки на данную модификацию и сравнив сложность их реализации с атаками на классическую криптосистему Нидеррайтера [4], можно сделать следующие выводы:

- *ISD-атаки:* Сложность декодирования возрастает до $O(2^{(n/20)+k})$ из-за невозможности прямого применения оптимизаций BJMM/Stern
- *Для атак на параметры:* Необходим перебор 2^k вариантов вектора $e_1 \cdot P_{\text{sec}}$, что делает атаки типа Сидельникова-Шестакова неэффективными при $k \geq 80$
- *Для матричных атак:* Дополнительная случайная матрица P_{sec} разрушает структуру, используемую в методах Overbeck
- *Критический параметр:* Вес s вектора e_1 должен быть ≥ 3 для предотвращения статистических атак

Таким образом, к сложности реализации рассмотренных атак добавляется дополнительный шаг - подбор вектора $e_1 \cdot P_{\text{sec}}$, который потребует 2^k операций.

Заключение Таким образом, была разработана и исследована модификация компактной версии криптосистемы Нидеррайтера. Основной целью работы было сохранение эффективности и безопасности классической схемы, оптимизация длины открытого ключа и устранение уязвимостей, характерных для компактной версии криптосистемы Нидеррайтера.

Полученные результаты вносят существенный вклад в развитие постквантовой криптографии и открывают новые перспективы для создания устойчивых криптографических систем в условиях развития квантовых вычислений. Практическая значимость работы подтверждается возможностью использования разработанных решений в системах защиты информации нового поколения.

ЛИТЕРАТУРА

- [1] Niederreiter H. Knapsack-type cryptosystems and algebraic coding theory // Problems of Control and Information Theory. — 1986. — Vol. 15, N 2. — P. 157–166.
- [2] Overbeck R., Sendrier N. Code-based cryptography // Post-Quantum Cryptography. — Springer, 2009. — P. 95–142.
- [3] A Tiny Public Key Scheme Based on Niederreiter Cryptosystem / A. Khalvan, A. Zali, M. Ahmadian Attari // arXiv preprint. — 2023. — arXiv:2310.06724. — Department of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran.
- [4] Decoding Random Binary Linear Codes in $2^{n/20}$: How $1 + 1 = 0$ Improves Information Set Decoding / A. Becker et al. // Journal of Cryptology. — 2012. — Vol. 25, N 2. — P. 175–193.

Методы дистилляции больших языковых моделей и их практические применения

Работа удостоена диплома 3 степени

Ковалев Григорий Павлович

Кафедра алгоритмических языков

e-mail: kaengreg@ya.ru

Научный руководитель — к.ф.-м.н. Тихомиров Михаил Михайлович

Введение. Современные достижения в области обработки естественного языка (NLP) во многом обусловлены развитием крупных языковых моделей (LLM), таких как GPT [3] и BERT [4]. Эти модели, основанные на архитектуре трансформеров [2] и обученные на масштабных корпусах данных, таких как Common Crawl и Википедия, демонстрируют выдающиеся результаты в задачах генерации текста, машинного перевода, ответов на вопросы и информационного поиска.

Однако, несмотря на их эффективность, LLM требуют значительных вычислительных ресурсов, что ограничивает их применение на мобильных, встро-енных и других устройствах с ограниченными вычислительными возможностями. Размер таких моделей часто достигает десятков и сотен миллиардов параметров, что существенно снижает их практическую применимость в задачах, где критична оптимизация вычислительных затрат.

Одним из подходов к решению этой проблемы является дистилляция моделей, представляющая собой процесс переноса знаний от крупной модели-учителя к более компактной модели-ученику. В ходе этого процесса ученик обучается не только на исходных данных, но и на вероятностных выходах (мягких метках) учителя, которые содержат более богатую информацию по сравнению с детерминированными метками (жесткими метками). Это позволяет ученику достичь лучшей обобщающей способности и устойчивости при меньших вычислительных затратах.

Таким образом, дистиллированные модели обеспечивают сочетание высокого качества с эффективностью, что делает их пригодными для внедрения на устройствах с ограниченными ресурсами, таких как смартфоны и IoT-устройства.

В данной работе рассматривается задача уменьшения больших языковых моделей посредством усовершенствованной схемы дистилляции, сочетающей итеративную, основанную на данных оценку важности слоев с последующим восстановлением качества. Предложенный метод позволяет систематически выявлять и удалять избыточные слои трансформеров на основе эмпирической оценки производительности и дообучать сокращенную модель с использованием комбинированной функции потерь — как по логитам, так и по скрытым состояниям исходной модели. Эксперименты на модели Qwen2.5-3B показывают, что данный подход позволяет существенно сократить размер модели — до трети слоев — при незначительном снижении каче-

ства, что делает современные языковые модели более пригодными для внедрения в условиях ограниченных вычислительных ресурсов. Полученные результаты подтверждают эффективность итеративной дистилляции и открывают перспективы дальнейших исследований в области эффективной адаптации крупных языковых моделей. Исходный код открыт и размещен на GitHub <https://github.com/kaengreg/layer-wise-distillation>.

Дистилляция знаний. Современная концепция дистилляции знаний была предложена Хинтоном, где компактная модель-ученик обучается имитировать крупную модель-учителя, используя вероятностные предсказания учителя («мягкие» метки). Целевая функция дистилляции объединяет стандартную кросс-энтропию с дивергенцией Кульбака–Лейблера, что позволяет студенту воспроизводить не только жесткие метки, но и вероятностное распределение выходов учителя:

$$\mathcal{L}_{KD} = \frac{1}{N} \sum_{i=1}^N \left(- \sum_{j=1}^K y_{ij} \log p_{ij} + \lambda D_{KL}(\mathbf{p}_i \| \mathbf{q}_i) \right) \quad (1)$$

где y_{ij} — жесткая метка, p_{ij} и q_{ij} — вероятности классов для студента и учителя соответственно. Такая «скрытая информация» в выходах учителя отражает взаимосвязи между классами и обеспечивает более информативное обучение по сравнению с обучением только на жестких метках.

ShortGPT. Метод ShortGPT [5] направлен на уменьшение избыточности модели за счет выявления наименее значимых слоев трансформера с помощью метрики Block Influence (BI):

$$BI_i = 1 - \mathbb{E}_{X,t} \frac{X_{i,t}^T X_{i+1,t}}{\|X_{i,t}\|_2 \|X_{i+1,t}\|_2} \quad (2)$$

где низкое значение BI указывает на избыточность слоя.

Использование данной метрики позволяет эффективно находить и вырезать избыточные слои, снижая вычислительные затраты и расширяя возможности применения LLM.

Новые методы. В данной работе представлены два принципиальных улучшения подхода ShortGPT [5] для вырезания слоев в больших языковых моделях. Во-первых, предложен итеративный процесс, в ходе которого важность слоя, измеряемая с помощью метрики Block Influence, как в ShortGPT, пересчитывается после каждого удаления слоя. Такой подход учитывает изменения внутренней структуры модели, обеспечивая более взвешенные решения при дальнейшем прореживании. Кроме того, исследуется влияние дополнительного дообучения с использованием выходов (логитов и/или скрытых состояний) исходной модели.

Во-вторых, чтобы преодолеть ограничения метрики Block Influence и повысить объективность оценки важности слоев, предлагается новый алгоритм, основанный на тестировании производительности модели на наборе разнообразных задач. Для каждого варианта модели, полученного путем удаления одного из n слоев, измеряется качество на семи представительных датасетах,

охватывающих понимание текста, суммаризацию, копирование документов и параграфов, а также двусторонний перевод. Для оценки используются метрики F-Score и ROUGE; наименее значимые слои определяются по минимальному падению качества. Итоговый показатель (Benchmark score) рассчитывается как среднее по всем выбранным датасетам, что обеспечивает принятие решений на основе комплексной, задачно-ориентированной оценки.

Датасет для дистилляции. Для этапа дообучения и дистилляции использовался датасет из работы [1], основанный на коллекции IlyaGusev/rulm, включающей русскоязычные и англоязычные статьи из Википедии, художественные тексты, сообщения из социальных сетей и новостные публикации. Такое разнообразие источников обеспечивает полноценную передачу знаний и объективную оценку качества при дистилляции модели.

Функции потерь и настройка гиперпараметров. Для этапа дистилляции после вырезания слоев были изучены различные функции потерь: дивергенция Кульбака—Лейблера (KLDivLoss) для согласования выходных распределений (логитов), среднеквадратичная ошибка (MSELoss) для выравнивания скрытых состояний, а также их комбинации. Наилучшие результаты достигнуты при сочетании KLDivLoss на логитах с масштабированием $1/100$ и MSELoss на последних обучаемых и последних слоях модели с использованием learning rate $1e-4$ и максимальной длиной последовательности 2048.

Оценка методов дистилляции. Все новые методы оценки важности слоев и дистилляции были применены к модели Qwen2.5-3B, сокращенной с 36 до 28 слоев (2.47 млрд параметров). Сравнивались пять стратегий: ShortGPT, итеративный Block Influence, итеративный Block Influence с дообучением, итеративное уменьшение с задачно-ориентированной оценкой важности слоев и итеративная дистилляция с дообучением после каждого шага удаления. Метод итеративной дистилляции с задачно-ориентированной оценкой стабильно превосходил другие подходы, достигая значения Benchmark score 0.574 — существенно выше, чем у моделей без дообучения, которые теряли более 70% качества после вырезания 8-ми слоев.

Как видно из рисунка 1, модели, полученные только за счет вырезания слоев, показывают значительное падение качества, в то время как итеративная дистилляция позволяет сохранить высокое качество даже после значительного сокращения числа слоев. Анализ частоты удаления показывает, что промежуточные слои (17-24) наиболее избыточны, в то время как последние слои критически важны для генерации. Лучшая дистиллированная модель доступна на HuggingFace <https://huggingface.co/kaengreg/Qwen2.5-2B-layerwise-distilled>.

Заключение. В данной работе показано, что итеративное, основанное на данных сокращение числа слоев модели в сочетании с целенаправленным дообучением — как по логитам, так и по скрытым состояниям учителя — позволяет существенно уменьшить размер больших языковых моделей при минимальной потере качества. Применение этих методов к модели Qwen2.5-3B

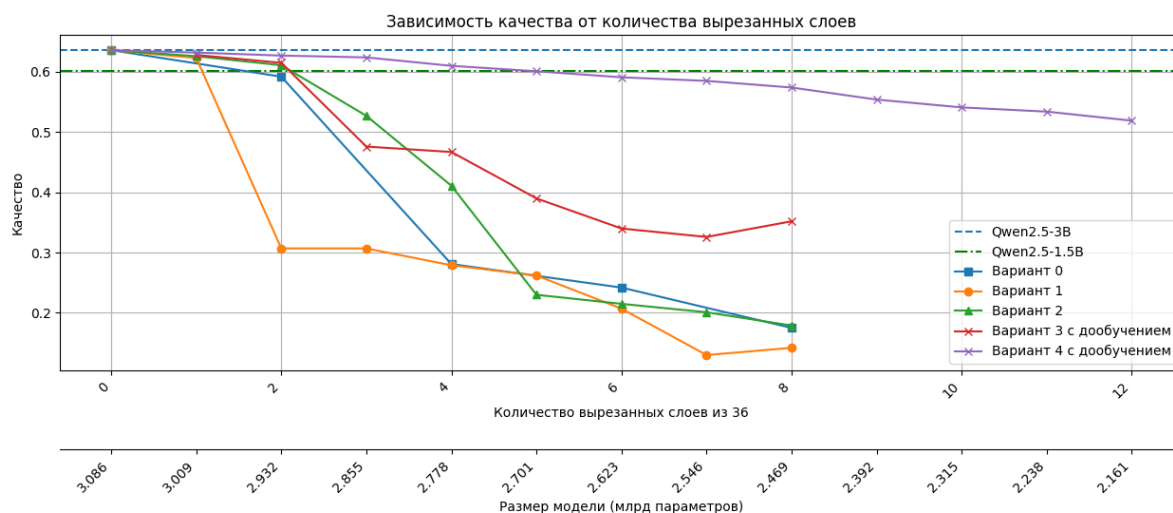


Рис. 1: Результаты различных подходов для модели Qwen2.5-3B

позволило значительно снизить количество параметров при сохранении высокой производительности, причем итеративная дистилляция с дообучением на каждом этапе оказалась значительно эффективнее простого сокращения слоев без дополнительной настройки.

Анализ показал, что промежуточные слои наиболее подвержены удалению без существенного ущерба для качества, в то время как финальные слои критически важны для сохранения функциональности модели. В целом, полученные результаты подчеркивают потенциал сочетания адаптивной оценки значимости слоев и последующей дистилляции для создания компактных и высокоэффективных языковых моделей, подходящих для внедрения в условиях ограниченных вычислительных ресурсов.

ЛИТЕРАТУРА

- [1] Тихомиров М.М., Чернышев Д.И. Facilitating Large Language Model Russian Adaptation with Learned Embedding Propagation // Journal of Language and Education. 2024. Т. 10, № 4. С. 130–145.
- [2] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I. Attention is all you need // Advances in neural information processing systems. 2017. Vol. 30.
- [3] Brown T., Mann B., Ryder N., Subbiah M., Kaplan J.D., Dhariwal P., Neelakantan A., Shyam P., Sastry G., Askell A. и др. Language models are few-shot learners // Advances in neural information processing systems. 2020. Vol. 33. P. 1877–1901.
- [4] Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics, 2019. P. 4171–4186. DOI: 10.18653/v1/N19-1423.

-
- [5] Men X., Xu M., Zhang Q., Wang B., Lin H., Lu Y., Han X., Chen W. ShortGPT: Layers in Large Language Models are More Redundant Than You Expect // CoRR. 2024.

Разработка и исследование метода оценки производительности инференса сверточных нейросетевых моделей

Работа удостоена диплома 1 степени

Козинцева Анжелика Александровна

Кафедра суперкомпьютеров и квантовой информатики

e-mail: s02210423@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. доц. Попова Нина Николаевна

Введение. Работа посвящена исследованию производительности применения обученной модели к новым данным - этапа инференса. Поскольку инференс предполагает многократное использование, его оптимизация с учетом особенностей архитектуры является актуальной задачей.

Цель. Целью работы является исследование методов оценки и предсказания производительности инференса сверточных нейросетевых моделей. Мотивация в предоставлении пользователям суперкомпьютера «МГУ 270» информации для выбора и настройки параметров для инференса уже обученных моделей.

Исследование параметров, влияющих на производительность инференса. На основе проведенного анализа были выделены для исследования следующие параметры, влияющие на время инференса: количество GPU, размер батча, формат представления нейросетевых моделей, характеристики датасета.

Для проведения исследований был написан код на Python с использованием фреймворка PyTorch. Исследовались 15 предобученных сверточных нейросетевых моделей из официального репозитория TorchVision на датасете CIFAR-10. Запуски проводились на узле суперкомпьютера «МГУ 270».

Одной из самых значимых оказалась зависимость времени инференса от количества GPU и размера батча. Увеличение размера минибатча ускоряет время выполнения датасета, но слишком большие минибатчи приводят к перегрузке памяти GPU, что негативно сказывается на производительности. Таким образом, для каждого количества GPU свой оптимальный размер батча.

Аналитический подход. Подход заключается в моделировании процесса инференса через математические формулы. В реализованном аналитическом подходе рассчитываются время вычислений и время доступа к памяти по представленным формулам. Далее используется модель Roofline[1], которая позволяет определить, какое из двух ограничений — вычислительная мощность или пропускная способность памяти — является узким местом производительности при выполнении задач. Кроме того учитываются коммутационное время и дополнительные накладные расходы[2].

Предложенный аналитический метод представлен в виде функции на Python, принимающей на вход переменные:

- batch_size размер батча;

- `model_flops` количество вычислений с плавающей точкой для одного изображения;
- `num_gpus` количество используемых GPU;
- `memory_bytes_per_sample` байты, занимаемые моделью в памяти;
- `input_size`, `output_size` размеры входа и выхода модели.

В ходе тестирования метода на предобученных моделях была получена метрика R-квадрат = 0.52, что указывает на умеренную корреляцию между предсказанными и реальными значениями.

Подход на основе машинного обучения. Проведённые эксперименты выявили несколько ключевых факторов, существенно влияющих на производительность инференса: количество GPU, размер батча и характеристики архитектуры модели. Они были выбраны входными признаками предсказательной модели.

В качестве характеристик архитектуры модели были предложены `flops` (коррелирует с временем выполнения) и количество параметров (как мера памяти и передачи данных).

На основе этих признаков предложено для построения и сравнение несколько предсказательных моделей машинного обучения: Линейная регрессия, Random Forest, Gradient Boosting. У линейной регрессии легко интерпретируемые коэффициенты, что позволяет понять, какие факторы больше всего влияют на время инференса[3]. Random Forest хорошо обрабатывает нелинейные зависимости[4], а в экспериментах такие были замечены. Gradient Boosting тоже обрабатывает нелинейности, но кроме того эффективно работает с разнородными данными (в нашем случае это число GPU и FLOPs)[5].

Результаты тестирования обученных моделей представлены таблицей.

	Linear Regression	Random Forest	Gradient Boosting
MSE	0.057545	0.00406	0.00372
R-squared	0.3543	0.956	0.962

Для использования лучшего из разработанных методов (обученная модель градиентного бустинга) была реализована библиотека `ml_package`.

Ссылка на библиотеку: https://github.com/angelika2003/ml_package.git.

Для получения предсказания времени инференса необходимо скачать библиотеку и вызвать функцию `predict` с необходимыми параметрами.

Основные результаты. На основе анализа определены параметры для проведения экспериментального исследования инференса на узле суперкомпьютера МГУ-270. В число параметров включены: количество GPU, размер батча, формат представления модели, характеристики датасета. В качестве исследуемых моделей выбраны предобученные модели PyTorch. Выявлены зависимости производительности инференса от выбранных параметров.

Предложены и реализованы аналитический метод и регрессионная модель для предсказания времени выполнения инференса сверточных нейросетей.

Показано, что модель, построенная на основе Gradient Boosting, имеет наиболее высокую точность R-squared 0.962.

Программная реализация разработанной регрессионной модели оформлена в виде библиотеки и размещена в свободном доступе.

ЛИТЕРАТУРА

- [1] Samuel Williams, Andrew Waterman, David Patterson. Roofline: An Insightful Visual Performance Model for Multicore Architectures// Communications of the ACM. 2009.
- [2] Multi-GPU Training Guidelines// NVIDIA. 2023.
- [3] Trevor Hastie, Robert Tibshirani // The Elements of Statistical Learning. Springer. 2009.
- [4] Leo Breiman. Random Forests. //Machine Learning. 2001.
- [5] Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. //The Annals of Statistics. 2001.

Исследование начально-краевых задач для обобщенного уравнения типа Шрёдингера

Работа удостоена диплома 1 степени

Колобродов Андрей

Кафедра математической физики

e-mail: pikart@kc.twcbc.com

Научный руководитель — д.ф.-м.н. проф. Разгулин Александр Витальевич

Введение. Во второй половине 1990-х годов для описания распространения коротких световых импульсов в нелинейной среде было получено новое обобщенное нелинейное уравнение Шрёдингера (ОНУШ) в рамках медленно эволюционирующего волнового приближения. ОНУШ содержит смешанные производные по времени и по пространственным координатам, производную по времени от кубического нелинейного отклика среды, а также содержит члены, описывающие эффекты дисперсии второго порядка (D_2) и дифракции пучка (D) [1]. Для описываемых ОНУШ финитных по времени и поперечным координатам пучков в [2] получены инварианты (энергии, гамильтониан, спектральный, и др.). В отличие от статей [1, 2], где рассматривается уравнение с финитными граничными условиями по времени, в нашей работе рассматривается случай периодических условий по времени.

Постановка задачи. Распространение пучка в среде с нелинейным кубическим откликом описывается ОНУШ:

$$\left(1 + i\gamma \frac{\partial}{\partial t}\right) \frac{\partial A}{\partial z} + iD_2 \left(1 + i\gamma \frac{\partial}{\partial t}\right) \frac{\partial^2 A}{\partial t^2} + iD \frac{\partial^2 A}{\partial x^2} + i\alpha \left(1 + i\gamma \frac{\partial}{\partial t}\right)^2 (|A|^2 A) = 0, \quad (1)$$

где $A = A(z, x, t)$, $\Omega = [0, L_z] \times [0, L_x] \times [0, L_t]$ и $A|_{z=0} = A_0(x, t)$ - входная волна. Пучок распространяется вдоль оси z , x - поперечная координата, которая нормализована относительно радиуса пучка, $\gamma = \frac{1}{\omega_0 \tau_p}$, ω_0 - несущая частота волнового пакета, τ_p - длительность распространения пучка. Переменная t нормализована относительно τ_p . Ставятся граничные условия по x и по t :

$$A|_{x=0} = A|_{x=L_x} = 0, \quad (2)$$

$$A|_{t=0} = A|_{t=L_t}; \quad \frac{\partial A}{\partial t} \Big|_{t=0} = \frac{\partial A}{\partial t} \Big|_{t=L_t}; \quad \frac{\partial^2 A}{\partial t^2} \Big|_{t=0} = \frac{\partial^2 A}{\partial t^2} \Big|_{t=L_t}. \quad (3)$$

Цели исследования и результаты. Цель выпускной квалификационной работы заключалась в аналитическом и численном изучении начально-краевой задачи (1)-(3). Сначала были исследованы различные комбинации граничных условий по времени. Было установлено, что наборы периодических условий являются регулярными по Наймарку, а наборы, состоящие только из нулевых граничных условий, нерегулярны. Далее были выведены свой-

ства оператора $D_\gamma = (1 + i\gamma \frac{\partial}{\partial t})$: его обратимость, собственные значения, симметричность и выражение для обратного оператора. Чтобы получить выражения для инварианта энергии и гамильтониана, порядок уравнения (1) был понижен с помощью D_γ^{-1} .

Теорема 1. *Задача (1)-(3) обладает инвариантом:*

$$I_E(z) = \int_0^{L_x} \int_0^{L_t} |A(z, x, t)|^2 dx dt.$$

Теорема 2. *Задача (1)-(3) обладает инвариантом:*

$$I_H(z) = \int_0^{L_x} \int_0^{L_t} \left[iD_2 \cdot D_\gamma^{-1} \left(\frac{\partial A}{\partial t} \right) \cdot \frac{\partial A^*}{\partial t} + iD \left| D_\gamma^{-1} \left(\frac{\partial A}{\partial x} \right) \right|^2 - \frac{i\alpha}{2} |A|^4 \right] dx dt.$$

Для численного решения (1)-(3) разработана нелинейная разностная схема и двухстадийный процесс ее реализации. На первой стадии значения A ищутся методом трехдиагональной циклической прогонки по переменной t . На второй стадии используется трехдиагональная прогонка по переменной x . Схема реализована в составе программного комплекса на языке С для моделирования начально-краевой задачи. Проведенные методические расчеты для линейного и нелинейного уравнения с контролем сохранения инварианта энергии показали эффекты распыливания пучка при отрицательном коэффициенте α , самофокусирования при положительном коэффициенте α и влияния дисперсии второго порядка (D_2) и дифракции (D).

ЛИТЕРАТУРА

- [1] Brabec T., Krausz F. Nonlinear optical pulse propagation in the single-cycle regime // Phys. Rev. Lett. 1997. Vol. 78. № 17. P. 3282–3285.
- [2] Trofimov V., Stepanenko S., Razgulin A. Conservation laws of femtosecond pulse propagation described by generalized nonlinear Schrodinger equation with cubic nonlinearity // Mathematics and Computers in Simulation. 2021. Vol. 182. P. 366–396.

Разработка метода построения отображения в однородное пространство субъективных оценок минимально различимых искажений изображений, собранных в различных условиях

Костычев Александр Михайлович

Кафедра интеллектуальных информационных технологий

e-mail: aleksandr.kostychev@graphics.cs.msu.ru

Научный руководитель — к.ф.-м.н. с.н.с. Ватолин Дмитрий Сергеевич

Введение. Сегодня наибольшую часть трафика в сети занимают видео и изображения. Благодаря стремительному развитию технологий хранения, передачи и отображения контента, пользователи предъявляют все более высокие требования к качеству получаемых изображений и видео. Однако при этом важным аспектом остается размер отправляемого контента. Это привело к развитию методов сжатия, способных достигать высокого уровня сжатия без заметного для человека ухудшения визуального качества. Для оценки визуального качества изображений применяются как объективные (например, пиковое отношение сигнала к шуму (PSNR), индекс структурного сходства (SSIM) [1]), так и субъективные методы. Объективные методы используют математические модели для оценки искажений, тогда как субъективные методы основаны на восприятии качества реальными наблюдателями. Несмотря на прогресс в разработке объективных моделей качества изображений, субъективная оценка является ключевым показателем качества, поскольку именно она непосредственно отражает восприятие контента пользователем.

Одним из важных способов субъективной оценки качества занимает концепция минимально различимых искажений (JND), определяющая максимальный уровень сжатия (далее — точка JND), при котором изменения в изображении остаются незаметными для человеческого глаза. Этот подход используется при сжатии медиаданных. Обычно разрабатывается нейросетевая модель, обучаемая на субъективных данных и предсказывающая точку JND.

Нейросетевые модели требуют достаточно большого количества обучающих данных. Один из способов увеличить число данных — объединить несколько наборов данных в один. Однако существует серьезная проблема: субъективные оценки из разных наборов данных собраны в различных экспериментальных условиях, с использованием разных методик и групп наблюдателей. Поэтому они находятся в разных пространствах, и их нельзя просто объединить в один набор данных.

Данная работа направлена на разработку метода, позволяющего отображать субъективные JND оценки изображений, полученные в разных условиях, в единое однородное пространство, что позволит объединять данные из различных экспериментальных исследований.

Предложенный метод состоит из нескольких частей:

1. Выбор репрезентативных изображений из исходных наборов данных;
2. Субъективная разметка выбранных изображений;
3. Подсчет значимых признаков изображений по специальным картам;
4. Обучение модели машинного обучения на размеченных изображениях;
5. Применение обученной модели машинного обучения к неразмеченной части наборов данных.

В ходе работы были протестированы следующие признаки изображений:

— Яркостные признаки:

1. **Среднее значение канала.** Оно отражает общую интенсивность канала, что важно при оценке JND, так как человеческий глаз более чувствителен к изменениям в ярких областях, чем в темных.
2. **Однородность канала.** Низкая однородность указывает на высокую вариативность пикселей (например, шум, детали), где искажения могут быть менее заметны из-за маскирующего эффекта текстур. В однородных областях даже малые искажения (например, артефакты сжатия) становятся заметными, что критично при оценке JND.
3. **Энтропия канала.** Энтропия отражает предсказуемость изображения: в предсказуемых областях изменения легче обнаружить.

— Пространственные признаки:

1. **Пространственная информация канала.** В областях с высокой пространственной активностью (например, сложные текстуры) искажения менее заметны из-за маскировки, тогда как в гладких областях даже малые отклонения видны.
2. **Красочность изображения.** Высокая красочность означает, что искажения цветов (например, сдвиги оттенков) могут быть более заметными.

— Точка JND, полученная в исходной разметке набора данных.

Также описанные выше признаки считались по различным маскам изображений. Это было сделано из расчета, что зритель начинает замечать искажения не сразу на всем изображении, а в его конкретных областях. В работе признаки считались по следующим маскам:

- Левый верхний угол изображения;
- Правый верхний угол изображения;
- Левый нижний угол изображения;
- Правый нижний угол изображения;
- Центральный квадрант изображения;
- Часть изображения, где PSNR между эталонным изображением и его сжатой версией меньше определенного порога;

— Часть изображения, где SSIM [1] между эталонным изображением и его сжатой версией меньше определенного порога.

Для отбора наиболее значимых признаков была вычислена F-статистика [2] между каждым признаком и целевой переменной на трех наборах данных [3, 4, 5]. Было получено, что наиболее значимыми признаками в основном являются те, которые были вычислены на основе масок с учетом PSNR или SSIM [1].

В работе в качестве модели, предсказывающей точки JND, были протестированы шесть моделей машинного обучения. Перед обучением моделей все признаки были приведены к единому масштабу с помощью линейного преобразования, которое центрирует данные (вычитает среднее значение) и масштабирует их (делит на стандартное отклонение). Для каждой модели подбирались оптимальные гиперпараметры, в том числе количество признаков, лучших по F-статистике, которые использовались при обучении. Все методы были протестированы на трех наборах данных [3, 4, 5]. В Таблице 1 представлены лучшие для каждой модели результаты по средней абсолютной ошибке предсказания на кросс-валидации для наборов данных MCL-JCI [3], VVC-Dataset [4] и KonJND-1k [5]. Наилучшее качество продемонстрировал метод на основе опорных векторов — ошибка предсказания составила менее 3%.

Метод	MCL-JCI [3]	VVC-Dataset [4]	KonJND-1k [5]
Среднее стандартное отклонение	13.12	12.65	14.86
Случайный лес	3.43	<u>2.18</u>	3.38
Метод k-ближайших соседей	3.38	2.24	3.41
Линейная регрессия	3.55	2.43	3.61
Лассо-регрессия	2.97	2.36	3.18
Гребневая регрессия	<u>2.79</u>	2.19	<u>3.06</u>
Метод опорных векторов	2.48	2.17	2.98

Таблица 6: Средняя абсолютная ошибка предсказания точки JND на кросс-валидации на разных наборах данных.

Также в ходе работы были разработаны два пользовательских интерфейса для субъективной оценки точки JND изображений с помощью краудсорсинга:

1. **«Бок-о-бок»**. Пользователю показываются рядом друг с другом исходное изображение I и его сжатый вариант I_d . Пользователь должен ответить, видит ли он разницу между представленной парой изображений. Видео взаимодействия с интерфейсом доступно по ссылке: <https://clck.ru/3M5Veb>
2. **«Мерцающий тест»**. Пользователю показывается исходное изображение I , которое с частотой 8 Гц меняется с его сжатым вариантом I_d . Пользователь должен ответить, видит ли он мерцание на изображении или нет. Если мерцание заметно, то считается, что между парой изображений

есть разница. Видео взаимодействия с интерфейсом доступно по ссылке: <https://clck.ru/3M5Veq>

Предложенный метод позволяет экономить ресурсы при субъективной разметке наборов данных, а также довольно быстро переводить оценки из новых наборов данных в однородное пространство.

ЛИТЕРАТУРА

- [1] Image Quality Assessment: From Error Measurement to Structural Similarity / Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli // IEEE transactions on image processing. 2004. T. 13, № 1.
- [2] Guyon I., Elisseeff A. An introduction to variable and feature selection // Journal of machine learning research. 2003. T. 3. С. 1157–1182.
- [3] Statistical study on perceived JPEG image quality via MCL-JCI dataset construction and analysis / L. Jina, J. Y. Lin, S. Hu, H. Wang, P. Wang, I. Katsavounidis, A. Aaron, C.-C. J. Kuo // Electronic Imaging. 2016. № 13. С. 1–9.
- [4] A JND dataset based on VVC compressed images / X. Shen, Z. Ni, W. Yang, X. Zhang, S. Wang, S. Kwong // 2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW). 2020. С. 1–6.
- [5] Large-scale crowdsourced subjective assessment of picturewise just noticeable difference / H. Lin, G. Chen, M. Jenadeleh, V. Hosu, U.-D. Reips, R. Hamzaoui, D. Saupe // IEEE transactions on circuits and systems for video technology. 2022. T. 32, № 9. С. 5859–5873.

Оптимальная фильтрация состояний специального марковского скачкообразного процесса по комплексным наблюдениям

Работа удостоена диплома 2 степени

Куринов Юрий Наранович

Кафедра математической статистики

e-mail: kurinovurij@gmail.com

Научный руководитель — д.ф.-м.н. проф. Борисов Андрей Владимирович

В работе исследован новый класс случайных процессов, интересный с точки зрения приложений, предложены методы оптимальной фильтрации состояний рассматриваемого процесса по разнородным наблюдениям, а также проведён численный анализ точности полученных оценок на примере из области телекоммуникаций.

Цель работы: провести вероятностный анализ класса специальных марковских скачкообразных процессов (СМСП), разработать и проанализировать методы оценивания скрытых состояний СМСП по комплексу разнородных наблюдений.

Определение и вероятностный анализ класса СМСП. Определим случайный процесс, состоящий из двух блочных компонент $Z_t \triangleq \text{col}(\theta_t, Y_t)$, следующим образом

- θ_t : марковский процесс с конечным числом состояний, принимающий значения в $\mathbb{S}^N \triangleq \{e_1, \dots, e_N\}$, множестве единичных векторов из $\mathbb{R}^N$,
- Y_t : кусочно-постоянный процесс, моменты скачков которого совпадают с моментами скачков θ_t , а его распределение описывается вектором $\pi(y) \triangleq \text{col}(\pi_1(y), \dots, \pi_N(y))$:

$$P(Y_t \in B | \theta_t = e_n) = \int_B \pi_n(y) dy, \quad \forall B \in \mathcal{B}(\mathbb{R}^M), \quad n = \overline{1, N}.$$

Теорема 1. *Процесс Z_t — марковский.*

Следующая теорема позволяет описывать динамику функции исследуемого процесса в удобном для анализа виде. Рассматривается процесс, ассоциированный с некоторой функцией f от СМСП: $f_t \triangleq f(\theta_t, Y_t)\theta_t$ и $\mathbf{f}_t \triangleq \text{col}(\theta_t, f_t)$.

Теорема 2. *Процесс $\mathbf{f}_t$ — единственное сильное решение замкнутой системы линейных стохастических дифференциальных уравнений (СДУ):*

$$\mathbf{f}_t = \mathbf{f}_0 + \int_0^t D^f(s) \mathbf{f}_s ds + \mu_t^f,$$

где μ_t^f — мартингал, $D^f(t)$ — детерминированная матричнозначная функция времени.

Для СМСП получены его вероятностные характеристики, вполне описывающие его вероятностные свойства, такие как инфинитезимальный генератор, прямые и обратные уравнения Колмогорова и другие [1].

Постановка и решение задач фильтрации СМСП. Состояние СМСП $Z_t = \text{col}(\theta_t, Y_t)$ предполагается скрытым и подлежит оцениванию.

Доступные наблюдения косвенно зависят от Z_t и имеют следующий вид:

$$\xi_t = \int_0^t g(\theta_s, Y_s) ds + w_t \in \mathbb{R}^K,$$

– непрерывные наблюдения с аддитивными винеровскими шумами w_t ,

$$\eta_t = \int_0^t h(\theta_s, Y_s) ds + \mu_t^\eta \in \mathbb{R}^L,$$

– наблюдения со считающими компонентами, μ_t^η – мартингал.

$\mathcal{O}_t \triangleq \sigma(\{\xi_s, \eta_s, s \leq t\})$ – сигма-алгебра, порождённая наблюдениями.

Задача абсолютно оптимальной фильтрации состоит в нахождении $\hat{Z}_t$:

$$\hat{Z}_t \in \underset{\tilde{Z}_t \in \mathcal{E}_t}{\text{Argmin}} \mathbb{E} \left[\|\tilde{Z}_t - Z_t\|^2 \right],$$

где $\mathcal{E}_t$ – класс всех $\mathcal{O}_t$ -измеримых функций, таких, что $\mathbb{E}[\|\tilde{Z}_t\|^2] < \infty$.

Известно, что $\hat{Z}_t = \mathbb{E}[Z_t | \mathcal{O}_t]$ – одно из решений данной задачи. Для вычисления математического ожидания была рассмотрена условная плотность распределения относительно наблюдений $\hat{\psi}(t, y) \triangleq \text{col}(\hat{\psi}_1(t, y), \dots, \hat{\psi}_N(t, y))$:

$$\mathbb{P}(\theta_t = e_n, Y_t \in B | \mathcal{O}_t) = \int_B \hat{\psi}_n(t, y) dy, \forall B \in \mathcal{B}(\mathbb{R}^M).$$

В работе получен аналог уравнений Кушнера-Стратоновича, описывающий динамику условной плотности.

Теорема 3. *Существует $\hat{\psi}(t, y)$, удовлетворяющая системе стохастических интегро-дифференциальных уравнений [2]:*

$$\begin{aligned} \hat{\psi}_n(t, y) = & p_0^n \phi_n(y) + \int_0^t \left[\lambda_{nn}(s) \hat{\psi}_n(s-, y) + \pi_n(y) e_n^\top \bar{\Lambda}^\top(s) \hat{\theta}_{s-} \right] ds + \\ & + \int_0^t \hat{\psi}_n(s-, y) (g(s-, e_n, y) - \hat{g}_{s-})^\top (d\xi_s - \hat{g}_{s-} ds) + \\ & + \int_0^t \hat{\psi}_n(s-, y) (h(s-, e_n, y) - \hat{h}_{s-})^\top (\text{diag } \hat{h}_{s-})^{-1} (d\eta_s - \hat{h}_{s-} ds), \end{aligned}$$

$$n = \overline{1, N}.$$

Таким образом, оптимальная оценка произвольного сигнала – гильбертова случайного процесса, функции состояния системы, может быть задана как интеграл по условной плотности.

Несмотря на наилучшую возможную точность, на практике данный подход оказывается ресурсозатратным. Его численная реализация требует многомерной сеточной аппроксимации условной плотности и проведения дополнительных операций интегрирования. В прикладных областях, где необходима эффективность численного алгоритма, может оказаться полезной иная постановка задачи.

Задача оптимальной линейной фильтрации состоит в нахождении $\bar{Z}_t$:

$$\bar{Z}_t \in \underset{\tilde{Z}_t \in \mathcal{L}_t}{\text{Argmin}} \mathbf{E} \left[\|\tilde{Z}_t - Z_t\|^2 \right],$$

где $\mathcal{L}_t = \left\{ \bar{Z}_t : \bar{Z}_t = \alpha_t + \int_0^t \beta(t, \tau) d\xi_\tau + \int_0^t \gamma(t, \tau) d\eta_\tau \right\}$ – класс линейных преобразований наблюдений, $\mathbf{E}[\|\bar{Z}_t\|^2] < \infty$.

Для решения данной задачи были рассмотрены процессы:

$$X_t \triangleq \text{col} \left(\theta_t, Y_t^1 \theta_t, \dots, Y_t^M \theta_t, g_t^1, \dots, g_t^K, h_t^1, \dots, h_t^L \right),$$

$$U_t \triangleq \text{col} \left(\xi_t^1, \dots, \xi_t^K, \eta_t^1, \dots, \eta_t^L \right),$$

$g_t^i = g^i(t, \theta_t, Y_t) \theta_t$, $i = \overline{1, K}$ и $h_t^j = h^j(t, \theta_t, Y_t) \theta_t$, $j = \overline{1, L}$ – процессы, ассоциированные с полезными сигналами из наблюдений.

Теорема 2 позволяет представить систему наблюдения в виде линейной негауссовской:

$$X_t = X_0 + \int_0^t D^X(s) X_s ds + \int_0^t b^{\frac{1}{2}}(s) d\mu_s^X,$$

$$U_t = \int_0^t D^U(s) X_s ds + \int_0^t B^{\frac{1}{2}}(s) d\mu_s^U,$$

μ_t^X и μ_t^U – процессы с ортогональными приращениями и единичной интенсивностью.

К данной системе применим фильтр Калмана-Бьюси, и верна следующая теорема.

Теорема 4. *Оптимальная линейная оценка является решением системы линейных СДУ – уравнений фильтра Калмана-Бьюси:*

$$d\bar{X}_t = D^X(t) \bar{X}_t dt + k(t) (D^U)^T b(t)^{-1} [dU_t - D^U \bar{X}_t dt],$$

$$\bar{X}_0 = \mathbf{E} X_0.$$

Матрица ковариации ошибки оценки $k(t) \triangleq \text{cov}(\bar{X}_t - X_t, \bar{X}_t - X_t)$ – решение уравнения Рикатти:

$$\dot{k}(t) = D^X(t) k(t) + k(t) (D^X(t))^T - k(t) (D^U)^T b(t)^{-1} D^U k(t) + B(t),$$

$$k(0) = \text{cov}(X_0, X_0).$$

Пример СМСП: модель состояния сетевого канала. Полученные результаты могут применяться в задаче мониторинга состояния гетерогенного канала связи. Она заключается в оценивании состояния и характеристик канала по наблюдениям времени кругового обращения сегмента данных (RTT) и потоков потерь пакетов.

Модель канала описывается СМСП $Z_t = \text{col}(\theta_t, Y_t)$, его компоненты несут следующий смысл:

— θ_t – качественное состояние канала,

- $\{\theta_t = e_1\}$ – умеренная нагрузка,
- $\{\theta_t = e_2\}$ – состояние, предшествующее перегрузке,
- $\{\theta_t = e_3\}$ – перегрузка проводного участка,
- $\{\theta_t = e_4\}$ – потеря сигнала на беспроводном участке.
- $Y_t = \text{col}(Y_t^1, Y_t^2, Y_t^3)$ – числовые характеристики канала,
 - Y_t^1 – среднее RTT,
 - Y_t^2 – доля потерянных пакетов,
 - Y_t^3 – доля тайм-аутов.

Наблюдения:

- $\xi_t \triangleq \int_0^t Y_s^1 ds + w_t$ – зашумлённые наблюдения RTT,
- $\eta_t^1 \triangleq \int_0^t Y_s^2 / Y_s^1 ds + \mu_t^{\eta^1}$ – поток потерь пакетов,
- $\eta_t^2 \triangleq \int_0^t Y_s^3 / Y_s^1 ds + \mu_t^{\eta^2}$ – поток тайм-аутов.

Чтобы сравнить качество полученных оценок, был рассмотрен следующий показатель качества:

$$\rho_q \triangleq \frac{\int_0^T \mathbb{E}[\|q_t - \tilde{q}_t\|^2] dt}{\int_0^T \mathbb{E}[\|q_t - \mathbb{E}[q_t]\|^2] dt},$$

этот коэффициент может рассматриваться как отношение дисперсии ошибки оценки к дисперсии тривиальной оценки – безусловного математического ожидания, являющегося наилучшей априорной оценкой.

В таблице приведены результаты численных экспериментов:

	ρ_θ	ρ_{Y^1}	ρ_{Y^2}	ρ_{Y^3}
Линейный фильтр	0.624	0.088	0.172	0.240
Нелинейный фильтр	0.357	0.049	0.075	0.118

В работе исследован класс специальных марковских скачкообразных процессов, получены уравнения оптимальной линейной и нелинейной фильтрации, рассмотрен пример из телекоммуникаций и проведён численный анализ точности методов оценивания. Результаты опубликованы в статьях [1,2].

ЛИТЕРАТУРА

- [1] Борисов А. В., Куринов Ю. Н., Смелянский Р. Л. Вероятностный анализ класса марковских скачкообразных процессов // Информатика и её применения. 2024. Т. 18, № 3. С. 9–16.
- [2] Борисов А. В., Куринов Ю. Н., Смелянский Р. Л. Фильтрация состояний класса марковских скачкообразных процессов по разнородным наблюдениям с аддитивными шумами // Информатика и её применения. 2024. Т. 18, № 4. С. 45–53.

Анализ и применение многофакторных моделей динамики лекарственных средств в медицинских исследованиях

Лазар Владислав Игоревич

Кафедра математической статистики

e-mail: vlad.lazar.ik528@yandex.ru

Научный руководитель — к.ф.-м.н. доц. Захарова Татьяна Валерьевна

В работе рассматривается задача проверки лекарственных препаратов на биоэквивалентность. Исследования биоэквивалентности лежат в основе воспроизведения лекарственных препаратов, подтвердивших свою эффективность и безопасность. Основным методом проверки гипотезы биоэквивалентности является процедура двух односторонних тестов Шуирманна. Она используется в течение многих лет и подтвердила свою пригодность для доказательства эквивалентной биодоступности.

В исследованиях биоэквивалентности одной из используемых сравнительных характеристик лекарств является площадь под кривой (AUC) «концентрация—время». Основной проблемой классического критерия является нечувствительность к форме этой кривой, что может влечь за собой некорректную оценку действия воспроизведенного лекарства.

Авторами предложен новый подход в оценке параметров исследуемых лекарств. В качестве основы берётся оценка траектории с помощью математической модели **PBFTPK**. В оригинальной статье предлагается оценка параметров модели по конечному числу наблюдений методом минимизации среднеквадратичной функции потерь методом Левенберга-Марквардта. В работе представлены модификации оригинальной модели устойчивые к проблемам возникающим в реальных данных - неравномерности наблюдений по времени и наличию более одного пика траектории. Неоднородность наблюдений влечёт за собой уменьшение значимости более редких наблюдений и как следствие худшее качество модели на соответствующих участках. В случае многих пиков классические алгоритмы рассматривают побочные пики как шум в данных, вследствие чего теряется важная для пролонгированных форм информация о действии лекарства.

На начальном этапе исследований проводилась оценка траектории с помощью оригинальной модели. Методом последовательного квадратичного программирования (**SLSQP**) проводилась оценка параметров следующей параметризованной функции:

$$\begin{cases} C(t) = \frac{FD}{V_d} \frac{e^{-k_{el}t} - e^{k_a t}}{k_a - k_{el}}, & 0 \leq t \leq \tau, \\ C(t) = C(\tau)e^{-k_{el}(t-\tau)}, & \tau < t < +\infty \end{cases}$$

, где

— F - биодоступная доля дозы

— D - доза вещества

- V_d - объём распределения вещества
- k_a - коэффициент абсорбции
- k_{el} - коэффициент элиминации

Для дальнейшей работы важно выделить два этапа - этап абсорбции ($t \leq \tau$) и этап элиминации ($t > \tau$).

Для устранения первой проблемы авторами предложен метод взвешивания функции потерь в соответствии с фазой действия лекарственного вещества. Для этого рассматривается λ -взвешенная среднеквадратичная функции потерь ($\mathbf{WMSE}_\lambda$):

$$L_\lambda(t, C, \hat{C}) = \frac{1}{\#(t)} \sum_{t_i \in t} \left(C(t_i) - \hat{C}(t_i) \right)^2 \cdot \begin{cases} 1, & t_i \leq \tau \\ \lambda, & t_i > \tau \end{cases}$$

Такой метод взвешивания объясняется особенностями используемых данных, а конкретно – малой длительностью фазы абсорбции относительно фазы элиминации и большей плотностью наблюдений именно на этапе абсорбции.

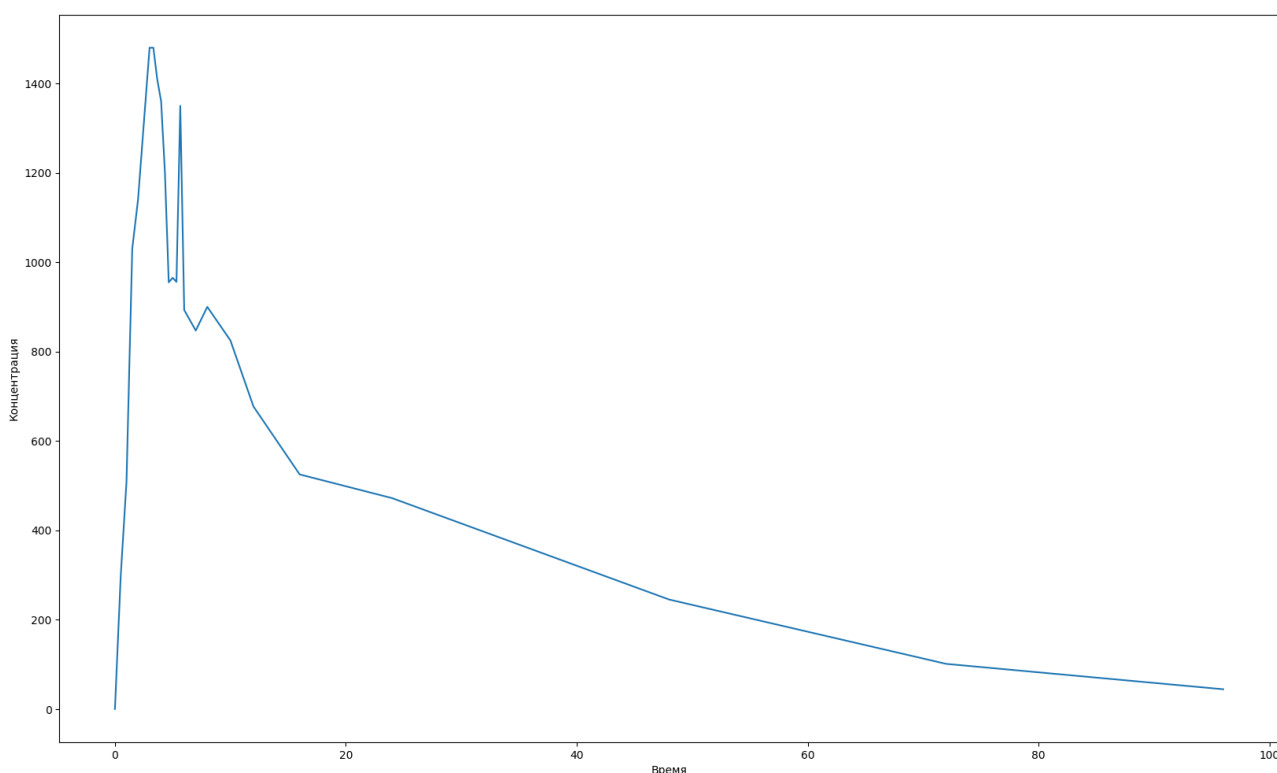


Рис. 1: Пример данных для исследования

Для сохранения информации о множественных пиках траектории был предложен метод ансамблирования математических моделей. Под ансамблированием понимается следующий алгоритм действий:

1. Задаётся размер ансамбля - количество базовых моделей.
2. На вход первой модели для оценивания подаются данные.

3. Вычисляется разность оценивания и исходных данных, такую разность будем называть **остатком**.
4. Полученный остаток подаётся на вход следующей модели в ансамбле, после чего процесс повторяется начиная с пункта 2.

Помимо повышения точности оценивания (оценивание остатка модели вместо его отбрасывания) полученный алгоритм не теряет физической интерпретируемости. В данном случае каждую модель в ансамбле можно рассматривать как отдельную независимо действующую фракцию исходного вещества. Тогда полная оценка траектории получается как сумма оценок всех моделей в ансамбле, то есть концентрация вещества в крови есть сумма концентраций его частей, что соотносится с физическим смыслом исследуемого явления.

Для улучшения качества ансамблирования была модифицирована исходная математическая модель - введён параметр задержки абсорбции τ_0 :

$$C(t) = \begin{cases} 0, & 0 \leq t \leq \tau_0 \\ \frac{FDk_a}{V_d(k_a - k_{el})}(e^{-k_{el}(t-\tau_0)} - e^{-k_a(t-\tau_0)}), & \tau_0 < t \leq \tau \\ C(\tau)e^{-k_{el}((t-\tau_0)-\tau)}, & t > \tau \end{cases}$$

Физический смысл параметра τ_0 состоит в задержке реакции организма на препарат. С точки зрения качества модели при помощи этого параметра можно оценивать траектории для которых не выполнено условие о старте из нуля. Особенно это актуально для остатков полученных при оценивании предыдущей моделью в ансамбле. Поскольку оптимизация по τ и τ_0 не является гладкой, требуется другой метод оценки для дальнейшего априорного использования при оценке параметров модели. В работе представлены два возможных для использования метода:

1. Минимаксный метод

$$\tau = \operatorname{argmax}_{t_i \in t} C(t_i) \quad \tau_0 = \operatorname{argmin}_{t_i \in t, t_i < \tau} C(t_i)$$

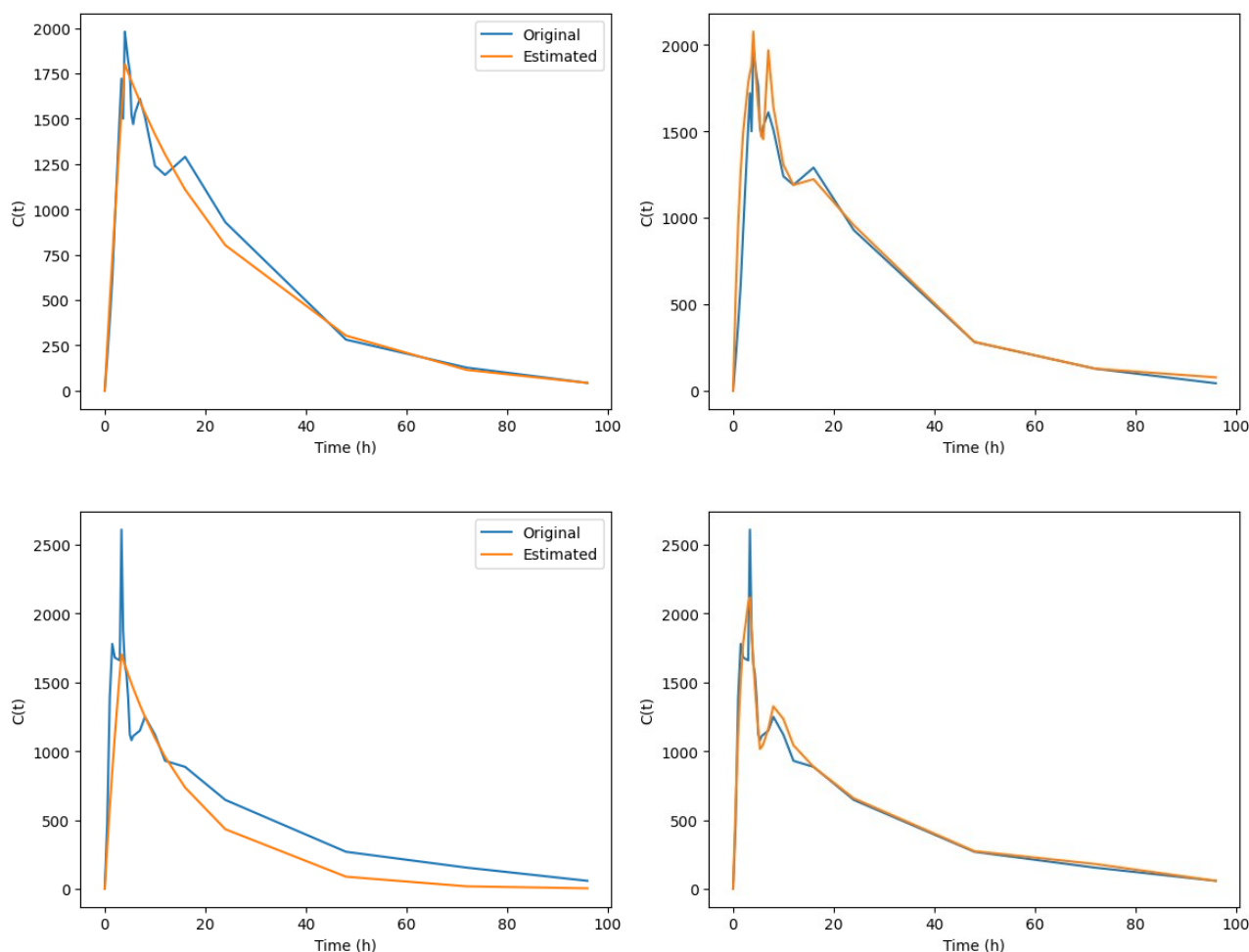
2. Пиковый метод

$$\tau = \max_{t_i \in t} \operatorname{Argmax} C(t_i) \quad \tau_0 = \max_{t_i \in t, t_i < \tau} \operatorname{Argmin} C(t_i)$$

Оба метода можно достаточно удачно использовать как в случае одной модели, так и при наличии ансамбля из нескольких моделей, но для ансамблирования зачастую больше подходит именно второй метод - он обнаруживает наиболее поздний пик, и при оценке получается остаток похожий на исходную траекторию без последнего пика, что идеально подходит для задачи оценки траектории ансамблем однотипных математических моделей.

Метод программно реализован на Python и оформлен в виде программного пакета (библиотеки). Проведено сравнение оригинального метода и модифицированного.

Как можно видеть, модифицированный метод справляется с поиском множественных пиков и проявляет себя лучше оригинального на «хвостах» оцениваемой траектории. Средняя квадратичная ошибка ансамблевого метода с



взвешенными ошибками - 164.2, в то время как оригинальная модель имеет почти вдвое худший показатель качества - примерно 321.6.

Полученный модифицированный метод позволяет не только более точно оценить форму кривой, но и перейти от сравнения исходных кривых к сравнению физически интерпретируемых показателей препарата (количество пиков, коэффициенты абсорбции и элиминации). Также это даёт возможность использовать другие методы анализа основанные на метрической близости объектов линейных пространств (например, кластерный анализ).

ЛИТЕРАТУРА

- [1] Macheras P., Chrysafidis P. Revising pharmacokinetics of oral drug absorption: I. Models based on biopharmaceutic, physiological, and finite absorption time concepts // 2020. P.137.
- [2] Chrysafidis P., Tsekouras A.A., Macheras P. Revising pharmacokinetics of oral drug absorption: II. Bioavailability–bioequivalence considerations // 2021. Pp. 1345–1356.
- [3] Chrysafidis P., Tsekouras A.A., Macheras P. Re-writing Oral Pharmacokinetics Using Physiologically Based Finite Time Pharmacokinetic (PBFTPk) Models // 2022. Pp. 691–701.

-
- [4] Dost F.H. Der Blutspiegel. Kinetik der Konzentrationsverläufe in der Kreislaufflüssigkeit // Dost F.H. 1953.
 - [5] Rowland M., Tozer T. N. Clinical Pharmacokinetics and Pharmacodynamics: Concepts and Applications. 4ed. — Philadelphia : Lippincott Williams & Wilkins, 2011. Pp. 1345–1356.

Разработка физических состязательных атак на нейросетевые модели путём обучения состязательных патчей

Работа удостоена диплома 1 степени
Леоненкова Виктория Николаевна

Кафедра интеллектуальных информационных технологий
e-mail: victoria.leonenkova@graphics.cs.msu.ru

Научный руководитель — к.ф.-м.н. с.н.с. Ватолин Дмитрий Сергеевич

Научный консультант — к.ф.-м.н. Анциферова Анастасия Всеволодовна

Современные модели оценки качества основаны на машинном обучении, демонстрируя более высокую корреляцию с субъективными оценками, чем классические методы. Однако исследования [1, 2] показывают, что такие модели уязвимы к состязательным атакам — преобразованиям, повышающим оценку качества модели. Поскольку модели глубокого обучения предоставляют доступ к градиентам, а модели оценки качества обычно общедоступны, злоумышленник может легко обучить состязательное возмущение и искусственно повысить оценку качества. Подобные атаки возможны в следующих сценариях в реальной жизни:

- мошенничество в бенчмарках;
- манипулирование результатами поиска;
- увеличение размера транскодированных видео в облачных хранилищах;
- воздействие на попадание видео-контента в рекомендации в социальных сетях.

В отличие от существующих исследований, фокусирующихся на атаках в пиксельном пространстве [1, 2], в данной работе впервые демонстрируется атака на безэталонные модели оценки качества (отображение $M : P \rightarrow \mathbb{R}$ из пространства изображений (видео) P в пространство вещественных чисел $\mathbb{R}$) в физическом пространстве с помощью состязательных патчей, которые обучены на повышение оценки качества видео, снятого на камеру.

В данной работе предлагается метод физической состязательной атаки на модели оценки качества видео или изображений, дополнительно рассматривался перенос атаки на модель распознавания лиц. Физическая состязательная атака на модель оценки качества видео представляет собой демонстрацию специально обученного патча перед камерой, что приводит к искусственному завышению показателей качества снимаемого видео, выдаваемых моделью, по сравнению с его реальным визуальным качеством. Формулировка атаки с использованием состязательного патча p на безэталонную модель оценки качества M выглядит следующим образом:

$$\operatorname{argmax}_p \left\{ \frac{1}{N} \sum_{k=1}^N (M(R(x_k, p)) - M(x_k)) \right\}, \quad \|p\|_0 \leq C \times D \times D, \quad (1)$$

где $R(x_k, p)$ – оператор случайного наложения патча:

$$R(x_k, p)_{i,j} = \begin{cases} & \text{если } i - x_s \geq 1, j - y_s \geq 1, \\ p_{i-x_s, j-y_s}, & i - x_s \leq D, j - y_s \leq D; \\ (x_k)_{i,j}, & \text{иначе.} \end{cases}$$

x_s, y_s — случайно выбранные координаты расположения патча в диапазонах $[1, W - D]$ и $[1, H - D]$ соответственно, p — искомый патч, $\{x_k\}_{k=1}^N$ — изображения из обучающего набора, N — размер обучающего набора, M — атакуемая модель оценки качества видео, D — ширина и высота патча.

Существует сложность в реализации физической атаки в отличие от цифровой, которая заключается в том, что физический патч дополнительно подвергается влиянию различных факторов: изменению освещения, перспективным искажениям и шумовым помехам.

Для снижения высокочастотных компонент в патче в процесс обучения добавлена минимизация полной вариации [3] (формула 2, где $p_{i,j}$ — пиксель патча p с координатами (i, j)).

$$TV(p) = \sum_{i,j} ((p_{i,j} - p_{i+1,j})^2 + (p_{i,j} - p_{i,j+1})^2)^{\frac{1}{2}}, \quad (2)$$

Для обеспечения корректного воспроизведения цветов патча при печати введена оптимизация показателя непечатаемости [4] (формула 3, где $T \subset [0, 1]^3$ — множество воспроизводимых при печати RGB-троек, $p_{i,j}$ — пиксель патча p с координатами (i, j)), минимизирующая разницу между цифровыми цветами патча и цветовым охватом принтера.

$$NPS(p_{i,j}) = \prod_{t \in T} |p_{i,j} - t|, \quad (3)$$

Для повышения устойчивости патча к изменениям освещения при съёмке видео в процесс обучения добавлено случайное варьирование яркости патча на каждой итерации. Этот подход позволяет сохранять эффективность атаки при различных условиях освещённости. Также была проверена эффективность чёрно-белых патчей. Для физических атак, где злоумышленник контролирует положение патча, было исключено случайное вращение из процесса обучения, что может повысить итоговую эффективность.

Таким образом, было предложено 7 различных методов (и один базовый) в результате комбинации предложенных добавок:

1. **База**: базовый метод.
2. **База+ПВ+ПН**: базовый метод с добавлением в оптимизацию функций потерь TV и NPS .
3. **БазаЯ**: базовый метод с добавлением случайного изменения яркости.
4. **БазаЯ+**: базовый метод с добавлением в оптимизацию функций потерь TV и NPS , а также дополнительным изменением яркости.

5. **Б-ЧБазаЯ+**: БазаЯ+ в чёрно-белом варианте.
6. **Б-ЧБазаБПЯ+**: БазаБПЯ+ в чёрно-белом варианте.
7. **БазаБПЯ+** (предложенный метод): базовый метод без случайного поворота патча с функциями потерь TV и NPS и дополнительным изменением яркости.
8. **БазаБП** (предложенный метод): базовый метод без случайного поворота патча при его наложении на изображение.

В качестве целевых моделей оценки качества для предложенных методов атак с помощью патчей были выбраны PaQ-2-PiQ с $M_{range}^1 = 100$, ARNIQA с $M_{range}^2 = 10$, Hyper-IQA с $M_{range}^3 = 10$, безэталонный вариант TOPiQ с $M_{range}^4 = 10$, TReS с $M_{range}^5 = 100$, чтобы покрыть различные архитектуры. Для обучения использовано 1000 изображений 256×256 из COCO, для проверки - 250 изображений того же размера. Тестирование проведено на 500 изображениях из COCO, 1000 изображениях 299×299 из NIPS 2017 и 12 видео 640×360 (по 75 кадров) из DERF 2001. Успешность предложенной атаки для одного изображения измеряется как разность значений модели оценки качества видео и изображений до и после атаки. Формально это выражается следующим образом:

$$gain = M(x_i^p) - M(x_i), \quad (4)$$

где M — целевая модель оценки качества, x_i^p — искажённое изображение, а x_i — исходное изображение.

Эксперименты в пиксельном пространстве проводились в два этапа: сначала все методы тестировались на модели PaQ-2-PiQ, затем лучшие из них проверялись на четырёх других моделях. В Таблице 1 представлены результаты экспериментов атаки на модель оценки качества PaQ-2-PiQ в пиксельном пространстве. Можно сделать вывод, что патч, обученный на одном наборе данных, хорошо переносится на неизвестные наборы данных с другим содержанием и разрешением. Два лучших метода были протестированы на других четырёх моделях оценки качества. По результатам этого эксперимента минимальный прирост среди всех моделей был 18%, максимальный — 87%.

Физические эксперименты включали три этапа:

1. **Тестирование одиночных патчей** (8 вариантов) в 3 локациях при разных расстояниях (22–28 см) с использованием специальной гирлянды из подряд приклеенных на прозрачную ленту патчей для стандартизации условий. Прирост качества оценивался по максимальному значению модели при прохождении патча через кадр.
2. **Патчи-мозаики** (4×6 патчей) сравнивались с одиночными вариантами в тех же условиях для оценки влияния площади покрытия.
3. **Состязательные обои** тестировались на больших расстояниях (125–206 см) с человеком перед экраном с размноженными по нему патчами для имитации реального сценария использования.

Метод	COCO		NIPS	DERF
	Обучение 1000 изобр.	Тестирование 500 изобр.	1000 изобр.	12 видео
База	17.44 ± 0.25	16.07 ± 0.35	15.23 ± 0.20	16.49 ± 0.26
База+ПВ+ПН	15.44 ± 0.24	15.50 ± 0.31	13.50 ± 0.20	14.56 ± 0.22
БазаЯ	16.36 ± 0.25	16.46 ± 0.35	14.25 ± 0.19	15.43 ± 0.26
БазаЯ+	16.39 ± 0.25	16.39 ± 0.34	14.22 ± 0.19	15.55 ± 0.25
Б-ЧБазаЯ+	9.88 ± 0.22	9.88 ± 0.30	8.55 ± 0.16	10.03 ± 0.22
Б-ЧБазаБПЯ+	18.81 ± 0.24	18.93 ± 0.30	17.23 ± 0.17	18.09 ± 0.23
БазаБПЯ+	29.11 ± 0.29	28.85 ± 0.37	26.30 ± 0.20	27.47 ± 0.27
БазаБП	30.18 ± 0.28	30.19 ± 0.39	27.29 ± 0.22	27.86 ± 0.25

Таблица 7: Средний прирост значений модели оценки качества PaQ-2-PiQ после атаки с помощью патчей в пиксельном пространстве для 8 методов и 3 наборов данных.

Анализ результатов эксперимента тестирования одиночных патчей выявил несколько ключевых закономерностей. Патч БазаБПЯ+ продемонстрировал наибольший прирост показателей модели оценки качества, подтверждая эффективность предложенных модификаций. Оптимальное расстояние съёмки составило 25 см с рабочим диапазоном 22-28 см, где вариации масштабирования патча минимально влияют на результат. Чёрно-белые патчи показали сниженную эффективность по сравнению с цветными аналогами, что свидетельствует о важности цветовых характеристик для успешной атаки. Согласно результатам экспериментов с одиночными патчами был выбран лучший патч — БазаБПЯ+ и проведён эксперимент с размножением патча. Как и предполагалось, размноженная версия патча — патч-мозаика — даёт больший прирост модели оценки качества, чем одиночный патч. Аналогично для эксперимента состязательными обоями был выбран метод БазаБПЯ+. Сгенерированный патч позволил достичь прироста значений модели оценки качества в 18-20 пунктов на каждом расстоянии, что является наилучшим результатом среди всех физических настроек.

Для проверки практической применимости метода была проведена атака на коммерческую систему распознавания лиц (атака методом чёрного ящика). Использован чёрно-белый вариант патча-мозаики (44×66), демонстрировавшийся с планшета. Обучающая выборка включала 10 аугментированных изображений тестируемого человека. Эксперимент с коммерческой системой распознавания лиц наглядно продемонстрировал эффективность разработанного метода. В ходе тестирования был реализован следующий сценарий: когда неавторизованный человек приближался к камере, система за 3 секунды принимала решение о запрете доступа. Однако при демонстрации под лицом человека планшета с чёрно-белым размноженным патчем реакция системы кардинально менялась — время принятия решения увеличивалось до 41 секунды (время до момента выключения планшета с патчами), что свидетельствует о существенном нарушении работы алгоритма распознавания. После

выключения планшета камера практически мгновенно (в течение 1 секунды) возвращалась к исходному режиму работы, повторно запрещая доступ.

Таким образом, основные результаты работы можно подытожить следующим образом:

1. Проведён обзор существующих методов атак и выявлена их ограниченность пиксельным пространством.
2. Предложено 7 модификаций базового метода атаки на модели оценки качества, с помощью которых увеличена эффективность атаки в физическом пространстве.
3. Впервые атакована безэталонная модель оценки качества видео в физическом мире и выявлена уязвимость 5 моделей оценки качества в пиксельном пространстве.
4. Реализованы физические состязательные обои и представлен реалистичный сценарий атаки на модель оценки качества изображений и видео PaQ-2-PiQ с приростом значений модели на 20 пунктов (20% диапазона модели).
5. Перенесен результат физической атаки на коммерческую камеру распознавания лиц и достигнут эффект зависания камеры.

ЛИТЕРАТУРА

- [1] Perceptual Attacks of No-Reference Image Quality Models with Human-in-the-Loop / W. Zhang, D. Li, X. Min, G. Zhai, G. Guo, X. Yang, K. Ma // Proceedings of the conference on Neural Information Processing Systems (NeurIPS). 2022.
- [2] Universal Perturbation Attack on Differentiable No-Reference Image- and Video-Quality Metrics / E. Shumitskaya, A. Antsiferova, D. Vatolin // British Machine Vision Conference (BMVC). 2022
- [3] Mahendran A., Vedaldi A. Understanding Deep Image Representations by Inverting Them // Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). 2015. С. 5188-5196
- [4] Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition / M. Sharif, S. Bhagavatula, L. Bauer, M. K. Reiter // Proceedings of the acm sigsac conference on computer and communications security. 2016. С. 1528-1540.

Автоматическое распознавание русских жестовых слов

Ли Сынгу

Кафедра алгоритмических языков

e-mail: nugejus@mail.ru

Научный руководитель — к.ф.-м.н. доц. Полякова Ирина Николаевна

Введение. Существующие подходы к задаче распознавания жестовой речи можно условно разделить на сенсорные и визуальные. Сенсорные методы предполагают использование специализированного оборудования, фиксирующего направление руки, движения предплечья и другие параметры [1]. Визуальные методы, напротив, основываются исключительно на анализе изображений, полученных с камеры. В таких подходах применяются искусственные нейронные сети, для обучения которых требуются большие объёмы данных. Настоящая работа ориентирована на использование визуального подхода для решения задачи распознавания русской жестовой речи.

Наборы данных. В мире не существует единого универсального стандарта жестового языка, поэтому для каждого языка требуется разработка отдельных моделей. В настоящей работе рассматривается задача распознавания изолированных жестов на основе широко используемого датасета WLASL [2], а также крупномасштабного датасета SLOVO, представленного Александром Капитановым и исследовательской группой Сбера в 2023 году [3].

Видеофайлы в датасете WLASL представлены в различных форматах — mp4 и swf, тогда как в датасете SLOVO используются исключительно файлы формата mp4. Однако прямое использование видеороликов в качестве входных данных оказалось неэффективным. Также для корректного моделирования необходимо унифицировать формат входных данных всех кадров. Это особенно важно, поскольку большинство предварительно обученных открытых моделей (например, ViViT) требуют фиксированного формата входа.

В данном исследовании из полного набора в 1001 класс были отобраны 100 наиболее часто употребляемых слов для формирования поднабора SLOVO100 для построения итоговой модели. Этот поднабор включает такие слова, как «есть», «сказать», «если», «делать», «идти», «знать» и другие.

Аугментация данных. В датасете WLASL большинство классов содержит менее 10 видеозаписей, а в случае с датасетом SLOVO, несмотря на равномерное распределение, общее количество данных на класс остаётся крайне ограниченным. Для решения данной проблемы применяется аугментация данных. Используемые методы включают классические техники, такие как поворот (rotation), размытие (blur), изменение цветовой гаммы (color jitter), которые широко используются в задачах классификации изображений.

На конференции «Тихоновские чтения — 2023» был представлен доклад «Повышение эффективности дополнения данных для малых выборок» [4], в котором была проведена серия экспериментов на ограниченном подмножестве датасета WLASL. Целью исследования было определение эффективно-

сти методов аугментации изображений в задачах распознавания человеческих жестов, в частности жестового языка. Результаты показали, что увеличение объема обучающей выборки за счёт аугментации способствует росту точности (Accuracy), а также улучшает стабильность сходимости функции потерь (Loss). Максимальное значение точности (Top-1 точность) увеличилось в 1,7 раза.

Метод временных сегментов. Видеодатасеты по своей природе характеризуются большим объёмом данных и требуют значительных вычислительных ресурсов, в частности большого объёма памяти GPU для обработки. При этом для эффективного моделирования зачастую достаточно лишь небольшой части информативных кадров из всего видеоролика. Остальные кадры, как правило, являются дублирующими, что может способствовать быстрому переобучению модели. Для решения данной проблемы применяется стратегия выборки кадров, известная как метод временных сегментов (Temporal Segment Networks, TSN) [5].

Идея данной стратегии заключается в разделении видео на несколько равных по длительности сегментов и случайном выборе одного кадра из каждого сегмента. На конференции «Ломоносовские чтения — 2024» в докладе «Повышение качества модели распознавания жестовых слов с использованием временных сегментов» [6] была представлена оценка эффективности данного подхода на ограниченном подмножестве датасета WLASL. По результатам экспериментов было установлено, что время обучения сократилось в 2,173 раза. Эти результаты подтверждают эффективность метода TSN в оптимизации использования вычислительных ресурсов.

Моделирование. Как уже упоминалось, в настоящем исследовании применяется визуальный подход к задаче распознавания жестов. Такой подход традиционно основывается на использовании искусственных нейронных сетей и во многом аналогичен задачам распознавания видеоконтента. Наиболее распространёнными архитектурами в данной области являются 3D CNN, CNN-LSTM и Transformer.

Архитектура 3D CNN демонстрирует высокую эффективность при извлечении временных признаков, однако её существенным недостатком является ограниченность по временной оси. Эта особенность делает её менее пригодной для моделирования языковых систем, в которых критически важна обработка непрерывной последовательности информации. В связи с этим настоящее исследование сосредоточено на использовании архитектур CNN-LSTM и Transformer, обладающих большей гибкостью при работе с временными зависимостями.

CNN-LSTM. В задачах распознавания видеоконтента архитектура CNN-LSTM предполагает извлечение признаков с использованием двумерной сверточной нейронной сети (2D CNN), после чего полученные признаки обрабатываются модулем LSTM. Однако в случае распознавания жестов требуется применение более глубокой архитектуры сверточной сети. Как правило, уве-

личение глубины слоёв способствует повышению точности модели, однако, как показано в ряде предыдущих исследований, при превышении определённого порогового значения глубины наблюдается снижение производительности [7].

Для преодоления данной проблемы, как правило, применяются предварительно обученные (pretrained) глубокие сверточные нейронные сети. В рамках данного эксперимента были протестированы различные конфигурации моделей семейства ResNet. Каждая из них обучалась в течение 50 эпох на малом подмножестве датасета WLASL. Наилучшие результаты показали модели ResNet18 и ResNet34. С учётом потребления вычислительных ресурсов, а также с позиции углеродной нейтральности, предпочтение было отдано модели ResNet18.

В результате обучения нейронной сети с использованием описанной архитектуры была достигнута Top-1 точность, составляющая 89

Transformer. Трансформеры являются одной из наиболее современных инноваций в области машинного обучения и считаются передовыми инструментами для моделирования последовательностей. В данном разделе рассматривается архитектура Video Vision Transformer [8]. В целом, архитектура Video Vision Transformer во многом схожа с Vision Transformer (ViT), однако отличается способом преобразования входного видеопотока в токены, а также подходами к факторизации размерностей.

В настоящем исследовании использовались открытые модели, размещённые на платформе Hugging Face. Эти модели основаны на архитектурах TimeSFormer, Xclip и VideoMAE и предварительно обучены на наборах данных жестового языка таких как ASL(американский), SSL(испанский), ISL(индейский) и т.д.. Из общего числа были выбраны 12 моделей с высокими показателями точности и тегом «sign-language». Их производительность оценивалась с использованием трансферного обучения на подмножестве SLOVO10, случайным образом сформированном из 10 слов основного датасета SLOVO.

Наилучший результат продемонстрировала модель VideoMAE-base, размещённая пользователем sokaina55. Полученные результаты свидетельствуют о высокой способности данной базовой модели эффективно извлекать признаки из данных, схожих по характеру с задачей распознавания человеческих жестов.

Детали обучения. В рамках данного исследования в качестве оптимизатора использовался AdamW, а в качестве планировщика скорости обучения — алгоритм CosineAnnealingWarmRestarts. Обучение модели проводилось в среде Google Colab с использованием одного графического процессора L4 с 22,5 ГБ оперативной памяти. В качестве функции потерь применялась CrossEntropyLoss.

Таблица 8: Сравнение с другими моделями с набором данных SLOVO

Модель	Параметры	Датасет	Устройство	Топ-1 точность(%)
MViTv2-S	52M	Slovo	four 80GB RAM H100 GPU	77.54
MViTv2-B	217M			79.31
I3D	12M			62.79
MViTv2-S	52M	SlovoExt		83.36
MViTv2-B	217M			84.14
I3D	12M			77.30
Ours	87M	Slovo100	GoogleColab 22.5GB L4 GPU	74.45

Остальные гиперпараметры — такие как размер батча, количество сегментов и число кадров в каждом сегменте — подбирались эмпирически в зависимости от конкретной архитектуры модели.

Оценка модели. Для оценки модели использовалась метрика Топ-1 точность, которая позволяет проводить сопоставление полученных результатов с данными других исследований в области классификации жестовых слов. По результатам обучения значение Топ-1 точность на тестовом наборе данных составило 74,45%.

Сравнение с другими моделями. В 2024 году Карина Кванчиани и её соавторы в статье «Training Strategies for Isolated Sign Language Recognition» обучили датасет SLOVO с использованием архитектур видеотрансформеров, включая MViT. Значения Топ-1 точность для различных методов, предложенных в данном исследовании, представлены в таблице 8 [9] (см. Таблица 8).

Как видно из таблицы, между настоящим исследованием и предыдущими работами имеются различия в количестве классов, что затрудняет прямое сопоставление и объективную оценку моделей. Для устранения этого ограничения используется коэффициент Тау, предложенный в 1995 году Ma Z.K. и R.L. Redmond [10]. $\tau = \frac{p_o - p_e}{1 - p_e}$, где p_o — это общая точность (overall accuracy), а $p_e = \sum_{c=1}^C p_c^{ref} p_c^{map}$ — произведение распределений истинных и предсказанных значений. Эта метрика эквивалентна показателю chance-adjusted accuracy (NTA) и входит в семейство метрик, таких как Balanced Accuracy и коэффициент Каппа Коэна.

Применяя данную метрику, можно более объективно сопоставить производительность нашей модели с альтернативными решениями:

$$\begin{aligned}\tau_{ViViT(ours)} &= \frac{0.7445 - 0.01}{0.99} = 0.742 \\ \tau_{MViTv2-S} &= \frac{0.8336 - 0.001}{0.999} = 0.833 \\ \tau_{MViTv2-B} &= 0.841 \\ \tau_{I3D-RGB} &= 0.772\end{aligned}$$

Как следует из приведённых расчётов, лучшая из предшествующих моделей превосходит нашу примерно в 1,13 раза. Тем не менее, с учётом различий в вычислительных ресурсах (4×H100 против 1×L4) и числе параметров

моделей (217 млн против 87 млн), наша модель демонстрирует конкурентоспособность с точки зрения соотношения затрат и эффективности.

Кроме того, даже при использовании Google Colab с GPU L4 достигается производительность в режиме реального времени (до 45 кадров в секунду) при задержке менее 25 мс. С точки зрения углеродной нейтральности энергопотребление модели составляет около 0,05 кВт·ч на одну эпоху обучения, что делает её значительно более энергоэффективной по сравнению с моделями, обученными на кластерных системах с GPU H100.

Заключение. Таким образом, были получены следующие основные результаты.

- Применены аугментация и техника метод временных сегментов (Temporal Segment Network) для «больших, но разреженных» видеодатасетов.
- На основе анализа 3D-CNN, CNN-LSTM и Video Vision Transformer (ViViT) выбраны подходящие архитектуры для построения модели распознавания жестовых слов.
- Проведены сравнения весов предобученных моделей, оптимизаторов и LR-планировщиков, и найдены настройки, обеспечивающие устойчивое улучшение точности на малых подмножественных выборках.
- Обучена модель на подмножестве из 100 наиболее частотных слов русского корпуса SLOVO100, которая показала Top-1 точность 74,45%

Результаты исследования были представлены на научных конференциях «Тихоновские чтения» и «Ломоносовские чтения», Международной конференции по науке и технологиям «Россия-Корея-СНГ» в 2023г. и 2024г., Молодежной конференции АНТОК 2024, а также приняты к публикации в научном журнале «Современные информационные технологии и ИТ-образование» (2025 год, том 21, номер 1).

ЛИТЕРАТУРА

- [1] K. Kudrinko, E. Flavin, X. Zhu, Q. Li Wearable Sensor-Based Sign Language Recognition: A Comprehensive Review // in IEEE Reviews in Biomedical Engineering, vol. 14, pp. 82-97, 2021, doi: 10.1109/RBME.2020.3019769.
- [2] Li D., Rodriguez C., Yu X., Li H. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. // In Proceedings of the IEEE/CVF winter conference on applications of computer vision (pp. 1459-1469).
- [3] Kapitanov A., Karina K., Nagaev A., Elizaveta P. Slovo: Russian sign language dataset. // In International Conference on Computer Vision Systems (pp. 63-73). Cham: Springer Nature Switzerland. (2023, September).
- [4] Ли С.; Полякова И.Н. Разработка эффективности дополнения данных для малого датасета. // В сб.: Тихоновские чтения. МАКС-Пресс, Москва, 2023, с. 108.

- [5] Wang L., Xiong Y., Wang Z., Qiao Y., Lin D., Tang X., Van Gool L. Temporal segment networks: Towards good practices for deep action recognition. // In European conference on computer vision (pp. 20-36). Cham: Springer International Publishing.
- [6] Ли С., Полякова И. Н. Повышение качества модели распознавания жестовых слов с использованием временных сегментов. // В сб.: Ломоносовские чтения-2024. МАКС-Пресс, Москва, 2024, с. 181–183.
- [7] Glorot X., Bengio Y. Understanding the difficulty of training deep feedforward neural networks. // In Proceedings of the thirteenth international conference on artificial intelligence and statistics (pp. 249-256). JMLR Workshop and Conference Proceedings.
- [8] Arnab A., Dehghani M., Heigold G., Sun C., Lučić M., Schmid C. Vivit: A video vision transformer. // In Proceedings of the IEEE/CVF international conference on computer vision (pp. 6836-6846).
- [9] Kvanchiani K., Kraynov R., Petrova E., Surovcev P., Nagaev A., Kapitanov A. Training Strategies for Isolated Sign Language Recognition. // arXiv preprint arXiv:2412.11553.
- [10] Ma Z., Redmond R.L. (1995). Tau coefficients for accuracy assessment of classification of remote sensing data. // Photogrammetric Engineering and Remote Sensing, 61(4), 435-439.

Исследование методов определения границ сегментов потока

Работа удостоена диплома 3 степени

Лысцов Илья Андреевич

Кафедра автоматизации систем вычислительных комплексов

e-mail: s02210455@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. асс. Степанов Евгений Павлович

Введение. В современных сетях ЭВМ балансировка трафика является важной задачей для обеспечения высокой пропускной способности, низких задержек и надежности доставки пакетов. Существующие подходы к балансировке включают балансировку по пакетам, по потокам и по сегментам потока. Пакетная балансировка может привести к нарушению порядка пакетов, из-за чего отправитель может уменьшить своё окно перегрузки, и скорость отправки данных будет снижена. Балансировка по потокам сохраняет порядок пакетов, но может привести к неравномерному распределению нагрузки. Для устранения недостатков обоих подходов была введена концепция «сегмента потока».

Сегментом потока называется последовательность таких пакетов данного потока, что интервал времени между соседними пакетами меньше заранее заданной величины δ . При балансировке по сегментам потока пакеты одного сегмента направляются на один выходной порт, что позволяет избежать перепорядочивания пакетов при сохранении возможностей для эффективной балансировки. Значение δ может быть как фиксированным, так и изменяющимся в зависимости от потока и времени.

Для использования методов балансировки по сегментам потока необходим алгоритм, который позволял бы определить, принадлежит ли поступивший пакет к уже существующему сегменту потока, или же он инициирует новый сегмент. Простая реализация требует хранения времени прихода последнего пакета для каждого потока, что ограничивает масштабируемость при увеличении числа потоков в условиях ограниченной памяти на сетевых устройствах.

Математическая постановка задачи. Пусть $E = \{e_1, e_2, e_3, \dots\}$ — множество поступающих пакетов на сетевое устройство. Каждый пакет принадлежит потоку, задаваемому отображением $u : E \rightarrow U$, где $U = \{u_1, u_2, u_3, \dots\}$ — множество всех потоков. Обозначим $u(e_i)$ — поток пакета e_i , $t(e_i) \in \mathbb{R}_+$ — время поступления пакета e_i . Каждому потоку в каждый момент времени сопоставляется длина временного интервала $\delta(u, t) : U \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$.

Требуется найти пару (S, Φ_S) , где S — структура данных для хранения информации о недавно поступивших пакетах, $\Phi_S(e_i, t, \delta) : E \times \mathbb{R}_+ \times \mathbb{R}_+ \rightarrow \{0, 1\}$ — функция, определяющая поступал ли за последние δ секунд пакет из потока $u(e_i)$. При этом должны выполняться условия:

— Объём памяти $M(S)$ не превышает заданного значения M_0

- Вероятность ошибки первого рода P_{fp} минимизируется
- Ошибки второго рода отсутствуют

Обзор методов. В работе рассмотрены пять методов, применимых для решения поставленной задачи: HoppingTimer [1], Time-based Age-Partitioned Bloom Filter (APBF) [2], Sliding Sketches [3], Timing Bloom Filters (TBF) [4] и MultiBloomFilter (MBF) [5]. В основе всех методов лежит использование фильтра Блума — вероятностной структуры данных, позволяющей эффективно по памяти определять принадлежность элемента множеству.

Анализ методов проводился по следующим критериям: временная сложность основных операций (вставка элемента, запрос элемента, удаление устаревших записей) и поддержка разных временных интервалов для разных потоков.

Проведенный обзор показал, что только метод MBF изначально поддерживает различные временные интервалы для разных потоков. Остальные четыре метода требуют адаптации для решения поставленной задачи.

Адаптация методов. Для методов HoppingTimer, APBF и TBF предложены модификации, позволяющие работать с переменными значениями $\delta(u, t)$. Основная идея заключается в использовании $\delta_{max} = \max_{u \in U, t \in \mathbb{R}_+} \delta(u, t)$ для операций удаления устаревших записей и индивидуальных значений $\delta(u, t)$ для операций запроса.

Метод Sliding Sketches нельзя адаптировать: значительные различия в значениях δ между потоками неизбежно увеличивают число ложноположительных результатов.

Реализация и экспериментальное исследование. Разработана программная реализация, представляющая собой дискретно-событийную имитационную модель компонента маршрутизатора, отвечающего за определение границ сегментов потока. Для хеширования используется алгоритм SHA-256.

Экспериментальное исследование проводилось на реальных сетевых трассах. Использовались два фрагмента трассы с различными характеристиками: первый содержит около 50000 пакетов, относящихся к 6500 потокам со средней скоростью 1500 пакетов в секунду; второй — 50000 пакетов 771 потока со скоростью 15500 пакетов в секунду. Значения δ для каждого потока выбирались из равномерного распределения в диапазоне от 0.01 до 0.1 секунды и обновлялись каждую секунду.

Методика исследования включала два этапа: анализ зависимости доли ложноположительных результатов от объема используемой памяти и анализ времени выполнения операций при различных уровнях точности.

Результаты. При анализе точности алгоритм APBF продемонстрировал наименьшую долю ложноположительных результатов при ограниченных объемах памяти, достигая нулевых значений уже при 0.75 КБ на первом фрагменте трассы и 0.5 КБ на втором. Алгоритмы HoppingTimer и TBF показали схожее поведение, но требовали больше памяти для достижения аналогичной

точности. MBF демонстрировал наиболее высокие значения доли ложноположительных результатов на всем диапазоне исследуемых объемов памяти.

Анализ производительности показал существенные различия между методами. TBF имеет стабильно низкое время выполнения операций вставки и запроса (5–20 микросекунд), что делает его эффективным в условиях интенсивного поступления пакетов. APBF демонстрирует значительные временные затраты на операцию вставки (800–1100 микросекунд), но обеспечивает наилучшую точность. Важно отметить, что производительность APBF существенно зависит от количества уникальных потоков в трафике: на втором фрагменте с меньшим числом потоков время операций сократилось в несколько раз.

Заключение. В работе составлена математическая постановка задачи обнаружения границ сегментов потока, проведен обзор пяти применимых методов и выявлено, что только MBF изначально поддерживает работу с различными временными интервалами. Предложены модификации для адаптации методов HoppingTimer, APBF и TBF под требования задачи.

Экспериментальное исследование на реальных сетевых трассах показало, что APBF является методом, достигающим наилучшей точности при заданных ограничениях по памяти, а TBF демонстрирует наилучшую производительность операций. Выбор между этими методами зависит от конкретных требований: при жестких ограничениях по памяти и необходимости минимизации ошибок следует использовать APBF, при высокой интенсивности трафика и критичности времени обработки — TBF.

ЛИТЕРАТУРА

- [1] Yang K., Wu J., Yang T. HoppingTimer: A near-optimal framework for basic estimation of data streams in hopping windows // Proceedings of the 2024 International Scientific and Technical Conference Modern Computer Network Technologies (MoNeTeC). 2024. P. 1–13.
- [2] Rodrigues A. et al. Time-limited Bloom Filter // Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing. 2023. P. 1285–1288.
- [3] Gou X. et al. Sliding sketches: A framework using time zones for data stream processing in sliding windows // Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020. P. 1015–1025.
- [4] Zhang L., Guan Y. Detecting click fraud in pay-per-click streams of online advertising networks // 2008 The 28th International Conference on Distributed Computing Systems. IEEE, 2008. P. 77–84.
- [5] Лысцов И. А. Динамическое определение сегментов потока : курс. раб. Москва : Московский гос. ун-т, 2024. 24 с.

Анализ фонетической интерференции в китайско-русской речи с использованием алгоритмов распознавания речи

Работа удостоена диплома 2 степени

Максимова Яна Дмитриевна

Кафедра математических методов прогнозирования

e-mail: yana.d.maximova@gmail.com

Научный руководитель — к.ф.-м.н. доц. Гуров Сергей Исаевич

Актуальной задачей современной фонетики является анализ фонетической интерференции в речи, в частности, в контексте изучения иностранного языка. Одной из наиболее распространённых групп обучающихся русскому языку являются носители китайского языка, для которых характерны фонетические трудности, обусловленные различиями в фонологической системе. В частности, отсутствие некоторых согласных и гласных звуков, присущих русскому языку, часто приводит к искажению звучания и потере смысла произнесённых слов. И с развитием технологий машинного обучения возникает возможность в автоматизации процесса выявления таких ошибок.

Цель данной работы является определение правильности произношения русских слов, выполненное носителями русского и китайского языка, с применением алгоритмов распознавания речи. В качестве основного критерия эффективности рассматривается способность алгоритмов различать корректные и некорректные произношения с высокой точностью, а также устойчивость моделей к вариативности речи. Для достижения этой цели решаются две взаимосвязанные подзадачи: классификация произнесённых слов на «правильные» и «неправильные» с точки зрения соответствия фонетическим нормам русского языка и распознавание произнесённого слова из заданного ограниченного словаря.

Исследование выполнено в рамках межфакультетского проекта, реализуемого совместно факультетом вычислительной математики и кибернетики, а также психологическим и филологическим факультетами, под названием «Механизмы формирования китайско-русской фонетической интерференции».

Данные. Используемые данные, были сформированы в ходе специально организованного эксперимента, проведённого на базе филологического, психологического факультетов и факультета вычислительной математики и кибернетики МГУ имени М. В. Ломоносова. В эксперименте приняли участие 11 респондентов: 5 носителей русского языка и 6 носителей китайского языка, обучающихся русскому как иностранному.

Лингвистами был сформирован целевой словарь, включающий порядка 1500 слов. В словарь вошли слова, представляющие наибольшую артикуляционную сложность, в том числе слова с твёрдыми и мягкими согласными,

редуцированными гласными, а также слова с ударением в нестандартной позиции (например, чертыхаться).

Ввиду ограниченности исходного набора данных было принято решение об увеличении обучающей выборки с помощью методов аугментации. Для этого использовалась библиотека *librosa*, предоставляющая инструменты обработки аудиосигналов в Python. Были применены следующие виды аугментации:

- Изменение громкости — коэффициент усиления выбирался случайным образом из диапазона от 0,5 до 1,5;
- Изменение высоты тона — количество полутонов для смещения варьировалось случайным образом в пределах от -5 до $+5$;
- Изменение скорости воспроизведения — коэффициент изменения темпа выбирался случайным образом из диапазона от 0,8 до 1,25.

Параметры каждой аугментации варьировались случайным образом в указанных пределах, что позволило создать более разнообразные и реалистичные искажения аудиосигналов. Это позволило увеличить объём обучающих данных в три раза. Благодаря аугментации, повысилась устойчивость моделей к вариациям в произнесении, что позволило компенсировать дефицит обучающей выборки.

Классификация корректности произношения слов. Для реализации подзадачи классификации правильности произнесенных слов было выбрано два алгоритма. Один из них — алгоритм динамического выравнивания по времени (Dynamic Time Warping, DTW) [1] представляет собой метод измерения схожести между двумя временными последовательностями, возможно, различающимися по длине. Основное преимущество данного метода заключается в возможности построения оптимального соответствия между такими последовательностями.

Рассматриваются две временные последовательности:

$$Q = (q_1, q_2, \dots, q_n), \quad C = (c_1, c_2, \dots, c_m),$$

где Q и C имеют длины n и m соответственно. Для сравнения этих последовательностей строится матрица размерности $n \times m$, в которой элемент (i, j) содержит локальное расстояние $d(q_i, c_j)$, рассчитываемое, как правило, по евклидовой метрике.

На основе данной матрицы с использованием динамического программирования рассчитывается матрица накопленных расстояний $g(i, j)$, определяемая следующим рекуррентным соотношением:

$$g(i, j) = d(q_i, c_j) + \min \{g(i-1, j-1), g(i-1, j), g(i, j-1)\}.$$

Оптимальная траектория выравнивания w представляет собой последовательность пар индексов, удовлетворяющих следующим условиям:

- граничное условие: $w_1 = (1, 1)$ и $w_k = (m, n)$: требуемая траектория выравнивания такова, чтобы она начиналась первым и заканчивалась последним элементом главной диагонали,

- непрерывность: заданные $w_k = (a, b)$ и $w_{k-1} = (a', b')$, где $a - a' \leq 1$ и $b - b' \leq 1$. Что является ограничением разрешенной траектории выравнивания для соседних элементов,
- монотонность: заданные $w_k = (a, b)$ и $w_{k-1} = (a', b')$, где $a - a' \geq 0$ и $b - b' \geq 0$. Это требование монотонного увеличения во времени точек W .

На основе DTW был придуман алгоритм, где для каждого слова проводилось обучение индивидуальной модели на основе акустических параметров его реализаций. На первом этапе из всего набора дикторов выделялась тестовая пара: один диктор — носитель русского языка, другой — носитель китайского языка. Эти дикторы исключались из обучающей выборки, а все их произнесения данного слова использовались исключительно на этапе тестирования. Остальные дикторы формировали обучающую выборку.

Реализации слова в обучающей выборке разделялись согласно аннотациям на две группы: корректные (метка 0) и некорректные (метка 1). Для каждой реализации извлекались акустические признаки — в эксперименте использовались мел-спектрограммы и коэффициенты MFCC. Таким образом, формировались два множества матриц признаков, представляющие собой эталонные модели корректного и некорректного произнесения данного слова. Это позволило задачу бинарной классификации интерпретировать как задачу измерения степени схожести новых реализаций с этими двумя моделями.

На этапе тестирования каждая реализация слова из тестовой пары дикторов сравнивалась с обученными моделями корректных и некорректных произнесений при помощи алгоритма динамического программирования (DTW). Для каждой тестовой реализации вычислялись два набора расстояний: до всех реализаций корректного класса и до всех реализаций некорректного класса. В рамках эксперимента рассматривались различные стратегии агрегирования этих расстояний: минимальное, среднее и другие варианты. Наилучшей показала себя стратегия усреднения расстояний, которая использовалась в финальной версии модели.

Для принятия решения о корректности произнесения использовался эмпирически подобранный порог. Если среднее расстояние от тестовой реализации до множества корректных реализаций оказывалось меньше порогового значения, произнесение классифицировалось как корректное. Такая схема классификации обусловлена особенностями разметки: в используемом датасете метка 0 соответствовала корректному произнесению, а метка 1 — наличию фонетических ошибок. Это требовало обратной интерпретации расстояний: меньшая дистанция должна была соответствовать метке 0.

В качестве второго алгоритма для задачи классификации корректности произношения использовалась модель `wav2vec2-large-xlsr-53-russian`, основанная на архитектуре `wav2vec 2.0` [2]. Модель принимает на вход аудиосигнал x и извлекает из него латентные признаки z_t с помощью сверточного энкодера. Далее применяется трансформер, формирующий контекстные

представления c_t . Поверх последнего слоя модели добавлялся классификатор:

$$\hat{y} = \sigma(Wc_t + b)$$

где σ — сигмоида, W и b — параметры линейного слоя. Обучение проводилось с использованием бинарной кросс-энтропии.

Метод	Accuracy	F1-score
Mel	0.71	0.39
MFCC	0.83	0.64

Табл. 1: Результаты классификации с использованием DTW.

Подход	Accuracy	F1-score
Дообучение последнего слоя	0.79	0.6
Дообучение всей модели	0.82	0.62

Табл. 2: Результаты классификации с использованием wav2vec

Распознавание произнесенных слов Для задачи распознавания речи применялась та же архитектура wav2vec 2.0, дополненная СТС-декодером (Connectionist Temporal Classification) [3]. Модель предсказывает вероятностное распределение по символам в каждом временном шаге на основе контекстных представлений:

$$P(\pi|x) = \prod_t P(\pi_t|c_t)$$

где π — возможная последовательность выравнивания, c_t — контекстный вектор на шаге t . Финальный текст извлекается с помощью жадного СТС-декодирования.

Поскольку в данных содержались только написания слов на русском языке без фонетических транскрипций, было принято решение оценивать качество распознавания не только по точному совпадению, но и с использованием расстояния Левенштейн, которое показывает минимальное количество операций вставки, удаления или замены символов, необходимых для преобразования одной строки в другую. Таким образом, данная метрика позволяет учитывать незначительные отклонения в распознанном слове.

Результаты исследования.

— нейросетевые методы обеспечивают значительное качества распознавания речи и могут эффективно обрабатывать широкий спектр задач, включая определение правильности произношения

Критерий	Только линейный слой	Вся модель
Левенштейн = 0	0.8	0.85
Левенштейн ≤ 1	0.85	0.93
Левенштейн ≤ 2	0.88	0.94
Левенштейн ≤ 3	0.91	0.96

Табл. 3: Точность распознавания при разных порогах расстояния Левенштейна

— классические методы могут эффективно применяться в ряде специфических задач. Особенно это касается ситуаций, когда известно, какое именно слово будет произнесено, например, в образовательных системах для изучения иностранного языка

В дальнейшем сочетание этих методов может привести к созданию более гибких и эффективных систем для распознавания и анализа речи, особенно в образовательных и специализированных приложениях.

ЛИТЕРАТУРА

- [1] Rabiner L., Juang B. Fundamentals of speech recognition. Englewood Cliffs : Prentice-Hall, 1993. 499 p.
- [2] Wav2vec 2.0: a framework for self-supervised learning of speech representations / A. Baevski, H. Zhou, A. Mohamed, M. Auli // Advances in neural information processing systems. 2020. Vol. 33. P. 12459–12478.
- [3] Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks / A. Graves, S. Fernandez, M. Schuster, Y. Bengio // Proceedings of the 23rd international conference on machine learning (ICML 2006). New York : ACM Press, 2006. P. 369–376.

Методы машинного обучения для анализа биологической значимости экспрессии генов

Малахов Александр

Кафедра математической статистики

e-mail: sasha.malakhov.03@mail.ru

Научный руководитель — д.ф.-м.н. доц. Горшенин Андрей Константинович

Развитие технологии высокопроизводительного секвенирования нуклеиновых кислот привело к накоплению огромного количества данных. Одним из ключевых вызовов остается интерпретация полученных данных: среди тысяч генов необходимо определить, какие из них действительно активны и в какой мере их экспрессия биологически значима. Простое числовое значение, например 0.1 или 1.0, не всегда дает однозначный ответ на этот вопрос. Проблема заключается не только в биологическом разнообразии — разные клетки имеют отличный профиль экспрессии генов в зависимости от фазы клеточного цикла и типа клетки, но и в технических ограничениях методов оценки экспрессии, обусловленный чувствительность оборудования, погрешность измерений, batch effects.

В данной работе использованы нейросетевые методы для получения биологически значимых порогов, получены устойчивые кластеры генов и транскриптов. Пороги экспрессии для транскриптов рассчитаны впервые, что открывает новые возможности в интерпретации транскриптомных данных.

Биологическая постановка задачи Каждая клетка организма содержит одну и ту же генетическую информацию — весь геном, включающий около 20 000 белок-кодирующих генов. Однако только часть этих генов «работает» в каждый конкретный момент времени и в каждой конкретной клетке. То есть, несмотря на одинаковый набор генов, разные клетки экспрессируют (включают) разные гены, в зависимости от их функций, условий окружающей среды, фазы клеточного цикла и других факторов [1].

Важно понимать, что не все гены экспрессируются одинаково интенсивно. Некоторые могут «включаться» на очень высоком уровне — например, при стрессе или в ответ на внешний сигнал. Другие постоянно работают на «базовом» уровне, поддерживая жизнедеятельность клетки (это так называемые гены домашнего хозяйства (housekeeping-гены) или НКГ-гены) [2]. При этом нет четкой границы, начиная с какого уровня экспрессии ген считается «включенным». Например, экспрессия на уровне 0.3 у одного гена может быть функционально значимой, а у другого — признаком неактивности.

В живых организмах гены редко работают изолированно. Чаще всего они образуют координированные сети: группы генов могут одновременно активироваться или подавляться в ответ на один и тот же сигнал, а кодируемые ими белки участвовать в одном процессе (например, клеточном делении). Поэтому логично предположить, что гены с похожими профилями экспрессии

по множеству клеточных линий могут быть взаимосвязаны по функциям, регуляции или локализации.

Математическая формализация постановки задачи Пусть дан ген g , для которого известны два вектора наблюдений длины n :

- вектор экспрессии $\mathbf{x}_g = (x_{g_1}, x_{g_2}, \dots, x_{g_n})$, где x_{g_i} — значение экспрессии гена g в клеточной линии i ;
- вектор выживаемости $\mathbf{s}_g = (s_{g_1}, s_{g_2}, \dots, s_{g_n})$, где s_{g_i} — значение выживаемости клеточной линии i при нокауте гена g .

Цель — определить порог экспрессии $t_g \in \mathbb{R}_+$, характеризующий границу между значимым и незначимым уровнем экспрессии гена g в терминах его влияния на жизнеспособность клеточной линии.

Задача формализуется как оптимизационная:

$$t_g = \arg \max_{t \in \mathbb{R}_+} \mathcal{I}(t),$$

где $\mathcal{I}(t)$ — функционал значимости, измеряющий разделяющую способность порога t по отношению к влиянию экспрессии гена на выживаемость клеток. Данный функционал не является известным и в работе за него принимается информационная энтропия.

Таким образом, определение порога экспрессии гена сводится к задаче одномерной оптимизации по порогу, с использованием информации как об экспрессии, так и о функциональной значимости гена, заданной выживаемостью.

Методология формирования эмбедингов и их применения в задаче анализа экспрессии На первом этапе выполняется очистка и фильтрация исходных данных. Из таблиц экспрессии и клеточной выживаемости отбираются только те гены и клеточные линии, по которым известно не менее 75% значений. Это позволяет минимизировать влияние пропусков и обеспечить статистическую надежность последующего анализа.

После предобработки таблиц производится сжатие исходных транскриптомных данных с помощью вариационного автоэнкодера. Это необходимо для устранения избыточности, снижения размерности и выделения компактных латентных признаков, содержащих биологически значимую информацию [3].

Исходная матрица экспрессии $X \in \mathbb{R}^{n \times d}$, где n — количество генов, $d = 703$ — число клеточных линий, подается на вход энкодеру нейросетевой архитектуры. Энкодер состоит из двух линейных слоев:

$$d = 1406 \rightarrow 512 \rightarrow 64.$$

Особенностью данной реализации является использование **логнормального распределения** в качестве априорного распределения $p(z)$. Это позволяет модели учитывать специфическое распределение экспрессии генов, которое, согласно эмпирическим данным, может аппроксимироваться логнормальным

После получения сжатого представления каждого гена с помощью вариационного автоэнкодера следующим этапом является кластеризация этих латентных признаков. Для этого используется модель глубоких гауссовских смесей (DGMM, Deep Gaussian Mixture Model) — обобщение классической модели GMM, способное моделировать многоуровневую вероятностную структуру в скрытом пространстве [4, 5].

В данной работе используется **двухслойная архитектура DGMM**:

- на верхнем уровне: $k^{(2)} = 7$ компонент;
- на нижнем уровне: $k^{(1)} = 15$ компонент.

Заключительным этапом алгоритма является определение пороговых значений экспрессии для каждого гена. В данной работе для нахождения порогов используется подход, основанный на обучении модели градиентного бустинга на решающих деревьях [9, 10].

После обучения модели важно не только получить предсказания, но и извлечь из структуры деревьев значения экспрессии, разделяющие биологически значимые и фоновое неактивные случаи.

Для этого анализируются все внутренние узлы деревьев ансамбля, в которых происходит разбиение по значениям признаков x_{ij} (экспрессия гена i в клеточной линии j). Для каждого такого узла определяется, насколько эффективно он разделяет выборку по меткам кластеров. Это измеряется с помощью энтропии.

Для каждого гена i из всех внутренних узлов во всех деревьях выбирается узел, соответствующий разбиению по одной из клеточных линий j и обладающий максимальной энтропией H . Значение x_{ij} , по которому происходит это разбиение, принимается за порог экспрессии для гена i .

Анализ экспрессии генов и транскриптов В качестве исходных данных в работе использовались результаты транскриптомного профилирования клеточных линий человека, взятые с портала *Project Achilles* (<https://www.broadinstitute.org/achilles>) — проекта, направленного на выявление критически значимых генов в сотнях генетически охарактеризованных линий раковых клеток.

При определении порогов экспрессии генов, как промежуточный шаг, было получено их разбиение на кластеры. Для анализа полученных кластеров генов была проведена агрегация ряда биологических (таблица 10) и статистических (таблица 9) характеристик по каждой группе. Целью данного этапа являлось выявление функциональных закономерностей и оценка структуры латентного пространства, сформированного после кластеризации в скрытом представлении.

Результаты, полученные с использованием предложенного метода определения порогов экспрессии, демонстрируют его согласованность с биологической интерпретацией и интуитивную понятность. В частности, можно отметить, что для большинства генов, чья экспрессия наблюдается лишь в ограниченном числе клеточных линий, метод успешно исключает все нулевые и околонулевые значения. Такие гены составляют наибольшую часть выбор-

ки, и характерны тем, что активны лишь в отдельных условиях. Механизм отсекающего работает одинаково эффективно как в случае бимодального распределения экспрессии, так и при унимодальном характере данных.

Для высокоэкспрессируемых генов, таких как так называемые гены «домашнего хозяйства», практически отсутствуют наблюдения с нулевыми значениями экспрессии. В таких случаях метод определяет порог, отсеивающий малозначимые наблюдения в «хвосте» распределения, что позволяет выделить не критичные клеточные линии. При этом основная часть наблюдений остается сохраненной, что соответствует более высокой биологической значимости этих генов. Повышенное пороговое значение, получаемое для таких случаев, также способствует их отчетливому отделению от остальных генов в латентном пространстве и способствует формированию устойчивых функциональных групп.

Отдельное внимание заслуживают особые случаи, когда у гена наблюдается крайне ограниченное количество уникальных значений экспрессии — например, всего три, и все они расположены вблизи нуля. Даже в таких ситуациях метод, основанный на поиске разбиения с максимальной энтропией, показывает устойчивую работу, корректно выделяя границу, отделяющую потенциально значимые наблюдения от фона.

Определив значение уровня биологической значимости экспрессии гена, мы можем посчитать, в какой доле клеточных линий каждый ген экспрессируется. Для этого посчитаем долю таких наблюдений экспрессии в линиях, которые будут превышать уровень значимости для этого гена. Большая часть генов имеет околонулевое количество экспрессируемых клеточных линий, хотя есть малое количество с высокой долей.

При определении порогов экспрессии генов, как промежуточный шаг, было получено их разбиение на кластеры. Для анализа полученных кластеров генов была проведена агрегация ряда биологических и статистических характеристик по каждой группе. Целью данного этапа являлось выявление функциональных закономерностей и оценка структуры латентного пространства, сформированного после кластеризации в скрытом представлении.

Результаты анализа показали, что в структуре кластеров не наблюдается выраженного преобладания одной конкретной функциональной группы. Ни один из кластеров не содержит исключительно, например, рибосомные или только метаболические гены. Вместо этого большинство групп включают разнообразные по биологическим функциям гены, что может свидетельствовать о том, что кластеризация в латентном пространстве отражает скорее общие закономерности экспрессии и выживаемости, чем строгую функциональную специализацию. Это объяснимо и тем, что для большинства генов известно участие более чем в одном процессе. При этом, суммарно гены человека хоть и наиболее функционально аннотированы, но мало детализированы — для большинства текущее представление является общим описанием возможной функции.

Тем не менее, можно отметить отдельные тенденции. Кластеры с наибольшей средней экспрессией и наименьшей дисперсией по этому признаку, как правило, демонстрируют высокую долю экспрессируемых клеточных линий. Это может указывать на их вовлеченность в универсальные клеточные процессы. Также такие кластеры характеризуются, как правило, пониженной средней выживаемостью, что соответствует более высокой критичности входящих в них генов.

В других группах, напротив, преобладают гены со сравнительно низкими уровнями экспрессии и высокой гетерогенностью по клеточным линиям. Для таких кластеров характерна высокая медианная выживаемость и меньшая представленность НКГ-генов, что может указывать на более узкую функциональную специализацию или активацию в специфических условиях.

Анализ аннотированных кластеров транскриптов показал наличие функционально обогащённых групп. В кластере 0 выявлены гены, связанные с эпителиальной функцией, транспортом и межклеточными взаимодействиями. Кластер 2 включает ферменты гликолиза и метаболизма углеводов. Кластер 4 характеризуется генами, связанными с репликацией, трансляцией и клеточным делением. В кластере 5 представлены гены, участвующие в митохондриальной активности и адгезии. Кластеры 6 и 7 включают элементы внеклеточного матрикса, регуляции апоптоза и межклеточной сигнализации.

Номер кластера	Колич. генов	Средняя экспр.	σ экспр.	Медиана экспр.	Миним. экспр.	Средняя выжив.	σ выжив.	Ср. доля экспр. линий
0	1162	2.92	0.96	2.80	0.715	-0.163	0.142	0.462
1	883	3.12	0.95	3.01	0.822	-0.168	0.146	0.479
2	6271	2.34	0.78	2.26	0.544	-0.126	0.135	0.373
3	1168	2.87	0.97	2.76	0.687	-0.154	0.141	0.460
4	942	2.91	0.96	2.81	0.650	-0.159	0.141	0.460
5	945	2.82	0.94	2.72	0.640	-0.127	0.139	0.464
6	984	2.78	0.96	2.67	0.629	-0.151	0.140	0.435
7	1016	2.79	0.94	2.68	0.637	-0.140	0.139	0.437
8	874	2.92	0.95	2.80	0.676	-0.156	0.142	0.450
9	1499	2.86	0.97	2.75	0.686	-0.161	0.142	0.440
10	1052	3.02	0.95	2.91	0.723	-0.157	0.144	0.465
11	860	2.94	0.98	2.82	0.677	-0.158	0.139	0.453

Таблица 9: Статистики по кластерам генов

Таким образом, кластеризация в скрытом пространстве позволила выделить группы генов с общими паттернами экспрессии и значимости, однако не во всех кластерах удалось выявить строгую привязку к отдельным функциональным подмножествам. Это может отражать сложную и многомерную природу транскриптомных данных, в которых структурные признаки и биологические функции проявляются на пересекающихся уровнях.

На основе информации, полученной с помощью описанного алгоритма, были получены устойчивые группы транскриптов. Для каждого кластера

Номер кластера	Доля генов, являющихся/участвующих				
	Рибосомальном комплексе	Протеасомном комплексе	Цитокинами	Процессе гликолиза	Процессе окислительного фосфорилирования
0	0.033	0.029	0.060	0.016	0.009
1	0.038	0.031	0.049	0.015	0.016
2	0.211	0.142	0.341	0.011	0.064
3	0.049	0.033	0.053	0.006	0.018
4	0.031	0.018	0.038	0.015	0.018
5	0.027	0.024	0.062	0.013	0.011
6	0.033	0.031	0.062	0.018	0.009
7	0.029	0.027	0.024	0.014	0.004
8	0.035	0.042	0.042	0.014	0.004
9	0.053	0.053	0.100	0.015	0.027
10	0.042	0.049	0.058	0.017	0.027
11	0.024	0.018	0.047	0.023	0.024

Таблица 10: Доля функциональных групп в кластерах генов

рассчитаны метрики, аналогичные с вычисленными для групп генов (таблица 11), а главное, впервые рассчитаны индивидуальные пороги экспрессии для каждого транскрипта.

Таким образом, результаты подтверждают, что кластеры, выделенные в ходе анализа, действительно существенно различаются по численности. В некоторых случаях встречаются кластеры, где преобладают транскрипты с низким уровнем экспрессии, высокой дисперсией, что может указывать как на редкие, так и узкоспециализированные транскрипционные активности. В других — наоборот, средние значения и медианные иногда достигали значительного уровня при относительно невеликом стандартном отклонении. Разнообразие значений минимумов экспрессии внутри кластеров также указывает на неоднородное распределение, и таким образом, действительно требует индивидуального подхода к порогам значимости.

Номер кластера	Колич. генов	Средняя экспр.	σ экспр.	Медиана экспр.	Миним. экспр.	Ср. доля экспр. линий
0	509.0	2.292	1.531	1.997	0.094	0.533
1	409.0	0.830	0.558	0.724	0.022	0.275
2	505.0	2.315	1.441	2.041	0.128	0.522
3	598.0	2.546	1.492	2.292	0.248	0.567
4	224889.0	0.531	0.356	0.462	0.034	0.181
5	521.0	2.271	1.488	2.000	0.083	0.519
6	491.0	1.987	1.525	1.664	0.027	0.504

Таблица 11: Статистики по кластерам транскриптов

В целом, использование DGMM и VAE позволило кластеризовать транскрипты с большей эффективностью и одновременно более качественно аннотировать и интерпретировать результаты. Тем самым, метод оказался уни-

версальным и применимым как к генам, так и к транскриптам, обеспечивая более точное и детальное представление о транскриптомном ландшафте.

ЛИТЕРАТУРА

- [1] Antonio De Maio. “Heat shock proteins: facts, thoughts, and dreams”. В: Shock 11.1 (1999), с. 1—12.
- [2] Chintan J Joshi и др. “What are housekeeping genes?” В: PLoS computational biology 18.7 (2022), e1010295.
- [3] Way G. P. et al. Compressing gene expression data using multiple latent space dimensionalities learns complementary biological representations //Genome biology. – 2020. – Т. 21. – С. 1-27.
- [4] Viroli, Cinzia. Deep Gaussian Mixture Models / Cinzia Viroli, G. McLachlan // Statistics and Computing. — 2017. — 11. — Vol. 29. DOI: 10.1007/s11222-017-9793-z.
- [5] Gorshenin A. Toward Supervised Deep Gaussian Mixture Models* //International Conference on Distributed Computer and Communication Networks. – Cham : Springer Nature Switzerland, 2024. – С. 350-363.

Модель специализированной рекомендательной системы

Работа удостоена диплома 2 степени

Маннанов Айрат Альбертович

Кафедра алгоритмических языков

e-mail: s02210463@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. доц. Абрамов Владимир Геннадьевич

Актуальность Современная тенденция персонализированных сервисов, основанных на анализе поведения и профилей пользователей, делает актуальной задачу рекомендательных систем и в области физической активности. Рекомендательные системы в спорте способны эффективно поддерживать пользователей в формировании индивидуальных программ упражнений, снижая риск травм и повышая интерес к занятиям.

Цель работы Цель работы состоит в разработке модели персонализированной рекомендательной системы для физических упражнений, учитывающей индивидуальные особенности пользователей и динамику их поведения, а также в практической реализации описанной архитектуры.

Классическая постановка задачи рекомендаций

- $\mathcal{U}$ — множество пользователей;
- $\mathcal{I}$ — множество объектов (упражнений);
- $\mathcal{I}_u \subseteq \mathcal{I}$ — множество упражнений, оценённых пользователем u ;
- r_{ui} — рейтинг упражнения i пользователем u ;
- $\mathcal{R}_u = (r_{ui})_{i \in \mathcal{I}_u}$ — вектор рейтингов пользователя u .

Необходимо для каждого пользователя $u \in \mathcal{U}$ оценить значение r_{ui} для $i \in \mathcal{I} \setminus \mathcal{I}_u$ и предложить множество объектов $L_u = (i_1, i_2, \dots, i_n)$, $i \in \mathcal{I} \setminus \mathcal{I}_u$, упорядоченное по предсказанным значениям для данного пользователя.

Представление упражнений и пользователей Выделим основные две сущности — *упражнение* и *пользователь* (рис. 1). Атрибуты, которые представляют собой сущность, можно объединить в группы: описательные, классификационные и медицинские для обеих сущностей. Последние две группы должны иметь взаимно однозначное отображение между сущностями.

Метод на основе знаний Результирующий рейтинг упражнения e_j для пользователя u_i вычисляется по формуле:

$$\text{Score}(u_i, e_j) = \omega_{0i} k_{ij}^{(0)} + \omega_{1i} k_{ij}^{(1)} + \omega_{2i} k_{ij}^{(2)}$$

где

$$k_{ij}^{(0)} = \begin{cases} 1, & t_j = g_i, \\ 0, & \text{иначе,} \end{cases} \quad k_{ij}^{(1)} = 1 - \frac{|d_j - l_i|}{D}, \quad k_{ij}^{(2)} = \frac{|m_j \cap p_i|}{|m_j|}$$

Здесь $\omega_{0i}, \omega_{1i}, \omega_{2i}$ — весовые коэффициенты пользователя u_i , а $k_{ij}^{(0)}, k_{ij}^{(1)}, k_{ij}^{(2)}$ — факторы соотношения классификационных атрибутов упражнения и пользователя. Предложенный алгоритм является универсальным, так как позволяет

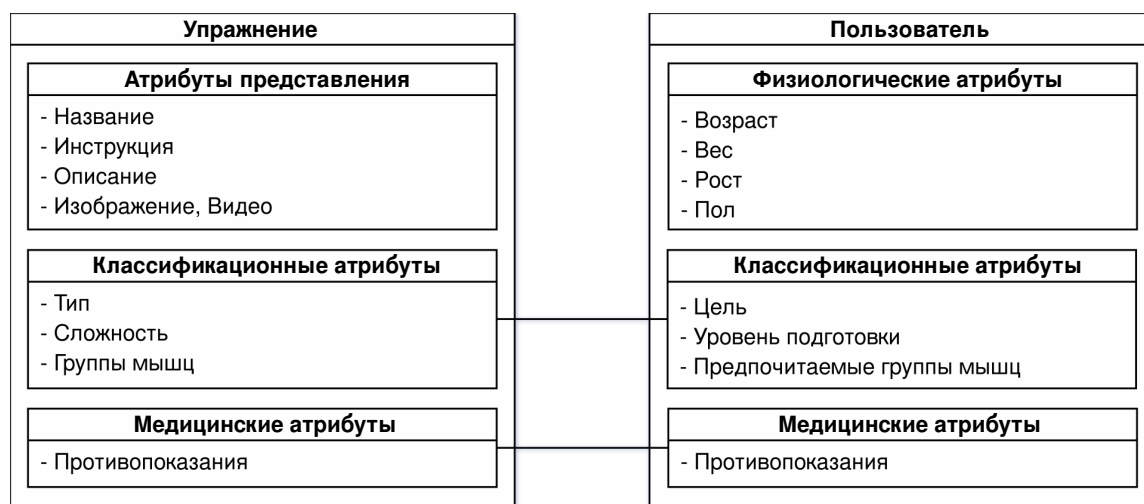


Рис. 1: Схема основных сущностей рекомендательной системы

соотносить упражнения и пользователей между собой, что сразу же разрешает как задачу регрессии рейтинга, так и задачу ранжирования объектов [1, стр. 138].

Метод на основе контента Профиль пользователя p_u формируется как взвешенное экспоненциальное среднее векторных представлений ранее оценённых упражнений:

$$p_u = \frac{\sum_{j=1}^n \gamma^{n-j} r_{u,i_j} \mathbf{e}_{i_j}}{\sum_{j=1}^n \gamma^{n-j} r_{u,i_j}},$$

где $\gamma \in (0, 1]$ — коэффициент затухания, $r_{u,i_j} \in \{0, 1\}$ — бинарный отклик, $\mathbf{e}_{i_j}$ — вектор признаков упражнения e_{i_j} . Для каждого кандидата e вычисляется косинусное сходство:

$$\text{sim}(p_u, \mathbf{e}) = \frac{p_u \cdot \mathbf{e}}{\|p_u\| \|\mathbf{e}\|},$$

упражнения ранжируются по убыванию sim , и в топ- N попадают наиболее релевантные. Данный алгоритм возносит к классическому подходу представлений контентных систем [1, 2].

Результаты Эффективность системы оценивалась на выборке из пяти пользователей с бинарной обратной связью (таб. 1). Метрики вычислялись по формулам:

$$\text{Precision@K} = \frac{|\{i \in L_u : r_{u,i} = 1\}|}{|L_u|}, \quad \text{NDCG@K} = \frac{\text{DCG@K}}{\text{IDCG@K}}.$$

Основные результаты

1. Проанализированы классические и современные методы рекомендаций и выделены основные подходы систем при проблеме холодного старта — отсутствия рейтинговых данных на начальном этапе работы системы.

Метод	Precision@3	Precision@10	NDCG@3	NDCG@10
Экспертный	0.87	0.92	0.86	0.90
Контентный	0.80	0.60	0.81	0.66

Таблица 1: Итоговые метрики алгоритмов

2. Разработана гибридная модель, сочетающая подходы на основе правил и контента, что обеспечивает высокую точность даже при ограниченных данных.
3. Создано прототипное веб-приложение с модульной архитектурой и подтверждающая работоспособность алгоритмов на практике.

ЛИТЕРАТУРА

- [1] Aggarwal Charu C. Recommender Systems: The Textbook. — Springer, 2016.
- [2] Ricci Francesco, Rokach Lior, Shapira Bracha. Recommender Systems Handbook. — Springer, 2022.

Численное моделирование тепловой эволюции малых тел планетных систем под действием излучения центральной звезды

Работа удостоена диплома 2 степени

Маркин Игорь Валерьевич

Кафедра суперкомпьютеров и квантовой информатики

e-mail: igormarkin2003@yandex.ru

Научный руководитель — к.т.н. доц. Русол Андрей Владимирович

О тепловой эволюции малых тел. Исследование тепловой эволюции [1] кометных ядер представляет важную задачу для понимания ранней истории планетных систем. Кометы сохраняют первичный состав протопланетного вещества и демонстрируют активность, связанную с сублимацией летучих веществ. Наблюдения (например, миссия Rosetta) показывают, что газовыделение неравномерно распределено и не всегда соответствует зонам максимального нагрева, что указывает на сложные тепловые процессы в приповерхностных слоях.

Понимание этих процессов важно как для изучения внутреннего строения и состава комет, так и для оценки реактивных эффектов, влияющих на движение малых тел. Моделирование требует учёта солнечного излучения, затенённости и фазовых переходов в пористом веществе. В качестве объекта выбрана комета 67P/Чурюмова–Герасименко, обладающая детально изученной формой и историей активности.

Математическая постановка задачи. Для описания тепловой эволюции в приповерхностных слоях малых тел используется одномерное уравнение теплопроводности:

$$\rho C_p(T) \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left(k(T) \frac{\partial T}{\partial x} \right),$$

где ρ — плотность, $C_p(T)$ и $k(T)$ — эффективные теплоёмкость и теплопроводность, зависящие от температуры. Глубина прогрева существенно меньше характерных размеров ядра, что оправдывает одномерную постановку.

Эффективная теплоёмкость модели учитывает фазовые переходы через добавление гауссового пика:

$$C_p(T) = C_p^{base} + \frac{mL}{\varepsilon\sqrt{\pi}} \exp \left(- \left(\frac{T - T_s}{\varepsilon} \right)^2 \right),$$

а также температурную зависимость базового уровня, например, через полином второй степени:

$$C_p(T) = a + bT + cT^2.$$

Аналогично, теплопроводность компонентов моделируется с температурной зависимостью экспоненциального вида:

$$k(T) = k_0 \cdot e^{-a/T}.$$

Температура на поверхности рассчитывается из баланса солнечного и собственного излучения:

$$(1 - A)SI = \epsilon\sigma T^4,$$

где A — альбедо, S — солнечная постоянная, I — освещённость, ϵ — эмиссионная способность, σ — постоянная Стефана–Больцмана.

Для учёта тепловой инерции применяется экспоненциальное приближение:

$$T_{surf}(t) = T_{prev} + (T_{target} - T_{prev}) \cdot (1 - e^{-\Delta t/\tau}).$$

Определение освещённости I является ключевой задачей модели: от неё напрямую зависит температурный режим и, следовательно, вся динамика дегазации при вращении тела.

Определение освещённости и затенённости. Освещённость поверхности [2] является ключевым параметром для определения температуры и, следовательно, для моделирования тепловых и фазовых процессов в поверхностных слоях. Для её расчёта используется геометрическая модель освещения. В рамках данной работы разрабатывается специализированная библиотека, предназначенная для высокоэффективного вычисления освещённости на основе трёхмерной геометрии объекта. Это позволяет применять её в задачах теплового моделирования поверхностей малых тел Солнечной системы.

Объект (например, ядро кометы) моделируется в виде трёхмерной полигональной сетки, состоящей из множества треугольных граней. Для каждой грани можно определить:

- $\vec{n}$ — единичный вектор нормали;
- $\vec{s}$ — направление на источник света от центра грани.

Освещённость грани I определяется как косинус угла θ между нормалью к грани и направлением на источник света:

$$I = \max(\cos \theta, 0) = \max\left(\frac{\vec{n} \cdot \vec{s}}{|\vec{n}| \cdot |\vec{s}|}, 0\right),$$

где:

- $\vec{n} \cdot \vec{s}$ — скалярное произведение нормали и направления на источник;
- $\max(\cdot, 0)$ — отсечение отрицательных значений, соответствующих теневой стороне.

Поскольку расчёт температуры на поверхности напрямую зависит от освещённости каждой грани, необходима эффективная реализация соответствующего алгоритма. Это особенно важно при работе с высокодетализированными моделями сложной формы (например, моделью ядра кометы 67P/Чурюмова–Герасименко), содержащими сотни тысяч или миллионы граней.

В ходе работы были разработаны и протестированы четыре версии реализации:

1. На `Python` — базовая версия с явными циклами. Простая в реализации, но крайне медленная при больших объёмах данных.
2. С использованием `NumPy` — векторизованная реализация, где все вычисления выполняются над массивами. Достигается значительный прирост скорости за счёт отказа от циклов и использования векторизованных функций `NumPy`.
3. На `C++` — производительная версия, реализованная с учётом низкоуровневой работы с памятью. Обеспечивает высокую скорость и возможность работы с большими массивами данных.
4. `C++ + Pybind11` — модуль на `C++`, интегрированный в `Python` с использованием библиотеки `Pybind11`. Совмещает удобство использования в `Python`-окружении с производительностью и управлением памятью на уровне `C++`.

В итоге выбрана реализация с использованием `Pybind11`, поскольку она обеспечивает производительность, сопоставимую с чистым `C++`, при этом оставаясь доступной из `Python`-среды и превосходя по эффективности версии на `Python` и `NumPy`.

Для корректного моделирования тепловой эволюции тел сложной формы необходимо учитывать эффект взаимного затенения граней поверхности. В рамках работы разработан алгоритм определения затенённости, основанный на трассировке лучей от центра каждой грани в направлении на источник света. Грань считается затенённой, если соответствующий луч пересекает любую другую грань модели.

С целью уменьшения вычислительной сложности ($O(n^2)$ при наивной реализации), алгоритм включает несколько этапов оптимизации: проецирование центров всех граней на плоскость, ортогональную направлению на источник света; использование KD-дерева, построенного по двумерным координатам проекций, для ускоренного поиска потенциально затеняющих граней.

KD-дерево(`scipy.spatial.cKDTree`) обеспечивает построение за $O(n \log n)$ и поиск соседей в заданном радиусе за $O(\log n)$ в среднем.

Для найденных кандидатов применяется строгая проверка на пересечение луча с треугольником методом Мёллера—Трумбора, эффективно работающим с произвольными треугольниками за $O(1)$ на одну проверку. В результате достигается существенное снижение вычислительной нагрузки до $O(n \log n)$.

Обработка высокополигональных моделей с десятками миллионов граней требует особых подходов к управлению памятью. Для эффективной работы реализована стриминговая архитектура с использованием `DuckDB`.

Хранилище на базе `DuckDB` обеспечивает колонковое хранение, отложенную загрузку и совместимость с `Pandas/NumPy`. Загрузка модели делится на два этапа: считывание вершин и определение параметров граней. В процес-

се расчётов данные подгружаются батчами, что исключает необходимость полной загрузки модели в ОЗУ и допускает работу с произвольно большими моделями. Для задачи определения затенённости реализован механизм пакетной выборки, извлекающий координаты центров, нормали, индексы и координаты вершин. Все вычисления выполняются в пределах текущего батча.

Для ускорения вычислений добавлена поддержка распределённой обработки на основе MPI [3] с использованием библиотеки `mpi4py`. Каждый процесс загружает независимый диапазон граней из DuckDB и обрабатывает его независимо, что обеспечивает масштабируемость и снижение общего времени вычислений.

Моделирование тепловой эволюции. Рассматривается задача теплопроводности в поверхностном слое ядра кометы на расстоянии 1,5 а.е. от Солнца. Моделирование проводится на протяжении 100 суток с использованием одномерной нестационарной модели и неявной разностной схемы.

Область глубины $[0, L]$ дискретизуется на N равных интервалов с шагом $\Delta x = \frac{L}{N}$, узлы: $x_i = i\Delta x$, $i = 0, \dots, N$. Температура в узлах: $T_i^n \approx T(x_i, t_n)$.

Используется неявная схема:

$$\rho C p_i^{n+1} \frac{T_i^{n+1} - T_i^n}{\Delta t} = \frac{k_{i+\frac{1}{2}}^{n+1}(T_{i+1}^{n+1} - T_i^{n+1}) - k_{i-\frac{1}{2}}^{n+1}(T_i^{n+1} - T_{i-1}^{n+1})}{\Delta x^2},$$

Граничные условия:

- $x = 0$: $T_0^n = T_0$,
- $x = L$: $\left. \frac{\partial T}{\partial x} \right|_{x_N} \approx \frac{T_N^n - T_{N-1}^n}{\Delta x} = 0$.

Получается система линейных уравнений вида $A T^{n+1} = b$. Формируется разреженная матрица A в формате `csr` с помощью функции `diags` из `scipy.sparse`, решение вычисляется с использованием `spsolve`. На каждом шаге происходит обновление A и b с учётом температурозависимости коэффициентов.

Выводы. Реализована библиотека определения освещённости для которой разработаны и протестированы различные реализации алгоритма расчёта освещённости поверхности. Используется ядро на `Pybind11`, которое даёт выигрыш по времени в 6 раз по сравнению с `NumPy`. С помощью DuckDB реализована поточная обработка модели и предоставлены параллельные версии алгоритмов для обработки на кластерах с помощью `mpi4py`. Результаты представлены на конференции «Ломоносовские чтения–2025» [4].

Полученные результаты моделирования тепловой эволюции согласуются с наблюдаемым поведением комет, где активные выбросы могут происходить с малой части поверхности, не всегда совпадающей с зонами максимального нагрева. Предложенная модель позволяет наблюдать процессы сублимации CO_2 и CO даже в тех случаях, когда рассматриваемый участок находится в тени, и интерпретировать интенсивные газопылевые выбросы как результат накопления в небольшой приповерхностной области сублимированных

летучих веществ, избыточное давление которых и приводит к образованию джетов.

ЛИТЕРАТУРА

- [1] Маров М. Я., Русол А. В., Дорофеева В. А. Численное моделирование длительной тепловой эволюции ядер короткопериодических комет: на примере ядра кометы 67Р/Чурюмова–Герасименко // Доклады академии наук. 2019. Т. 484, № 2. С. 150–155.
- [2] Маров М. Я., Русол А. В., Дорофеева В. А. Трёхмерная модель освещённости кометных ядер: на примере ядра кометы 67Р/Чурюмова–Герасименко // Доклады академии наук. 2017. Т. 474, № 1. С. 41–45.
- [3] Антонов А. С. Технологии параллельного программирования MPI и OpenMP. М. : Издательство Московского университета, 2012. 344 с. (Суперкомпьютерное образование).
- [4] Русол А. В., Маркин И. В. Моделирование освещённости поверхности космических объектов на современных высокопроизводительных архитектурах // Ломоносовские чтения–2025: научная конференция. Факультет ВМК МГУ имени М. В. Ломоносова. Тезисы докладов. Серия «Вычислительная математика и кибернетика». М. : МАКС Пресс, 2025. С. 133–134.

Повышение точности и полноты статического анализатора Svace с помощью специфицирования библиотечных функций большими языковыми моделями

Митрофанов Андрей Александрович

Кафедра системного программирования

e-mail: a.mitrofanov@ispras.ru

Научный руководитель — к.ф.-м.н. доц. Игнатъев Валерий Николаевич

Научный консультант — к.т.н. Шимчик Никита Владимирович

Статический анализ кода представляет собой метод исследования исходного кода программ без его фактического выполнения. Одной из проблем статического анализа является моделирование вызовов библиотечных функций, реализации которых отсутствуют в анализируемом проекте. В общем случае анализатору приходится делать консервативные предположения о состоянии программы после выполнения такого вызова — например, что значения всех переданных функции аргументов могут быть изменены произвольным образом или наоборот, что вызовы функций не влияют на значения переменных.

Поскольку такие общие предположения плохо описывают реальное поведение программы, на практике для наиболее важных и часто используемых библиотечных функций составляются *спецификации* — модельные описания поведения функций, вызываемых в исследуемой программе. Они необходимы, чтобы моделировать их важные для статического анализа свойства.

К спецификациям предъявляются следующие требования:

1. Спецификации должны быть без синтаксических ошибок и содержать определения используемых типов данных
2. Для каждого предопределенного набора свойств, рассматриваемого статическим анализатором, в спецификацию должен быть вставлен соответствующий код, моделирующий поведение функции для этого свойства.
3. Следует избегать вызовов функций, помимо API анализатора, циклов или другого кода, который усложняет семантику, поскольку такой код замедлит анализ и может ухудшить результаты.

На данный момент спецификации в Svace [1] создаются вручную и покрывают только небольшое множество часто используемых функций из популярных библиотек. Поэтому существующий подход трудоёмок, времязатратен и зачастую недостаточен для полноценного анализа проектов.

С точки зрения машинного обучения моделирование функций представляет собой задачу кодогенерации. Однако, общедоступные *инструменты генерации кода* [2] не знакомы с особенностями Svace и не обучены на документации всех библиотек функций, из-за чего они не подходят для полноценного решения задачи.

В то же время в последние годы популярность больших языковых моделей (LLM) кратно выросла, что связано с их умением работать с широким спектром задач, в том числе для генерации текста и кода.

Целью данной работы является повышение точности и полноты статического анализатора Svase с помощью специфицирования библиотечных функций большими языковыми моделями. Автоматизация процесса генерации спецификаций позволит существенно расширить покрытие библиотечных функций, снизить трудоёмкость ручного написания спецификаций и повысить качество статического анализа проектов.

Алгоритм генерации спецификаций. Для задачи генерации спецификаций входной информацией является исходный код специфицируемых библиотек, из которого при помощи *модуля разбора исходного кода библиотек функций* получается информация об определениях типов, а также о публичных функциях, доступных для использования извне.

На следующем этапе при помощи *модуля подготовки запросов* из собранных функций формируются запросы к LLM, а в результате получаются прототипы спецификаций.

Далее *модуль компиляции результатов* обрабатывает эти спецификации и приводит к необходимому формату. На вход данного модуля подаются как результаты генерации LLM, так и определения типов и структур из *модуля разбора исходного кода библиотек функций*. Из данного модуля получаются готовый к использованию файл спецификаций и спецификации, в которых были обнаружены ошибки.

Также не менее важным является *модуль сравнения спецификаций*, с помощью которого можно оценить качество сгенерированных спецификаций. Данный модуль получает на вход имеющийся в статическом анализаторе Svase набор спецификаций для часто используемых библиотечных функций из стандартных библиотек (*корпус*), включающий в себя 1461 функцию для языка C и набор файлов с новыми спецификациями и выводит таблицу метрик сравнения.

Предложенный алгоритм позволяет обрабатывать произвольные библиотеки исходного кода, выполнять предобработку под различные подходы запроса к LLM, а также получать итоговый файл со спецификациями вне зависимости от подхода к генерации прототипов спецификаций.

Модуль разбора исходного кода библиотек функций. Для получения определений типов, а также информации о публичных функциях, доступных для вызова пользователям этих библиотек, был разработан *метод разбора исходного кода библиотек функций*.

Использование регулярных выражений или других примитивных методов извлечения данных из библиотек нецелесообразно из-за разнообразия реализаций и уникальных структур объявлений в различных библиотеках. Поэтому был использован синтаксический парсер *Tree-sitter* [3] — библиотека с

открытым исходным кодом, предназначенная для синтаксического анализа и манипуляций с кодом на различных языках программирования.

С помощью скрипта, использующего Tree-sitter, получаются представления кода в виде определений типов данных и информации о функциях.

Модуль подготовки запросов. При помощи *модуля подготовки запросов* из полученных функций формируются запросы к LLM, чтобы в результате получать прототипы спецификаций.

Весь процесс решения задачи при помощи LLM можно разделить на три подхода, каждый следующий из которых является развитием предыдущего варианта:

1. Отправка *запросов с несколькими примерами* предполагает передачу вместе с целевым запросом нескольких пронумерованных примеров «вход-выход», демонстрирующих, как из описания функции получить спецификацию.
2. *Улучшение запросов* используется в случаях, когда качество результатов *генерации по примерам* оказывается недостаточным, модель выдаёт результаты в неправильном формате или в принципе не умеет решать задачу. Усложнение подразумевает под собой систематическую разработку и оптимизацию шаблонов запросов.
3. *Дообучение LLM* необходимо для решения задач, на которых модель не обучалась, например, при необходимости LLM оперировать доменно-специфичными терминами. Дообучение требует наличия очищенного и размеченного датасета, подбора гиперпараметров, но позволяет добиться наилучшего качества генерируемых спецификаций, учесть доменные нюансы конкретных API и получить необходимый формат результатов.

По результатам каждого подхода следует анализировать ключевые метрики, чтобы оценить прогресс в следующем варианте. Таким образом, с помощью данного модуля на выходе получают *прототипы спецификаций*, полученные после применения какого-либо из подходов.

Модуль компиляции результатов. Для обеспечения соответствия требованиям спецификации корректно обрабатываются при помощи *модуля компиляции результатов*, который принимает на вход:

- файл с результатами генерации в формате ассоциативного массива;
- определения типов и структур, извлечённые из исходных библиотек.

Результатом данного модуля будут новый, готовый к использованию файл спецификаций и спецификации, в которых были выявлены какие-либо проблемы (например, использование несуществующего API или несоответствие необходимому формату).

Модуль сравнения спецификаций. Основной подход к оценке качества спецификаций — запуск статического анализа проектов со спецификациями, сгенерированными LLM, и разметка новых и исчезнувших предупреждений. Однако такое получение метрик требует значительных трудозатрат.

В качестве альтернативного способа разработан *метод сравнения спецификаций*, который автоматически вычисляет схожесть сгенерированных спецификаций и корпуса. Для этого требуется, чтобы в обоих наборах присутствовали модели для одного и того же набора библиотечных функций.

Для такой оценки автоматически сгенерированных спецификаций вводятся метрики: *точность*, *полнота* и *F₁-мера*. При этом спецификации из корпуса Svace считаются эталоном, а метрики считаются по наличию/отсутствию вызовов спецфункций в спецификации, отдельно для каждой переменной или параметра.

Таким образом, появляется возможность формализовать и сравнивать качество работы различных моделей LLM при генерации спецификаций для библиотечных функций без необходимости ручной разметки предупреждений статического анализатора.

Тестирование. В качестве базовой LLM для генерации спецификаций — с учётом ограничений вычислительных ресурсов и необходимости работы с кодом — была выбрана модель Phind-CodeLlama-34B-v2.

Были протестированы различные подходы работы с LLM:

1. Тривиальный подход, заключающийся в генерации спецификаций на основе нескольких примеров «функция-спецификация», не позволил модели научиться работать с задачей. Использование API было некорректно, а сами спецификации не соответствовали необходимым требованиям.
2. Для решения обнаруженных проблем был разработан подход к генерации спецификаций, который основывался на создании специальных правил, описывающих моделируемые свойства, контекст использования и информацию о передаваемых параметрах Svace API. Это позволило модели генерировать спецификации с использованием API в корректных контекстах и с правильными аргументами вызовов. Однако, в связи с особенностями данного подхода, на вход требовалось подавать исходный код специфицируемых функций, а сами запросы получались объемными, около 8000 токенов, что часто приводило к эффекту забывания [4]. Также модели не понимали особенностей реализации спецификаций, в связи с чем большая часть моделей библиотечных функций генерировалась некорректно.
3. Далее был применен метод дообучения большой языковой модели на примерах спецификаций с использованием Low-Rank Adaptation [5]. Такой подход позволил модели внутри заданного контекста не только правильно применять и понимать API, но и сохранять компактность запроса и точность генерируемых прототипов спецификаций.

Сравнение корпуса со спецификациями, сгенерированными дообученными с разными гиперпараметрами моделями, позволило выявить модель, обученную наилучшим образом. Лучшая модель сгенерировала спецификации с 93.9% полнотой, 91.7% точностью, 89% F1-мерой и 75.8% полным совпадением по функциям.

Эта же дообученная модель была использована для генерации спецификаций для библиотеки GLIBC. Было сгенерировано 1179 новых спецификаций, из которых 1006 (85,3%) соответствовали требованиям.

Анализ проектов `tinypg`, `nginx`, `curl` и `GDB` с этими спецификациями позволил выявить 113 новых предупреждений, из которых 96 оказались истинными.

Заключение. В работе был представлен алгоритм генерации спецификаций для библиотечных функций на языке C с помощью LLM на основе дообучения. Подход был реализован в промышленном статическом анализаторе `Svace` и протестирован на реальных проектах с помощью больших языковых моделей с открытыми весами.

Использование спецификаций, сгенерированных с помощью LLM, в статическом анализаторе `Svace` для анализа реальных проектов улучшило полноту и точность анализа, позволив выявить 113 новых предупреждений, из которых 96 оказались истинными.

ЛИТЕРАТУРА

- [1] Иванников В. П., Белеванцев А. А., Бородин А. Е., Игнатьев В. Н., Журин Д. М., Аветисян А. И., Леонов М. И. Статический анализатор `Svace` для поиска дефектов в исходном коде программ // Труды Института системного программирования РАН. 2014. С. 231–250. DOI: 10.15514/ISPRAS-2014-26(1)-7.
- [2] Liu F., Liu Y., Shi L., Huang H., Wang R., Yang Z., Zhang L., Li Z., Ma Y. Exploring and evaluating hallucinations in llm-powered code generation // arXiv preprint arXiv:2404.00971. 2024.
- [3] Su N., Burr P., Donovan B. Tree-Sitter: An incremental parsing system // Proceedings of the ACM on Programming Languages. 2020.
- [4] Liu C., Wang Y., Flanigan J., Liu Y. Large language model unlearning via embedding-corrupted prompts // Advances in Neural Information Processing Systems. 2024. Т. 37. С. 118198–118266.
- [5] Hu E. J., Shen Y., Wallis P., Allen-Zhu Z., Li Y., Wang S., Wang L., Chen W. Lora: Low-rank adaptation of large language models // ICLR. 2022. Т. 1, N 2. С. 3.

Применение моделей машинного обучения для прогноза временных характеристик исполнения программы на супервычислителях в реальном времени

Могиленец Виктор Максимович

Кафедра автоматизации систем вычислительных комплексов

e-mail: mogilenec.v.m@asvk.cs.msu.ru

Научный руководитель — к.ф.-м.н. в.н.с Писковский Виктор Олегович

Введение Актуальность исследования обусловлена ростом сложности вычислительных задач и необходимостью эффективного использования ресурсов суперкомпьютеров. Традиционные методы прогнозирования времени выполнения программ не учитывают динамическую природу вычислительной среды, что снижает точность предсказаний. Применение адаптивных моделей машинного обучения позволяет учитывать широкий спектр факторов, включая характеристики задач и конфигурацию оборудования. Целью работы является разработка и сравнительный анализ моделей, способных оперативно адаптироваться к изменениям в реальном времени.

Методология Основная задача формализована как регрессионная проблема: для вектора признаков ($\vec{x} \in \mathbb{R}^n$), описывающего параметры задачи и среды, требуется построить модель ($F(\vec{x})$), минимизирующую отклонение от фактического времени выполнения (y). Данные для исследования были предоставлены центром коллективного пользования «Высокопроизводительные вычисления и большие данные» (ЦКП «Информатика») ФИЦ ИУ РАН (г. Москва) и представляют собой журнал работы суперкомпьютера за два года. В ходе исследования в них был обнаружен сильный дисбаланс классов в сторону задач с малым (до 500 секунд) временем исполнения. Для его устранения использовался метод синтетической генерации. При помощи генеративной сети CTGANSynthesizer из библиотеки SDV [1] был создан дополненный набор данных из 20 000 экземпляров, сбалансированный по распределению целевой переменной ElapsedRaw (времени исполнения задачи) и на 71% соответствующий исходному набору.

В работе реализован двухэтапный процесс обучения, состоящий из фазы предварительного обучения модели на синтетических данных и фазы дальнейшего ее дообучения с применением подходящего алгоритма дообучения.

Сравнивались три модели машинного обучения: многослойный перцептрон (MLP), ансамблевая модель TabM [2], основанная на MLP моделях, и деревья решений в связке с градиентным бустингом [3]. Для этих моделей были подобраны методы их дообучения: MLP и TabM — мини-батчевый градиентный спуск [4], для деревьев решений использовалась модификация градиентного бустинга, модифицирующая деревья решений [5]. Для оценки использовались метрики RMSE(корень среднеквадратичной ошибки) и MAPE(средняя абсолютная процентная ошибка).

Экспериментальные результаты Итоговые результаты работы представлены в таблице 12. Ключевые наблюдения:

- Ансамблевые архитектуры эффективнее адаптируются к динамическим изменениям;
- Оптимальной функцией активации для MLP оказался гиперболический тангенс;
- Деревья решений в связке с градиентным бустингом обеспечили наименьшее время обучения при приемлемой точности.

Ограничением TabM является повышенная вычислительная сложность, что требует оптимизации для систем с жесткими требованиями к задержкам.

	<i>MAPE (%)</i>	<i>RMSE (сек.)</i>
<i>MLP</i>	18.5	1350
<i>TabM</i>	18	1364
<i>CatBoost</i>	18.4	1464

Таблица 12: Итоговая точность рассмотренных моделей после дообучения

Заключение Рассмотренные модели машинного обучения позволяют прогнозировать время выполнения задач на суперкомпьютерах с ошибкой менее 19%. Доказана эффективность предложенного подхода, сочетающего синтетическую генерацию данных и инкрементальное обучение. Наилучшие результаты показала ансамблевая архитектура TabM, адаптированная для табличных данных. Перспективными направлениями дальнейших исследований являются разработка гибридных моделей и методов активного обучения для сокращения времени адаптации. Результаты работы могут быть интегрированы в системы управления суперкомпьютерными комплексами для оптимизации планирования задач и использования ресурсов.

ЛИТЕРАТУРА

- [1] Data to AI Lab at MIT. SDV: Synthetic Data Vault Documentation. 2025.
- [2] Gorishniy Y., Kotelnikov A., Babenko A. TabM: Advancing Tabular Deep Learning with Parameter-Efficient Ensembling // arXiv preprint. 2025.
- [3] Prokhorenkova L. et al. CatBoost: unbiased boosting with categorical features // Advances in Neural Information Processing Systems. 2018. Vol. 31.
- [4] Smith, Samuel & Kindermans, Pieter-Jan & Le, Quoc. Don't Decay the Learning Rate, Increase the Batch Size. 10.48550/arXiv.1711.00489. 2017.
- [5] Chen T., Guestrin C. Xgboost: A scalable tree boosting system // Proceedings of the 22nd ACM SIGKDD. 2016. P. 785–794.

Применение байесовских графовых моделей для решения некоторых обратных задач

Мозохин Николай Сергеевич

Кафедра математической физики

e-mail: nikolay.mozokhin@gmail.com

Научный руководитель — к.ф.-м.н. доц. Березин Сергей Борисович

Введение. Сегодня обратные задачи очень распространены и встречаются во многих прикладных областях технических наук. Существует множество способов их численного решения (например, регуляризация Тихонова [4]). Поэтому поиск хороших нестандартных методов для решения обратных задач является актуальным вопросом.

В данной работе был предложен способ решения обратных задач с использованием байесовских графовых моделей, а также рассматривалось его применение для нескольких конкретных обратных задач — для системы Лотки—Вольтерры, уравнения теплопроводности и уравнения Пуассона.

Принципы построения графовых моделей для решения обратных задач. Особенностью байесовских графовых моделей (байесовских сетей) является использование в своей основе случайных величин. Условная зависимость между ними задается в виде графа, отсюда и название. Получается, таким методом могут прогнозироваться распределения, а не только скалярные значения.

Существует несколько алгоритмов перерасчета распределений в графовой модели. Одними из них являются *Expectation propagation* [1] и *Variational message passing* [2]. Основа этих двух алгоритмов — передача по графу от вершины к вершине сообщений, меняющих распределения этих вершин.

В данной работе графовые модели строились на основе разностных схем решаемых задач. Перечислим основные шаги получения модели.

- Значения всех функций и коэффициентов в узлах сетки принимаются за случайные величины, использующиеся в вершинах графа.
- Задаются априорные распределения для всех вершин (используется нормальное распределение).
- Разностные уравнения формируют основу для связей в графе.
- Дополнительные условия (начальные, граничные) задают фиксированные (с вероятностью 1 в точке) распределения некоторым вершинам, из этого и происходит перерасчет распределений в остальных.

Система проведения вычислительных экспериментов. Эксперимент проводился в две стадии: сначала численно решалась прямая задача для получения входных данных, затем по этим входным данным графовая модель предсказывала решение обратной задачи, которое потом сравнивалось с точным.

Реализация графовых моделей была выполнена на языке программирования *C#* с использованием библиотеки байесовского прогнозирования *Infer.NET* [3], численное решение, анализ и построение графиков были выполнены на языке программирования *Python* с использованием библиотек *NumPy*, *SciPy* и *Matplotlib*.

Обратная задача для системы Лотки—Вольтерры. Рассмотрим классическую модель Лотки—Вольтерры следующего вида:

$$\begin{cases} u'(t) = au(t) - bu(t)v(t) \\ v'(t) = cu(t)v(t) - dv(t) \end{cases} \quad t \in [0, T] \quad (1)$$

Пусть известна динамика популяции, то есть сеточные функции $u(t)$ и $v(t)$ на промежутке времени $[0, T]$. Требуется найти коэффициенты a, b, c, d модели Лотки—Вольтерры, связанной с этими функциями.

Для построения графовой модели рассмотрим явную разностную схему Эйлера для системы (1), выразим из нее u_{n+1} и v_{n+1} .

$$\begin{cases} u_{n+1} = (au_n - bu_nv_n)h + u_n \\ v_{n+1} = (cu_nv_n - dv_n)h + v_n \end{cases} \quad n = \overline{0, N} \quad (2)$$

Особенностью данной модели является прогнозирование не самих коэффициентов a, b, c, d , а параметров их распределений: математических ожиданий и обратных величин к дисперсиям (ограничения пакета *Infer.NET*). Такой подход дает большую гибкость по сравнению с прогнозированием самих величин.

$$a \sim N(\xi_1, \frac{1}{\sigma_1}), \quad b \sim N(\xi_2, \frac{1}{\sigma_2}), \quad c \sim N(\xi_3, \frac{1}{\sigma_3}), \quad d \sim N(\xi_4, \frac{1}{\sigma_4})$$

Начальными распределениями для прогнозируемых параметров были выбраны

$$\xi_i \sim N(1, 100), \quad \sigma_i \sim \Gamma(1, 1), \quad i = \overline{1, 4},$$

где N обозначает нормальное распределение, Γ — гамма-распределение.

Для соответствия реальным условиям использования в приведенную выше модель были добавлены переменные, отвечающие за шумовой фактор в данных (они обозначены $\delta_{u,n}$ и $\delta_{v,n}$, а чертой обозначены наблюдаемые зашумленные функции).

$$\begin{aligned} \overline{u_n} &= u_n + \delta_{1,n}, \quad \overline{v_n} = v_n + \delta_{2,n} \\ \delta_{i,n} &\sim N(0, \frac{1}{\sigma_{i,noise}}), \quad \sigma_{i,noise} \sim \Gamma(1, 1), \quad i = \overline{1, 2}, \quad n = \overline{0, N} \end{aligned}$$

Таким образом, разностная схема принимает вид

$$\begin{cases} \overline{u_{n+1}} \approx (a\overline{u_n} - b\overline{u_n}\overline{v_n})h + \overline{u_n} + \delta_{1,n+1} - \delta_{1,n} \\ \overline{v_{n+1}} \approx (c\overline{u_n}\overline{v_n} - d\overline{v_n})h + \overline{v_n} + \delta_{2,n+1} - \delta_{2,n} \end{cases} \quad n = \overline{0, N}$$

Здесь влияние шума внутри скобки было опущено для ускорения времени прогнозирования, это позволяет сделать малость шага сетки h . Эти соотношения и были взяты в качестве связей между случайными величинами в

графе. Аналогичным образом можно построить графовую модель на основе неявной разностной схемы Эйлера, что также было сделано в работе.

Для вычислений по графовой модели использовался алгоритм *Expectation propagation*, каждый прогноз был основан на 50 итерациях алгоритма.

В качестве модели для сравнения был выбран метод минимизации невязки. Его суть заключается в минимизации относительно a, b, c, d вектора, состоящего из невязок разностных уравнений (2).

$$\begin{aligned}\phi_n &= u_{n+1} - u_n - (au_n - bu_nv_n)h, \quad n = \overline{0, N-1} \\ \psi_n &= v_{n+1} - v_n - (cu_nv_n - dv_n)h, \quad n = \overline{0, N-1} \\ \Psi &= (\phi_0, \psi_0, \phi_1, \psi_1, \dots, \phi_{N-1}, \psi_{N-1}) \\ \|\Psi\|_2 &= \sqrt{\phi_0^2 + \psi_0^2 + \dots + \phi_{N-1}^2 + \psi_{N-1}^2} \xrightarrow{a,b,c,d} \inf\end{aligned}$$

В эксперименте сравнивались три модели: две графовые, первая на основе явной разностной схемы Эйлера, вторая — на неявной, и модель, минимизирующая невязку.

Для тестирования генерировалась случайная выборка коэффициентов $a, b, c, d \sim U[1, 10]$ и пар $u_0, v_0 \sim U[1, 3]$ длиной $N = 500$. По ним с помощью явного метода Рунге—Кутты 5 порядка точности решалась прямая задача для получения функций u_n, v_n на отрезке $[0, 10]$ с шагами сетки $h = 0.01, 0.02, 0.04, 0.1$. После этого к функциям могли добавляться шумовые компоненты $\delta_{1,n}, \delta_{2,n} \sim N(0, (\max_{t \in [0, T]} v(t) - \min_{t \in [0, T]} v(t)) * 0.05)$.

Далее с помощью графовой модели строилось предсказание коэффициентов. После этого снова численно решалась задача Коши с шагом сетки $h = 0.01$ для построения графиков функций и анализа.

Ниже представлена ключевая метрика оценивания (для самих коэффициентов). Аналогичным образом строились метрики для максимума, минимума и периода предсказанных функций. Здесь переменные без черты обозначают параметр точного решения, а переменные с чертой — предсказанного. Верхний индекс i показывает номер элемента из выборки.

$$E_{a,b,c,d} = \frac{1}{2N} \left(\sum_{i=1}^N \frac{|a^i - \hat{a}^i|}{a^i} + \sum_{i=1}^N \frac{|b^i - \hat{b}^i|}{b^i} + \sum_{i=1}^N \frac{|c^i - \hat{c}^i|}{c^i} + \sum_{i=1}^N \frac{|d^i - \hat{d}^i|}{d^i} \right)$$

шаг сетки	графовая модель (явная схема)	графовая модель (неявная схема)	минимизация невязки
0.01	0.95%	0.95%	1.26%
0.02	1.82%	1.85%	2.75%
0.04	3.63%	3.90%	6.37%
0.1	10.59%	12.29%	20.35%

Табл. 1: Значение метрики $E_{a,b,c,d}$ для незашумленных данных.

шаг сетки	графовая модель (явная схема)	графовая модель (неявная схема)	минимизация невязки
0.01	1.42%	1.42%	8.18%
0.02	2.54%	2.51%	6.67%
0.04	4.76%	4.64%	8.67%
0.1	11.24%	10.85%	21.56%

Табл. 2: Значение метрики $E_{a,b,c,d}$ для зашумленных данных.

Результаты эксперимента представлены в Табл. 1 и Табл. 2. Для остальных метрик были получены качественно сходные результаты.

Обратная задача для уравнения теплопроводности. Рассмотрим обратную задачу для одномерного уравнения теплопроводности на отрезке $[0, a]$ с фиксированным коэффициентом теплопроводности следующего вида.

$$\begin{cases} u_t(x, t) = u_{xx}(x, t) \\ u(0, t) = u(a, t) = 0 \\ u(x, T) = \phi(t) \end{cases} \quad x \in [0, a], \quad t \in [0, T] \quad (3)$$

Требуется найти $u(x, 0)$ — распределение температуры в начальный момент времени. Задача по своим свойствам является некорректной.

Построим графовую модель. Рассмотрим чисто неявную четырехточечную разностную схему для прямой задачи, связанной с задачей (3). Здесь $\gamma = \tau/h^2$, где τ — шаг по времени, h — по пространству.

$$(1 + 2\gamma)y_i^{n+1} = y_i^n + \gamma(y_{i-1}^{n+1} + y_{i+1}^{n+1}), \quad n = \overline{0, N-1}, \quad i = \overline{0, k-1} \quad (4)$$

В качестве неизвестных случайных величин, которые будут вершинами графа, выберем значения функции на сетке y_i^n . Их начальные распределения зададим нормальными с параметрами ниже.

$$y_i^n \sim N(0, 100), \quad n = \overline{0, N}, \quad i = \overline{0, k}$$

Наложим ограничения на узлы в виде условий, полученных из разностной схемы (4). Основная идея — взять невязку схемы и постулировать ее эквивалентность некой известной фиксированной случайной величине с нулевым математическим ожиданием и малой дисперсией $\sigma = 0.0001$. Получается, предположением модели является наличие малой погрешности между каждым слоем схемы.

$$(1 + 2\gamma)y_i^{n+1} - y_i^n - \gamma(y_{i-1}^{n+1} + y_{i+1}^{n+1}) \sim N(0, \sigma), \quad n = \overline{0, N-1}, \quad i = \overline{0, k-1}$$

Аналогичным способом задаются граничные условия.

$$y_i^n \sim N(0, \sigma), \quad n = \overline{0, N}, \quad i = \overline{0, k}$$

В этой модели прогнозировались значения функции на сетке в нулевой момент времени y_0^n . Для вычислений по графовой модели использовался алгоритм *Variational message passing*, реализованный в программном пакете *Infer.Net*. Количество итераций зависело от размера сетки задачи.

В эксперименте графовая модель сравнивалась с регуляризацией Тихонова, которая заключалась в минимизации функционала с $\alpha = 0.01$ (был взят на основе метода невязки в зависимости от погрешности входных данных).

$$J(v) = \|Av - \phi\|^2 + \alpha \|v\|^2 \xrightarrow{v} \inf, \text{ где } \|f\|^2 = \sum_{i=0}^k f_i^2 h$$

Вычисления производились для трех различных функций начального распределения температуры.

$$\psi_1(x) = \frac{10}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-0.5}{0.1})^2}, \quad \psi_2(x) = \operatorname{sh} \pi \sin \pi x, \quad \psi_3(x) = \begin{cases} 1, & x \in (0.4, 0.6) \\ 0, & x \notin (0.4, 0.6) \end{cases} \quad (5)$$

К функциям прибавлялся случайный гауссов шум с дисперсией $\sigma = 0.01$. Задача рассматривалась при $x \in [0, 1]$, $t \in [0, 0.001]$ с шагами сеток $h = 0.01$, $\tau = 5 * 10^{-5}$. Метрикой сравнения графовой модели и регуляризации Тихонова выступила нормированная аппроксимация L_2 -нормы разницы точного решения y и предсказанного $\hat{y}$.

$$\|y - \hat{y}\|^2 = \frac{\sum_{i=0}^k (y_i - \hat{y}_i)^2 h}{\sum_{i=0}^k y_i^2 h} \quad (6)$$

функция	графовая модель	регуляризация Тихонова
ψ_1	$1.526 * 10^{-5}$	$1.963 * 10^{-4}$
ψ_2	$2.216 * 10^{-4}$	$1.202 * 10^{-4}$
ψ_3	$3.815 * 10^{-2}$	$6.225 * 10^{-2}$

Табл. 3: Значения метрики после эксперимента

Результаты эксперимента представлены в Табл. 3.

Обратная задача для уравнения Пуассона. Рассмотрим обратную задачу для двумерного уравнения Пуассона в прямоугольнике $[0, a] \times [0, a]$.

$$\begin{cases} u_{xx}(x, y) + u_{yy}(x, y) = f(x)g(y) \\ u(0, y) = u(a, y) = 0 \\ u(x, 0) = u(x, a) = 0 \end{cases} \quad x \in [0, a], \quad y \in [0, a] \quad (7)$$

Пусть помимо граничных условий известны $g(y)$, $g(0) = g(a) = 0$, а также значения производных на нижней и верхней границах $\phi_1(x) = u_y(x, 0)$ и $\phi_2(x) = u_y(x, a)$. Требуется найти функцию $f(x)$ при таких предположениях.

Для построения графовой модели рассмотрим пятиточечную разностную схему (умноженную на h^2).

$$y_{m-1}^n + y_{m+1}^n + y_m^{n-1} + y_m^{n+1} - 4y_m^n = f_m g_m h^2, \quad m, n = \overline{1, N-1} \quad (8)$$

Теперь выпишем аппроксимацию условий $\phi_1(x) = u_y(x, 0)$ и $\phi_2(x) = u_y(x, a)$ на верхней и нижней границах.

$$\frac{y_m^1 - y_m^0}{h} + \frac{y_{m-1}^0 - 2y_m^0 + y_{m+1}^0}{2h} = \phi_{1,m}, \quad m = \overline{1, N-1} \quad (9)$$

$$\frac{y_m^n - y_m^{n-1}}{h} + \frac{y_{m-1}^n - 2y_m^n + y_{m+1}^n}{2h} = \phi_{2,m}, \quad m = \overline{1, N-1} \quad (10)$$

Неизвестными случайными величинами, являющимися вершинами графа, выберем значения сеточной функции y_m^n . Их начальные распределения зададим нормальными с параметрами ниже.

$$y_i^n \sim N(0, 100), \quad m, n = \overline{0, N}$$

В качестве основы для связей в графе использовались соотношения (8), (9), (10). Для вычислений был выбран алгоритм *Expectation propagation*, каждый прогноз был основан на 275 итерациях алгоритма (подбиралось под $h = 0.01$).

Для эксперимента при $x, y \in [0, 1]$ с шагом сетки $h = 0.01$ были выбраны 3 функции $f(x)$, задаваемые (5). Метрика имела вид (6). В качестве известной функции $g(y)$ была выбрана

$$g(y) = -|x - 0.5| + 0.5$$

Данные не зашумлялись.

функция	графовая модель
ψ_1	$2.868 * 10^{-4}$
ψ_2	$1.965 * 10^{-4}$
ψ_3	$6.680 * 10^{-3}$

Табл. 4: Значения метрики после эксперимента

Результаты эксперимента представлены в Табл. 4.

Заключение. В данной работе было рассмотрено три обратных задачи и для каждой из них была построена графовая модель.

- Модель, решающая обратную задачу для системы Лотки—Вольтерры, показала себя лучше, чем метод минимизации невязки.
- Модель, решающая обратную задачу для уравнения теплопроводности, дала результаты, сопоставимые по качеству с регуляризацией Тихонова.
- Модель, решающая обратную задачу для уравнения Пуассона, смогла строить предсказания с высокой точностью.

ЛИТЕРАТУРА

- [1] Thomas P. Minka Expectation propagation for approximate Bayesian inference // Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence : Morgan Kaufmann Publishers Inc., 2001. P. 362–369.
- [2] John Winn, Christopher M. Bishop Variational Message Passing // Journal of Machine Learning Research. 2005. Vol. 6, N 23. P. 661–694.
- [3] Infer.NET 0.3 / T. Minka, J. Winn, J. Guiver, Y. Zaykov, D. Fabian, J. Bronskill : Microsoft Research Cambridge, 2018. <http://dotnet.github.io/infer>
- [4] Тихонов А. Н. О регуляризации некорректно поставленных задач // Докл. АН СССР. 1963. Т. 153, № 1. С 49–52

Нейросетевой метод построения панорамных микроскопических изображений геологических образцов

Работа удостоена диплома 1 степени

Николаев Глеб Витальевич

Кафедра математической физики

e-mail: s02210493@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. н.с. Хвостиков Александр Владимирович

В задачах геологии, например, при разведке месторождений, часто необходимо исследовать геологические образцы, в частности аншлифы — образцы диаметром в несколько сантиметров, отполированные для изучения в отраженном свете под поляризационным микроскопом. Данный анализ — очень трудоемкая задача, требующая от исследователя большой экспертизы. Ведется активная работа над автоматизацией исследований аншлифов, направленная на повышение скорости и качества такого анализа.

На типичном для таких задач увеличении угол обзора микроскопа покрывает лишь малую часть аншлифа, что негативно сказывается на качестве анализа. Для исследования областей большей площади существуют электронные сканирующие микроскопы, однако высокая стоимость такого оборудования ограничивает его доступность. Поэтому исследователи часто отдают предпочтение программной сборке панорамных изображений из набора отдельных изображений, снятых с перекрытием. Данная выпускная работа посвящена разработке алгоритма построения панорамных микроскопических изображений аншлифа по набору снимков. Рис. 1

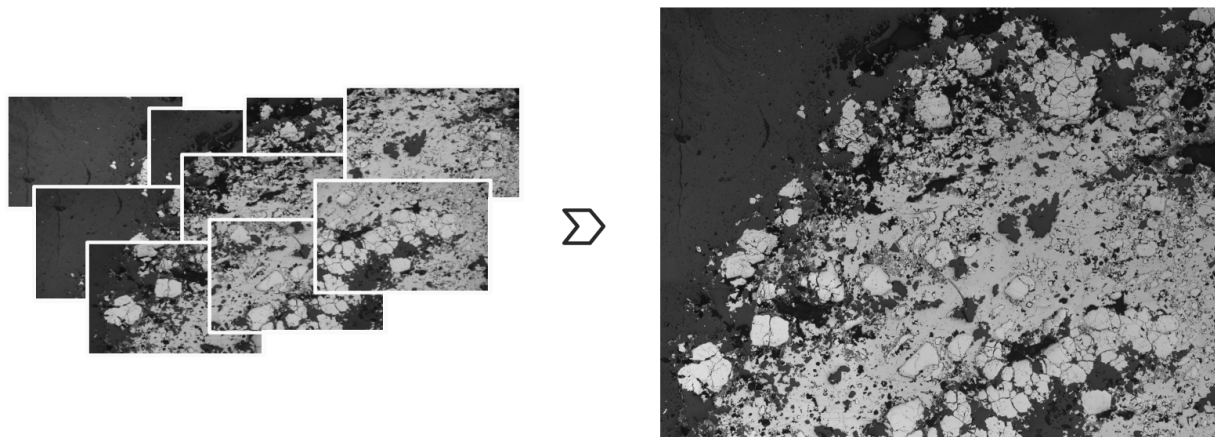


Рис. 1: Идея метода. Серия изображений аншлифа (слева), панорама из этих изображений (справа).

Для различных задач автоматического анализа в геологии лабораторией математических методов обработки изображений факультета ВМК МГУ и кафедрой геологии, геохимии и экономики полезных ископаемых геологического факультета МГУ была собрана и размечена открытая коллекция изоб-

ражений аншлифов LumenStone *. Специально для данной работы коллекция была дополнена набором P1, состоящим из 33 серий, по 20-25 изображений. Каждая серия состоит из фотографий одного аншлифа, сделанных вручную с перекрытием 30-50% и расположенных в виде сетки из нескольких строк и нескольких столбцов. Этот набор используется для иллюстраций и тестирования разработанного алгоритма.

В ходе выпускной работы был разработан и реализован метод построения панорамных изображений аншлифов, учитывающий особенности микроскопической съемки.

Метод состоит из двух основных этапов:

1. построение первичной панорамы;
2. улучшение панорамы.

На первом этапе на изображениях набора корректируются геометрические и фотометрические искажения, вызванные особенностями съемки. После чего все изображения преобразуются в единые координаты. Результатом работы этого этапа является первичная панорама (коллаж) из обработанных входных изображений. При этом коллаж содержит заметные стыки между изображениями. На этапе улучшения панорамы используется информация из областей наложения соседних изображений, что позволяет сделать переходы между изображениями менее заметными. Результат работы этого этапа — финальная панорама.

Этап 1. Построение первичной панорамы

Перед построением панорамы необходимо устранить геометрические и фотометрические искажения, вызванные особенностями используемой оптической системы. Эти искажения могут быть не очевидны при визуальном осмотре отдельных изображений, но существенно ухудшают качество совмещения изображений и приводят к артефактам на итоговой панораме.

Объектив камеры вызывает дисторсию — эффект бочкообразного, подушкообразного или более сложного искривления изображения по мере удаления пикселей от оптической оси фотоаппарата. Дисторсия затрудняет корректное совмещение изображений и приводит к искривлению панорамы. Объектив используемой камеры соответствует модели камеры обскура, для описания которой используют модель дисторсии Брауна-Конради [1]. В ней дисторсия делится на радиальную и тангенциальную. Влияние радиальной дисторсии на изображение описывается формулами:

$$\begin{aligned}x_{\text{corr}} &= x \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \right), \\y_{\text{corr}} &= y \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \right),\end{aligned}$$

где r — расстояние до центра изображения, k_1 , k_2 , k_3 — коэффициенты дисторсии. В данной задаче предполагается, что оптическая ось перпендикулярна поверхности аншлифа, поэтому эффекта тангенциальной дисторсии на

*<https://imaging.cs.msu.ru/en/research/geology/lumenstone>

данных изображениях нет. Для компенсации дисторсии используется калибровочный шаблон — фотография OLED-дисплея смартфона, на ней центры субпикселей зеленого канала образуют регулярную равномерную сетку. После вычисления коэффициентов дисторсии с помощью калибровочного шаблона изображения корректируются.

Съемка аншлифов проводится в отраженном свете с использованием встроенного источника освещения микроскопа. Это приводит к неравномерной освещенности образца, кроме того, оптическая система добавляет эффект виньетирования. Эти эффекты приводят к появлению в панораме заметных неестественных переходов между изображениями. Яркость изображения I моделируется как произведение истинного изображения L и карты освещенности M :

$$I = M \cdot L.$$

Для компенсации освещенности используется метод коррекции плоского поля (flat-field correction), включающий съемку двух калибровочных изображений: плоское поле W — изображение равномерно отражающей поверхности, в данной работе используется фотография зеркала; и черное поле B — изображение при закрытом объективе, учитывающее шумы камеры. Тогда фотометрическая коррекция осуществляется по формуле:

$$I_{\text{corr}} = \overline{W - B} \cdot \frac{I - B}{W - B},$$

где $\overline{W - B}$ — средняя яркость откалиброванного плоского поля.

Ключевым шагом в сборке панорамы является совмещение изображений — приведение всех изображений к общей системе координат, так чтобы области перекрытия на изображениях совпали. В предложенном методе применяется современный нейросетевой трансформерный метод LoFTR [2], который находит на двух изображениях соответствующие друг другу точки. Изображения в панораме связаны друг с другом проективным преобразованием (гомографией), задающимся матрицей 3×3 :

$$H = \begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ c_1 & c_2 & 1 \end{pmatrix}$$

Для нахождения матрицы гомографии по набору соответствий применяется алгоритм DLT, а для фильтрации выбросов применяется RANSAC.

Разработанный алгоритм вычисляет всевозможные попарные гомографии изображений с их соседями в панораме. После чего, последовательно применяя их, приводит все изображения в координаты опорного изображения. В качестве опорного изображения берется изображение, имеющее наибольшее число соответствий с другими изображениями панорамы.

После получения всех гомографий, для повышения точности алгоритм реализует минимизацию среднеквадратичного функционала ошибки репроекции:

$$J = \sum_{(j,j')} \sum_i \|H_j p_i - H_{j'} p'_i\|^2 \rightarrow \min.$$

Где p_i и p'_i — соответствующие друг другу точки изображений I_j и $I_{j'}$, H_j и $H_{j'}$ — гомографии, приводящие эти изображения в финальные координаты. Минимизация проводится методом Левенберга — Марквардта, в качестве начального приближения выбираются гомографии, полученные на предыдущем этапе. После такой минимизации среднеквадратичная ошибка совмещения панорамы становится в районе 0.5–0.6 пикселя для всех используемых серий изображений.

Этап 2. Устранение артефактов склейки

Первичная панорама, полученная после совмещения изображений, содержит заметные стыки между частями панорамы, появляющиеся из-за особенностей съемки, например, не зафиксированная экспозиция на разных кадрах одной панорамы. Для исправления таких эффектов применяются следующие техники:

1. Глобальная компенсация освещенности.

В ходе глобальной компенсации освещенности для каждого изображения панорамы находится коэффициент, на который умножаются интенсивности пикселей изображений. Это позволяет затемнить или высветлить изображения в зависимости от экспозиции на данном кадре. Для нахождения этих коэффициентов предложенный метод минимизирует следующий функционал по всем областям наложения изображений:

$$J(\alpha) = \sum_{k \in \{R, G, B\}} \sum_{i=1}^N \sum_{j=1}^N \left((1 + \alpha_i) S_{ijk}^i - (1 + \alpha_j) S_{ijk}^j \right)^2 \rightarrow \min_{\{\alpha_i\}},$$

где S_{ijk}^i — сумма интенсивностей в k -ом канале в области перекрытия изображений i и j , взятая с изображения i , $\alpha_i \in [-0.3, 0.3]$. Для избежания общего затемнения или высветления панорамы средняя яркость всей сцены сохраняется до начала оптимизации и восстанавливается в конце.

2. Построение оптимальных швов методом разреза графа.

В области наложения двух изображений можно часть пикселей выбрать от одного изображения, а часть от второго, при этом разделение можно произвести так, чтобы шов был наименее заметен. Для этого применяется метод разреза графа [3]. Область пересечения представляется в виде 4-связного графа. В котором вершинам соответствуют пиксели, а веса на ребрах задаются по формуле:

$$E(s, t) = \|I_1(s) - I_2(s)\| + \|I_1(t) - I_2(t)\| \\ + \|\nabla I_1(s) - \nabla I_2(s)\| + \|\nabla I_1(t) - \nabla I_2(t)\|,$$

Для ускорения работы этого шага в данной работе предлагается двухэтапный подход: на пониженном разрешении строится грубый шов, после чего строится уточненный шов в небольшой области вокруг грубого шва.

3. Смешивание изображений.

Для устранения остаточных артефактов на швах применяется финальное смешивание. Для избежания размытия высокочастотных объектов изображения раскладываются по частотам и смешиваются на каждой частоте по-своему. Для низких частот смешивание происходит в широкой полосе вблизи шва, а для высоких частот в узкой [4]:

$$I_{\sigma_{k+1}}^i = I_{\sigma_k}^i * g_{\sigma_{k+1}}; \\ W_{\sigma_{k+1}}^i = W_{\sigma_k}^i * g_{\sigma_{k+1}}; \\ B_{\sigma_{k+1}}^i = I_{\sigma_k}^i - I_{\sigma_{k+1}}^i; \\ I_{\sigma_k}^{\text{multi}}(x, y) = \frac{\sum_{i=1}^n B_{\sigma_k}^i(x, y) W_{\sigma_k}^i(x, y)}{\sum_{i=1}^n W_{\sigma_k}^i(x, y)}.$$

где $I^i(x, y)$ — интенсивность i -го изображения, $W^i(x, y)$ — весовая функция, $B_{\sigma_k}^i$ — частотная компонента изображения I^i , $g_{\sigma}(x, y)$ — гауссов фильтр с параметром сглаживания, определяемым среднеквадратичным отклонением σ , а символ $*$ обозначает операцию свёртки.

Результаты

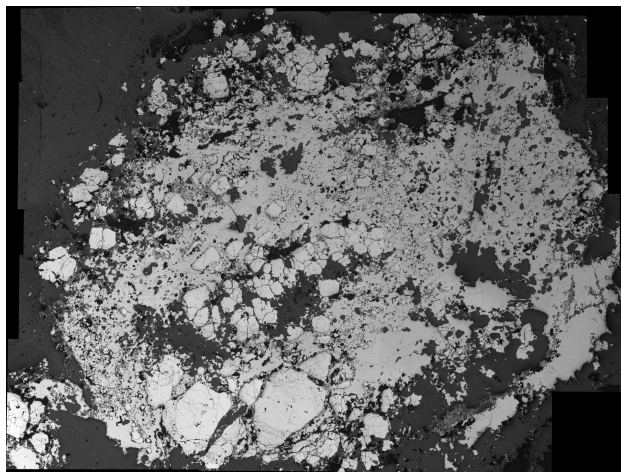
Для всех 33 серий набора LumenStone P1 предложенным методом успешно были построены панорамы [†]. Примеры полученных панорам приведены на Рис. 2.

Для оценки визуального качества склейки произведено сравнение артефактов на панорамах, построенных предложенным методом и другими существующими решениями (Adobe Photoshop, Fiji/ImageJ, Panorama Studio). Были выявлены следующие артефакты:

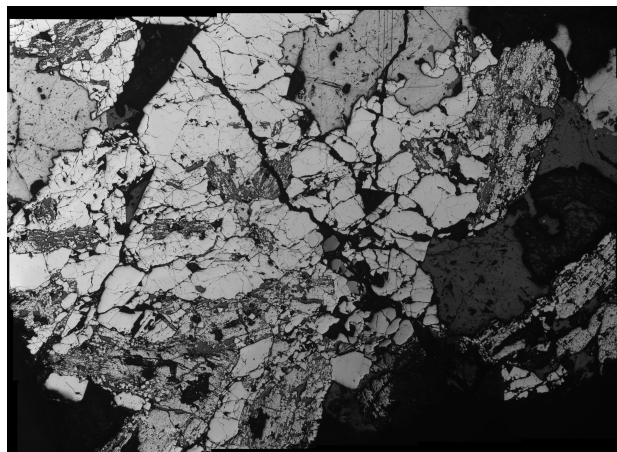
- Fiji: сильное размытие в областях наложения;
- Photoshop: изломы линий на стыках и излишнее преобразование цвета;
- PanoramaStudio: общее изменение яркости в панораме;
- Предложенный метод: минимальные артефакты.

Сравнение показало, что метод не уступает современным профессиональным решениям по качеству панорам, а, учитывая специфику микроскопии,

[†]примеры исходных изображений и результирующих панорам размещены на сайте <https://imaging.cs.msu.ru/en/research/geology/lumenstone>



Панорама из 25 изображений, итоговое разрешение 12231×9317



Панорама из 25 изображений, итоговое разрешение 10925×7891

Рис. 2: Примеры панорам, полученных в результате работы разработанного метода на некоторых сериях коллекции LumenStone P1.

часто и превосходит. Предварительные результаты работы были доложены на международной конференции PDA24 и опубликованы в статье [5] (WoS, Scopus). Реализованный метод доступен в рамках открытой Python библиотеки для анализа геологических изображений *petroscope*[‡].

ЛИТЕРАТУРА

- [1] Brown D. C. Decentering distortion of lenses // Photogrammetric Engineering. 1966. Vol. 32, N 3. P. 444–462.
- [2] LoFTR: Detector-free local feature matching with transformers / Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, Xiaowei Zhou // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2021. P. 8922–8931.
- [3] Boykov Y., Kolmogorov V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2004. Vol. 26, N 9. P. 1124–1137.
- [4] Brown M., Lowe D. G. Automatic panoramic image stitching using invariant features // International Journal of Computer Vision. 2007. Vol. 74, P. 59–73.
- [5] Nikolaev G., Korshunov D., Khvostikov A. Automatic stitching of panoramas for geological images of polished sections // ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2024. Vol. X-2/W1-2024. P. 39–46.

[‡]<https://pypi.org/project/petroscope/>

Алгоритм имитации отжига для планирования вычислений с минимизацией энергопотребления

Осотов Степан Романович

Кафедра автоматизации систем вычислительных комплексов

e-mail: osotovsr@my.msu.ru

Научный руководитель — к.т.н. доц. Костенко Валерий Алексеевич

Научный консультант — к.ф.-м.н. с.н.с. Балашов Василий Викторович

В настоящее время в системах стремительно растет количество задач, которые нужно выполнить системе, их сложность, потребность в выполнении этих задач в определенные сроки. Однако пропорционально росту указанных факторов растет энергопотребление в системах. В связи с этим становится актуальным вопрос энергосберегающего планирования вычислений, в частности, построение списочных расписаний.

Общая задача построения списочных многопроцессорных расписаний заключается в следующем. Задан ориентированный ациклический граф потока данных. Вершинами графа являются работы. Для каждой работы задано время, требуемое для ее выполнения. Дуги задают отношение частичного порядка на множестве работ. Если от i -ой вершины идет дуга к j -ой вершине, то работа j может начать выполняться только после завершения работы i . Требуется построить расписание выполнения работ на заданном числе процессоров, а именно определить привязку и порядок. Привязка - всюду определенная на множестве работ функция, которая задает их распределение по процессорам. Порядок устанавливает ограничения на последовательность выполнения работ и является отношением частичного порядка, удовлетворяющим условиям ацикличности и транзитивности. Отношение порядка на множестве работ, распределенных на один и тот же процессор, представляет отношение полного порядка. Расписание корректно если выполняются следующие ограничения:

- каждая работа должна быть назначена на процессор;
- любую работу обслуживает лишь один процессор;
- частичный порядок, заданный в графе потока данных, сохранен в расписании.

В данной работе в качестве критерия оптимальности расписания рассматривается энергопотребление необходимое для его выполнения и задается ограничения на допустимое время выполнения расписания.

Целью работы является разработка и исследование свойств алгоритма имитации отжига для построения списочных расписаний с минимизацией энергопотребления на различных классах исходных данных. Также требуется сравнить последовательную и параллельную версию алгоритма по точности найденного решения и скорости выполнения программ, изучить поведение алгоритма при использовании различных законов понижения температур.

Для построения списочных расписаний с минимизацией энергопотребления рассматривается алгоритм имитации отжига: последовательная [1] и параллельная версия алгоритма [2]. В качестве механизма изменения и расчета энергопотребления в системе взят механизм изменения тактовой частоты процессора [3].

Введем HP^* - множество всех корректных расписаний. Также введем вероятность изменения расписания как:

$$p = e^{-\frac{\Delta F}{T}} \quad (1)$$

Тогда последовательный алгоритм имитации отжига состоит из следующих шагов:

1. Задать начальное корректное расписание $X_0 \in HP^*$ и считать его текущим вариантом решения: $X = X_0$.
2. Установить начальную температуру T_0 , приняв ее текущей: $T = T_0$.
3. Применить операции преобразования решения к X и получить новый вариант решения $X' \in HP^*$. Если это решение является лучшим из ранее найденных, то запомнить его.
4. Найти изменение функции оценки качества решения: $\Delta F = F(X') - F(X)$:
 - если $\Delta F \leq 0$
 - иначе с вероятностью (1) принять в качестве текущего решения новый вариант решения X'
5. Повторить заданное число раз шаги 3 и 4 без изменения текущей температуры.
6. Если критерий останова выполнен, то завершение работы.
7. Понизить текущую температуру перейти к шагу 3.

Параллельная версия алгоритма представляет собой $n \in \mathbb{N}$ параллельно выполняющихся последовательных версий алгоритма, которые раз в заранее заданное количество итераций обмениваются своими лучшими решениями X_{best} и устанавливают лучшее из них в качестве текущего решения X . В работе рассматривается 2 версии параллельного алгоритма: с использованием системных вызовов *fork* и средств *MPI*.

Экспериментальное исследование алгоритма проводилось на случайных направленных ациклических графах. Для каждого графа заранее известно оптимальное значение целевой функции - энергопотребления. Ниже будем обозначать количество работ (вершин в графе потока данных) через m . Тогда входные данные можно разбить на следующие классы по количеству работ:

- маленькие: $m \in \{10, 20, 30, 40, 50\}$
- средние: $m \in \{100, 200, 300, 400, 500\}$
- большие: $m \in \{1000, 2000, 3000, 4000\}$

Таким образом, в результате исследования алгоритма были получены следующие результаты:

- последовательная версия алгоритма работает быстрее параллельных версий, но показывает хуже значения целевой функции;
- параллельная версия алгоритма с использованием системных вызовов `fork` работает быстрее версии с использованием средств MPI;
- закон Коши понижения температуры в среднем показывает лучшие результаты, в то время как общий закон понижения температуры является самым быстрым. Закон Больцмана может показать как лучшие результаты, так и худшие, таким образом, он является наименее предсказуемым.

ЛИТЕРАТУРА

- [1] Д. А. Зорин, В. А. Костенко. Алгоритм имитации отжига для решения задач построения многопроцессорных расписаний // Автоматика и телемеханика, 2014, № 10, С. 97–110.
- [2] А. В. Калашников, В. А. Костенко. Параллельный алгоритм имитации отжига для построения многопроцессорных расписаний // Известия РАН. Теория и системы управления. 2008. № 3. С. 133–142.
- [3] Д. В. Костыгов. Энергоэффективные вычислительные системы реального времени на кристалле // Диссертация на соискание учёной степени кандидата технических наук, «Концерн «ЦНИИ «Электроприбор», 2020.

Исследование влияния применения размеченных данных в алгоритмах построения признаков пространств

Работа удостоена диплома 2 степени

Панкина Алина Алексеевна

Кафедра Суперкомпьютеров и Квантовой Информатики

e-mail: s00210505@cs.msu.ru

Научный руководитель — к.ф.-м.н. доц. Буряк Дмитрий Юрьевич

В настоящее время нейронные сети и другие алгоритмы машинного обучения используются регулярно и находят широкое применение в самых различных областях. С ростом объёма доступных наборов данных — как с точки зрения количества выборок, так и числа признаков в каждой выборке — возрастает важность эффективного представления информации. Одной из ключевых задач становится снижение размерности данных с сохранением их информативности.

Для решения данной задачи активно развиваются методы обучения представлению данных. Они используются для построения компактных и информативных признаков пространств с целью уменьшения нагрузки на вычислительные ресурсы и повышения качества последующей работы алгоритмов.

В рамках работы был проведён обзор методов обучения представлению данных, в котором были выделены три основные группы подходов:

1. Методы, основанные на обучении без учителя;
2. Контрастное обучение;
3. Обучение с учителем;

Для дальнейшего исследования был выбран метод извлечения признаков на основе обучения без учителя – VICReg (Variance-Invariance-Covariance Regularization) [1]. Данный подход позволяет обучать нейронную сеть для генерации информативных пространств представлений без использования размеченных данных. Принцип работы данного метода заключается в регуляризации выходных представлений по трём параметрам: минимизация инвариантности между аугментациями входных данных, увеличение дисперсии признаков для предотвращения их схождения в одну точку, а также снижение избыточности признаков за счёт декорреляции каналов. Совместно эти компоненты обеспечивают устойчивое обучение для генерации компактных и хорошо структурированных представлений.

Основная задача работы заключалась в исследовании применения метода VICReg с использованием размеченных данных. Наличие меток в данном методе влияет на формирование пакетов обучающих данных и на функцию потерь. В ходе работы были разработаны и реализованы два метода формирования пакетов данных и две новые функции потерь для метода VICReg с использованием меток классов.

Для формирования пакетов данных в методе VICReg используются аугментации. Они позволяют сформировать входные пары X, X' , состоящие из двух разных представлений исходного объекта.

Разработанные методы предлагают следующие подходы:

1. Использование меток классов для формирования входных пар X_1, X_2 , состоящих из двух разных объектов, принадлежащих одному классу. Обозначим данный метод M_1 .
2. Применение аугментаций и использование меток классов для формирования входных троек X_1, X'_1, X_2 , состоящих из двух разных представлений одного объекта и любого другого объекта, принадлежащего тому же классу. Обозначим данный метод M_2 .

Функция потерь метода VICReg представляет собой сумму трёх компонент-функций: дисперсии – $v(Z)$, инвариантности – $s(Z, Z')$, ковариации – $c(Z)$, и выглядит следующим образом:

$$L_{VICReg} = \sum_{i \in D} \sum_{t' \in T} [\lambda s(Z, Z') + \mu \{v(Z) + v(Z')\} + \nu \{c(Z) + c(Z')\}]$$

где D - исходная выборка, T - применённые аугментации, Z и Z' - пары эмбедингов, полученные на выходе из энкодера, λ, μ, ν - положительные гиперпараметры, введённые для регуляризации каждого из слагаемых функции.

Разработанные функции потерь предлагают следующее:

1. Использовать функцию L_{VICReg} , заменив в ней компоненту инвариантности $S(Z, Z')$ на кросс-энтропию. Обозначим данную функцию L_1 .
2. Использовать функцию L_{VICReg} , добавив к ней дополнительно компоненту кросс-энтропии. Обозначим данную функцию L_2 .

Полученные функции выглядят следующим образом:

$$\begin{aligned} L_1 &= \sum_{i \in D} \sum_{t' \in T} \left[\mu \{v(Z) + v(Z')\} + \nu \{c(Z) + c(Z')\} \right. \\ &\quad \left. + \eta \{ce(Pr, M_a) + ce(Pr', M_a)\} \right] \\ L_2 &= \sum_{i \in D} \sum_{t' \in T} \left[\lambda s(Z, Z') + \mu \{v(Z) + v(Z')\} \right. \\ &\quad \left. + \nu \{c(Z) + c(Z')\} + \eta \{ce(Pr, M_a) + ce(Pr', M_a)\} \right] \end{aligned}$$

где Pr и Pr' — предсказания линейного классификатора для эмбедингов Z и Z' , $ce(Pr, M_a)$ — функция кросс-энтропии, η — положительный гиперпараметр для регуляризации кросс-энтропии.

В результате применения разработанных методов и функций было получено 9 схем обучения энкодеров, включая базовую реализацию метода VICReg.

Для оценки эффективности разработанных подходов было проведено экспериментальное сравнение с оригинальным методом VICReg и с классическим методом обучения с учителем, который заключается в обучении нейронной сети для задачи классификации и последующее использование сети без последнего слоя для генерации представлений.

Для экспериментального сравнения обученных энкодеров использовалось тестирование линейной оценкой, которое заключается в обучении линейных классификаторов, принимающих на вход предсказанные энкодерами представления. Для оценки точности классификации использовалась метрика $Accuracy = \frac{N_{correct}}{N_{total}}$, где $N_{correct}$ - количество правильно классифицированных объектов, N_{total} - общее количество объектов.

Эксперименты проводились на базах CIFAR-10 и CIFAR-100 [2]. В качестве архитектуры энкодера была выбрана сверточная сеть ResNet-18 [3].

В таблицах 13, 14 представлены точности классификации баз CIFAR-10, CIFAR-100:

Схема эксперимента	Метод формирования пакетов	Функция потерь	Точность Top-1
VICReg	M_{VICReg}	L_{VICReg}	69.70%
A	M_{VICReg}	L_1	68.20%
B	M_{VICReg}	L_2	86.24%
C	M_1	L_{VICReg}	81.07%
D	M_1	L_1	70.24%
E	M_1	L_2	77.73%
F	M_2	L_{VICReg}	85.79%
G	M_2	L_1	72.28%
H	M_2	L_2	87.10%

Таблица 13: Результаты тестирования на базе CIFAR-10

Для базы CIFAR-100 также проводился эксперимент с разделением по классам, при котором для обучения и тестирования энкодеров база разделялась на две непересекающиеся части по меткам классов. В таблице 15 представлены результаты точности классификации:

Согласно результатам, приведённым в таблицах, лучшее качество демонстрирует модель, обученная по схеме H. Данная схема предполагает использование метода M_2 формирования пакетов данных и функцию потерь L_2 .

Применение предложенных методов и функций позволило разработать подход, с помощью которого удалось повысить точность классификации по сравнению с методом VICReg на 17.4 п.п. для базы CIFAR-10 и на 22.98 п.п. для базы CIFAR-100. По сравнению с методом обучения с учителем удалось повысить точность на 5.3 п.п. для базы CIFAR-100.

Проведенные исследования показали, что применение размеченных данных позволяет существенно повысить эффективность энкодера по сравнению

Схема эксперимента	Метод формирования пакетов	Функция потерь	Точность Тор-1	Точность Тор-5
VICReg	M_{VICReg}	L_{VICReg}	49.96%	70.57%
A	M_{VICReg}	L_1	50.40%	72.65%
B	M_{VICReg}	L_2	64.39%	82.89%
C	M_1	L_{VICReg}	62.12%	82.50%
D	M_1	L_1	56.54%	80.11%
E	M_1	L_2	57.15%	83.08%
F	M_2	L_{VICReg}	63.78%	86.36%
G	M_2	L_1	61.43%	84.55%
H	M_2	L_2	<u>72.94%</u>	<u>96.79%</u>

Таблица 14: Результаты тестирования на базе CIFAR-100

Схема эксперимента	Метод формирования пакетов	Функция потерь	Точность Тор-1	Точность Тор-5
Обучение с учителем	-	CrossEntropy	65.50%	91.77%
VICReg	M_{VICReg}	L_{VICReg}	48.33%	73.14%
A	M_{VICReg}	L_1	51.88%	73.72%
B	M_{VICReg}	L_2	63.10%	84.51%
C	M_1	L_{VICReg}	60.91%	83.48%
D	M_1	L_1	57.43%	79.10%
E	M_1	L_2	58.40%	82.24%
F	M_2	L_{VICReg}	66.06%	90.95%
G	M_2	L_1	60.36%	87.06%
H	M_2	L_2	<u>70.83%</u>	<u>96.68%</u>

Таблица 15: Результаты тестирования на базе CIFAR-100 с разделением по классам

с базовым методом VICReg и получить более высокие результаты, чем при классическом подходе обучения с учителем. Таким образом, предложенные подходы могут быть рекомендованы для построения пространств представлений с использованием размеченных данных.

ЛИТЕРАТУРА

- [1] Bardes A., Ponce J., LeCun Y. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning. – 2022.
- [2] University of Toronto. The CIFAR datasets. // <https://www.cs.toronto.edu/~kriz/cifar.html> - 2009.

-
- [3] Sai Abhishek, Allena Venkata. Resnet18 Model With Sequential Layer For Computing Accuracy On Image Classification Dataset / Allena Venkata Sai Abhishek. — 2022. — 07. — Vol. 10. — Pp. 2320–2882.

Проверка истинности предупреждений статического анализатора с помощью больших языковых моделей

Работа удостоена диплома 3 степени

Панов Данила Дмитриевич

Кафедра системного программирования

e-mail: panovdd@my.msu.ru

Научный руководитель — к.ф.-м.н. доц. Игнатъев Валерий Николаевич

Научный консультант — к.т.н. Шимчик Никита Владимирович

Введение. Одним из важнейших методов поиска ошибок в коде является статический анализ исходного кода. Разработка инструмента статического анализа подразумевает нахождение компромисса между точностью, полнотой и временем выполнения анализа программы, что приводит к появлению ложных предупреждений. Это связано с тем, что высокоточный анализ отличается плохой масштабируемостью на большие проекты, из-за чего появляется необходимость ограничивать чувствительность к путям и к контексту. Наличие ложных срабатываний у детекторов приводит к тому, что при анализе больших проектов пользователю приходится вручную просматривать тысячи предупреждений, чтобы проверить их истинность, что является крайне трудоёмкой задачей.

Большие языковые модели (англ. Large Language Models, LLM) являются новым и перспективным инструментом для повышения точности и полноты статического анализатора. LLM могут заменить ручную проверку предупреждений, используя те же средства, что и эксперт: отчёт статического анализатора и кодовую базу проекта.

На данный момент LLM, основанные на архитектуре трансформер, имеют существенный недостаток в виде ограниченного размера контекстного окна и снижения качества ответов при увеличении размера запросов, из-за чего передавать на вход модели весь код проекта целиком не представляется возможным. В то же время, попытки классифицировать сложные предупреждения (например, о разыменовании `null`) с использованием небольшого контекста (до 40 строк вокруг предупреждения) привели к значительному снижению полноты при отсутствии прироста в точности [1]. Возникает необходимость обдуманного извлечения из проекта релевантного и компактного контекста для каждого предупреждения.

Для преодоления вышеописанной проблемы в работе предлагается использовать различные структуры, генерируемые статическим анализатором в процессе анализа кода. Такие структуры позволяют определять потенциально значимые для анализа фрагменты кода, а также заменять большие определения функций и классов на сводку их важных свойств.

Существующие методы фильтрации предупреждений [2, 3] показывают, что LLM действительно могут успешно применяться для решения данной задачи. Однако предложенные инструменты не демонстрируют одновременно высокие точность и полноту классификации предупреждений на реальных

проектах. Более того, они основываются на использовании закрытых моделей, таких как GPT-4, что может ставить под угрозу конфиденциальность данных пользователей.

Таким образом, *целью* данной работы является повышение точности статического анализа при сохранении полноты за счёт фильтрации предупреждений при помощи больших языковых моделей с открытыми весами с использованием дополнительной информации, специально собранной статическим анализатором.

В первую очередь необходимо определить, какая информация из множества структур, используемых в статическом анализаторе, может быть полезна для извлечения из кода релевантного контекста предупреждений.

Обзор структур из статического анализатора. Из статического анализатора предлагается экспортировать следующие структуры: трассы предупреждений, абстрактные синтаксические деревья (АСД), таблица символов, резюме типов и функций.

Трасса предупреждения представляет собой набор точек в коде и соответствующих им сообщений, по мнению анализатора важных для понимания выданного предупреждения. Поскольку трасса содержит критически важную информацию о предупреждении, разумно использовать точки трассы в качестве основы для извлечения релевантного кода, а соответствующие этим точкам сообщения помещать в код как комментарии.

Абстрактные синтаксические деревья используются для поиска и навигации по коду. Так как в данной работе в качестве минимальной единицы кода рассматривается не строка, а оператор (другими словами, инструкция), АСД позволяют достичь целостности извлекаемого кода.

Под *таблицей символов* понимается база данных, содержащая объявления, определения и использования всех символов в коде. При помощи запросов к этой базе данных возможно определять зависимости по данным, в частности, находить определения интересующих переменных.

Резюме типов и функций представляют собой краткую выжимку полезных для анализа свойств типа или функции и обычно имеют размер, меньший размера соответствующего определения. Резюме типа аккумулирует в себе информацию о всех наследованных от родительских классов свойствах типа, а резюме функции — информацию о всех вызываемых в ней функциях. Таким образом, одно резюме способно заменить целый ряд определений без потери существенной информации.

Схема разработанного метода. В работе была предложена следующая архитектура метода.

Статический анализатор после анализа проекта помимо отчёта с предупреждениями формирует хранилище с рассмотренными ранее структурами с помощью *модуля экспорта информации*. Предупреждения преобразуются в унифицированный формат и передаются вместе с проектом и дополнительной информацией в *модуль генерации запросов*, который формирует для

каждого предупреждения фрагмент кода. Полученные запросы передаются в LLM, и на основании ответов модели формируется отчёт с отфильтрованными предупреждениями.

Модуль экспорта информации, который является частью статического анализатора, отвечает за сбор, преобразование из внутреннего представления и организацию хранения рассмотренных структур. Разработка модуля является специфичной задачей для каждого статического анализатора вследствие их различий в архитектуре, в наличии и внутреннем устройстве структур, в процессе анализа.

Генератор запросов составляет для каждого предупреждения релевантный контекст: извлекает код вокруг точек трассы, загружает определения переменных и добавляет резюме типов и функций. Полученный контекст вставляется в шаблон, содержащий инструкции LLM и дополнительные сведения об анализируемом предупреждении.

Рассматриваемый подход был реализован для проверки предупреждений статического анализатора SharpChecker [4], который используется для поиска ошибок в программах, написанных на языке C#.

Тестирование. Для тестирования предложенного подхода из базы данных размеченных вручную предупреждений, полученных на реальных проектах, статического анализатора SharpChecker был собран датасет, состоящий из 3 типов предупреждений: разыменование `null`, утечка ресурсов и недостижимый код. Оценка производилась с помощью двух открытых LLM:

1. модель Phind/Phind-CodeLlama-34B-v2;
2. рассуждающая модель deepseek-ai/DeepSeek-R1-Distill-Qwen-32B.

Было показано, что составление запросов к LLM при помощи дополнительной информации из статического анализатора в совокупности с использованием рассуждающей модели позволило получить:

- на наборе предупреждений типа «разыменование `null`» точность фильтрации в 88% (прирост 14 процентных пунктов по сравнению с точностью анализатора) при полноте 86%;
- на наборе предупреждений типа «утечка ресурсов» точность фильтрации в 91% (прирост 3 процентных пункта по сравнению с точностью анализатора) при полноте 97%;
- на наборе предупреждений типа «недостижимый код» точность фильтрации в 75% (прирост 2 процентных пункта по сравнению с точностью анализатора) при относительно низкой полноте в 62%; тем не менее, использование дополнительной информации позволило увеличить полноту фильтрации для обеих моделей на 11–27 процентных пунктов по сравнению с подходом, не использующим дополнительную информацию о предупреждении.

Анализ ответов модели на наборе предупреждений типа «утечка ресурсов» позволил выявить два основных паттерна, на которых происходит некор-

ректная классификация предупреждений. В шаблон запроса были добавлены строчки, корректирующие поведение LLM в таких спорных ситуациях, что позволило увеличить точность фильтрации предупреждений ещё на 2 процентных пункта (точность 93% и полнота 97%). Полученный результат указывает на необходимость давать модели инструкции, которые подсказывают ей, какой ответ от неё ожидается в неоднозначных случаях.

Кроме того, был произведён эксперимент по дообучению не рассуждающей модели с помощью низкорангового адаптера LoRA [5]. Набор предупреждений типа «утечка ресурсов» был разделен на обучающую и тестовую части. Для обучения в запрос были добавлены эталонные ответы, состоящие из одного слова: «Yes» или «No» — в зависимости от того, истинно ли предупреждение. Обученная модель с использованием дополнительной информации продемонстрировала прирост точности и полноты в 3 процентных пункта по сравнению с моделью без дообучения (увеличились точность с 78% до 81% и полнота с 91% до 94% при базовой точности анализатора в 77%).

Заключение. В работе был представлен метод фильтрации предупреждений статического анализатора при помощи LLM с использованием дополнительной информации о предупреждениях, собранной статическим анализатором. Подход был реализован в промышленном статическом анализаторе SharpChecker и протестирован на реальных проектах с помощью LLM с открытыми весами. Предложенный подход позволил увеличить точность автоматической проверки истинности предупреждений статического анализатора типов «разыменование null» и «утечка ресурсов» при сохранении высокой полноты, а также увеличить полноту фильтрации предупреждений типа «недостижимый код» по сравнению с подходами, не использующими дополнительную информацию из статического анализатора.

ЛИТЕРАТУРА

- [1] Large language models in source code static analysis / V.N. Ignatyev, N. V. Shimchik, D. D. Panov, A. A. Mitrofanov // 2024 Ivannikov Memorial Workshop (IVMEM). 2024. P. 28–35.
- [2] Effectiveness of ChatGPT for static analysis: How far are we? / Mohammad Mahdi Mohajer, Reem Aleithan, Nima Shiri Harzevili, Moshi Wei, Alvine Boaye Belle, Hung Viet Pham, Song Wang // Proceedings of the 1st ACM International Conference on AI-Powered Software. 2024. P. 151–160.
- [3] Automatically inspecting thousands of static bug warnings with large language model: How far are we? / Cheng Wen, Yuandao Cai, Bin Zhang, Jie Su, Zhiwu Xu, Dugang Liu, Shengchao Qin, Zhong Ming, Tian Con // ACM Transactions on Knowledge Discovery from Data. 2024. Vol. 18, N 7. P. 1–34.
- [4] SharpChecker: Static analysis tool for C# programs / V. K. Koshelev, V. N. Ignatiev, A. I. Borzilov, A. A. Belevantsev // Programming and Computer Software. 2017. Vol. 43. P. 268–276.

-
- [5] Lora: Low-rank adaptation of large language models / Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Che et al. // ICLR. 2022. Vol. 1, N 2. P. 3.

Исследование и разработка векторного представления для задачи разметки программного кода

Работа удостоена диплома 2 степени
Пасконова Людмила Владимировна

Кафедра интеллектуальных информационных технологий
e-mail: paskonovav@gmail.com

Научный руководитель — к.ф.-м.н. доц. Чернов Александр Владимирович

В данной работе предлагается метод автоматической семантической сегментации программного кода с учётом его внутреннего представления. *Сегментация, или разметка, программного кода* – это выделение компонентов (*сегментов*), вносящих наибольший вклад в решение последующей прикладной задачи. Эти сегменты, представляющие собой последовательности подряд идущих токенов, могут быть синтаксическими (конструкции языка) или логическими (фрагменты, выполняющие схожие функции). Кроме того, в ряде задач важно понимать, какую роль сегменты играют в контексте анализа. Для этого, помимо кода программ, необходимо учитывать их *области интерпретации (ОИ)*: формулировки задач, тесты — т.е. дополнительные условия, в рамках которых независимо интерпретируются свойства программ.

Формально *задачу разметки программного кода* можно сформулировать следующим образом. Дано множество программ X на одном языке программирования. Каждая программа $x \in X$ представляется в виде последовательности токенов: $x = \{t_0, t_1, \dots\}$. Для каждой программы задана область интерпретации $\hat{a} \in \hat{A}$ так, что множества программ, относящихся к разным областям интерпретации, не пересекаются:

$$X = \{X_{\hat{a}_0}, X_{\hat{a}_1}, \dots, X_{\hat{a}_{q-1}}\}, \quad \bigcup_{i=0}^{q-1} X_{\hat{a}_i} = X, \quad X_{\hat{a}_i} \cap X_{\hat{a}_j} = \emptyset \quad (i \neq j).$$

Каждой программе можно поставить в соответствие какой-либо класс из множества $\hat{C} = \{\hat{c}_0, \hat{c}_1, \dots, \hat{c}_{n-1}\}$, обозначим $x \in \hat{c}_i$.

Входные данные:

- Область интерпретации программы $\hat{a}$;
- Исходный код программы $x = \{t_0, t_1, \dots, t_{l-1}\} \in X_{\hat{a}}$;

Выход:

- Предсказанный класс программы

$$\hat{y} = \arg \max_{\hat{c} \in \hat{C}} P(y = \hat{c} | \hat{a}, x : x \in X_{\hat{a}})$$

- Дискретная функция, обнаруживающая сегменты, соответствующие предсказанному классу:

$$f : (x, \hat{y}, \hat{a}) \rightarrow \{p_0, p_1, \dots, p_g \in [0 \dots l - 1] \mid \hat{s}_{p_i} \in \hat{y}\}$$

где $\hat{s}_{p_i} = \{t_{p_i}, \dots, t_{p_i+k-1}\}$ – сегмент, началом которого является p_i -й токен.

Однако существующие решения той или иной частной задачи разметки программного кода обладают рядом ограничений. Во-первых, они часто интерпретируют код как обычный текст, хотя эмпирически было получено, что программный код и естественный язык имеют разное распределение [1]. Во-вторых, для работы может требоваться большой объем ручной аннотации — заранее размеченный набор сегментов или подробная информация об области интерпретации [2, 3]. Кроме того, многие методы ограничены одной задачей или языком программирования [4] или не учитывают специфику областей интерпретации [5].

Целью работы является создание программной системы, решающей задачу разметки кода в общем виде, удовлетворяющей следующим требованиям:

1. использовать абстрактное синтаксическое дерево (АСД) для анализа структуры программного кода;
2. работать исключительно на основе входных данных, предоставленных в рамках задачи разметки, без привлечения дополнительных источников информации;
3. обеспечивать возможность анализа кода, относящегося к различным областям интерпретации;
4. поддерживать широкий спектр языков программирования.

Система должна была быть апробирована на следующих задачах:

1. Выявление уязвимостей в программном коде;
2. Поиск ошибок в студенческих программах;
3. Обнаружение программ, сгенерированных большими языковыми моделями.

Набор данных включает исходный код — коллекцию компилируемых программ — и целевую информацию для каждой программы: область интерпретации и класс. Целевая информация получается естественным образом: так, например, для задачи поиска ошибок в студенческих программах ОИ является каждая отдельная задача, а классов два — задача решена верно или неверно. Для построения экспериментальных корпусов используются следующие источники:

1. *Project CodeNet* — массив данных от компании IBM, содержащий примерно 14 миллионов программ на более, чем 50 языках программирования, являющихся решением к одной из 4000 алгоритмических задач.
2. *FormAI Dataset* — коллекция компилируемых программ на языке C, порождённых различными большими языковыми моделями для задач сетевого управления, логических игр и шифрования. Для каждой программы есть информация об уязвимых строках.

3. *NIST Software Assurance Reference Dataset (SARD)* – это набор программ на языках C, C++, Java, PHP, и C# с документированными уязвимостями.

Для каждой программы с помощью инкрементального парсера Tree-Sitter строится синтаксическое дерево (АСД), поддерживающее более 100 языков программирования. Обходом в глубину извлекаются *АСД-токены* — пути от корня дерева к терминалам, представленные как последовательности индексов узлов в грамматике языка. Такая *токенизация* позволяет одновременно учитывать линейный порядок конструкций и их структурный уровень, что важно, например, для различения вложенных присваиваний. Также на этапе предобработки удаляются комментарии.

Архитектура модели — сверточная нейронная сеть с механизмом внимания. На первом этапе АСД-токены превращаются в векторные представления, *эмбединги*. Рассматриваются два их варианта: использование предобученных Word2Vec представлений и инициализация эмбедингов произвольными значениями с дальнейшим их обучением вместе с сетью. Несколько подряд идущих эмбедингов в результате применения операции свертки формируют сегмент.

Механизм внимания, получив на вход ОИ и последовательность сегментов, оценивает важность $\tilde{\alpha}_i$ каждого из них. Взвешенная сумма свёрток – векторное представление всей программы – пройдя через полносвязный слой, позволяет предсказать её класс $\hat{y}$ внутри данной ОИ. Сегменты, веса внимания которых превышают экспериментально подобранный порог, считаются значимыми для предсказанного класса.

Таким образом, искомая функция f — множество позиций токенов, соответствующих значимым сегментам.

$$f(x, \hat{y}, \hat{a}) = \{p_0, p_1, \dots, p_g \in [0 \dots l - 1] \mid \tilde{\alpha}_{p_i} \geq \text{threshold}\} \quad (1)$$

Для оценки качества сегментации (*pred*) для тех наборов данных, где есть информация о разметке отдельных сегментов (*gt*), предложена метрика *RSDM* на основе расстояний между множествами:

$$RSDM(pred, gt) = 1 - \frac{\sum_{i=0}^{M-1} \min_{j \in \{0 \dots K-1\}} |pred_i - gt_j|}{\text{worst_distance}(L, gt)} \quad (2)$$

Важно отметить, что разметка, если она доступна, используется исключительно на этапе оценки модели, но не в процессе обучения. Модели обучались на сервере кафедры интеллектуальных информационных технологий с применением графических ускорителей.

Исследование предобученных Word2Vec представлений для *задачи поиска ошибок в студенческих программах* показало, что АСД-токены, имеющие общие префиксы, склонны группироваться в один кластер. Эти кластеры, во первых, имеют нитевидную форму, а во-вторых, задают различные синтаксические конструкции: функции, классы, конструкторы. Анализ отдельных

сверток продемонстрировал, что для каждой области интерпретации выделялись характерные ей сегменты: ввод или вывод, многомерные массивы. В результате экспериментов с различными гиперпараметрами было получено, что АСД-токенизация в данном случае превосходит обычную в качестве *классификации* (Для оценки использовалась метрика F_1).

Для задачи выявления уязвимостей в программном коде в силу наличия в наборе данных для каждой программы множества уязвимых строк, сгенерированных статическим анализатором, с помощью *RSDM* можно было оценить качество *сегментации*. Эксперименты продемонстрировали, что в случае одной области интерпретации АСД-токенизация сыграла меньшую роль: пословная токенизация демонстрировала близкие результаты метрики. Кроме того, каждый токен считал важными для себя примерно 15 своих соседей – именно такое значение длины сегмента было у оптимальной модели. Интересно, что то же значение показал и анализ векторных представлений. Семантический порог был найден экспериментально и составил $threshold \approx 0.5$.

Для задачи обнаружения программ, сгенерированных большими языковыми моделями, модель на тестовом наборе данных продемонстрировала $F1 = 0.99$! Однако столь высокий показатель вызван, скорее, недостатками используемого набора данных FormAI: в нем много примеров, где большие языковые модели, в отличие от человека, почти для всех программ подробно печатают информацию о своих действиях.

Таким образом, с учетом выдвигаемых требований в общем случае решена задача разметки программного кода:

1. Представлен метод токенизации программного кода на основе АСД: требуя больших размеров словарей по сравнению с пословной токенизацией, он тем не менее улучшает качество сегментации программного кода, причем в случае множественных ОИ, – значительно.
2. Для работы метода нужен лишь набор программ, разделенных на области интерпретации. Представленная в рамках работы метрика *RSDM* позволяет оценить качество полученной сегментации в том случае, когда в тестовом наборе данных доступна разметка-эталон.
3. Метод учитывает специфику областей интерпретации программ.
4. Поддержка множества языков программирования достигается за счёт свойств инкрементального парсера и независимого извлечения АСД-токенов из синтаксического дерева.

Результаты работы реализованы в виде программной платформы *ASTCODA*[§]. Система прошла успешную апробацию на трёх прикладных задачах. В дальнейшем планируется её расширение на другие языки и области интерпретации.

[§]Abstract Syntax Tree Convolutions Operating on Domain Attention (ASTCODA) <https://github.com/Liudmila-Paskonova/ASTCODA>

ЛИТЕРАТУРА

- [1] Evaluating large language models trained on code / M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F.P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W.H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A.N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, W. Zaremba // arXiv preprint arXiv:2107.03374 [cs.LG], 14 Jul. 2021. P. –, doi:10.48550/arXiv.2107.03374, (доступно: <https://arxiv.org/abs/2107.03374>). P. 4.
- [2] Logical segmentation of source code / J. Dormuth, B. Gelman, J. Moore, D. Slater // Proceedings of the 31st International Conference on Software Engineering and Knowledge Engineering (SEKE 2019, Lisbon, 10–12 July 2019). KSI Research Inc. and Knowledge Systems Institute Graduate School, 2019. P. 717–722, doi:10.18293/SEKE2019-026, (доступно: <http://dx.doi.org/10.18293/SEKE2019-026>).
- [3] DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature / E. Mitchell, Y. Lee, A. Khazatsky, C.D. Manning, C. Finn // arXiv preprint arXiv:2301.11305 [cs.CL], 23 Jul. 2023. doi:10.48550/arXiv.2301.11305, (доступно: <https://arxiv.org/abs/2301.11305>).
- [4] Automated vulnerability detection in source code using minimum intermediate representation learning / X. Li, L. Wang, Y. Xin, Y. Yang, Y. Chen // Applied Sciences. 2020. Vol. 10 (Mar. 2020). P. 1692, doi:10.3390/app10051692, (доступно: <https://doi.org/10.3390/app10051692>).
- [5] GPTSniffer: A CodeBERT-based classifier to detect source code written by ChatGPT / P. Nguyen, J. Di Rocco, C. Di Sipio, R. Rubei, D. Di Ruscio, M. Di Penta // Journal of Systems and Software. 2024. Vol. 214 (Apr. 2024). P. 112059, doi:10.1016/j.jss.2024.112059.

Разработка модели классификации последовательностей рецензий исходного кода

Петрова Полина Андреевна

Кафедра системного программирования

e-mail: ppolina@ispras.ru

Научный руководитель — к.ф.-м.н. Турдаков Денис Юрьевич

Научный консультант — Марков Сергей Игоревич

Рецензирование кода (англ. Code Review) — важная часть цикла разработки программы. Это процесс проверки исходного кода другими разработчиками с целью повышения качества программы и выявления ошибок. Благодаря этому процессу в команде появляется достаточно много положительных эффектов для качества кода и продуктивности команды. Но у Code Review есть и отрицательные стороны. Самая главная — большое количество времени тратится на рецензирование. По результатам исследований [1] от 5 до 20 процентов рабочего времени программиста занимает рецензирование, обсуждение и исправление кода. Также увеличивается время принятия кода в репозиторий.

Каждый комментарий, написанный рецензентом, содержит в себе рекомендации по исправлению кода или вопрос. Диалоги — это последовательности таких комментариев между рецензентами и автором кода. После сбора статистики было выявлено, что более 25% диалогов длины больше двух и количество участников в среднем равно двум. Можно выделить две основные группы последовательностей сообщений на Code Review. Первый тип включает в себя два сообщения: рекомендации по исправлению от рецензента и ответ автора с согласием на исправление. Второй тип содержит уже больше двух сообщений — это дискуссия между рецензентами и автором кода. Такое происходит, когда автор не соглашается с рекомендациями коллег и вступает с ними в спор. Дискуссия может длиться долго, это является проблемой для принятия кода в репозиторий.

Данная работа направлена на сокращение времени рецензирования кода. Классификация последовательностей рецензий может быть полезна руководителю команды. Он может отслеживать в реальном времени, о чем переписываются программисты и вовремя вмешаться в процесс, не читая все сообщения и не тратя много своего времени, а просто посмотрев на какие темы (классы) общаются его коллеги. Классификация помогает анализировать статистику комментариев, чтобы в дальнейшем оптимизировать Code Review, например, добавив линтеры, для того чтобы следить за стилем кода. Классификация может выделять признаки для прикладных задач. Например, если последовательность комментариев помечена классом Explanation (ответ на вопрос), то в дальнейшем на таких последовательностях можно обучать модели искусственного интеллекта отвечать на вопросы рецензентов, чтобы уменьшить время, которое тратит программист на ответы на простые

вопросы, а также ускорить процесс принятия кода в репозиторий, так как рецензенты будут быстрее дожидаться ответов и смогут быстрее одобрить программу.

Целью данной работы является реализация классификатора последовательностей рецензий к исходному коду. Классификатор должен работать с приемлемой точностью для решения прикладных задач. Для достижения цели были поставлены подзадачи: повысить качество уже разработанного классификатора отдельных рецензий, основанного на модели CodeBERT [2], выявить классификацию для последовательностей сообщений путем автоматической кластеризации, реализовать модель классификации.

Для повышения качества существующего классификатора, обученного на предыдущем этапе работы, несколькими экспертами нашей команды был пересмотрен набор данных для обучения, состоящий из вручную размеченных данных и четырех открытых датасетов, найденных соответственно в статьях: китайского университета National University of Defense Technology; автора Muntazir Fadhel; авторов из Польши и Швеции; авторов из университета США — Wayne State University, Бангладеша — Bangladesh University of Engineering and Technology. Инструмент основанный на модели CodeBERT из последней статьи [3] показал лучшие результаты на объединенном наборе данных. В результате обработки датасета было очищено около 9% данных: исправлены явные ошибки для тривиальных классов, для близких классов (функциональность, дизайн, рефакторинг) уточнены инструкции для разметки и исправлены ошибки, из комментариев удалены цитирования предыдущих комментариев. Были размечены новые данные, добавилось 10% к набору и его объем увеличился до 11 009 рецензий. Значение f1-масто было увеличено с 0.52 до 0.60 при классификации на 16 классов.

Проведены следующие эксперименты по поиску классификации для последовательностей рецензий: кластеризация с помощью KMeans, алгоритмы поиска наибольшей общей подпоследовательности с помощью суффиксного дерева, Generalized Sequential Pattern и подсчет n-grams. В результате алгоритм основанный на n-grams показал результат не хуже, при этом он работает быстрее других методов на большом наборе данных. Для создания классификации была собрана статистика подпоследовательностей в диалогах, самые популярные занесены в итоговую классификацию. Для нахождения часто встречающихся подпоследовательностей были опробованы разные методы, но выбран подсчет уникальных n-grams с весами. Также список классов обсужден и согласован с командами, которые много лет занимаются Code Review.

В результате исследования построена общая схема работы классификатора отдельных рецензий. Комментарии выгружаются и собираются в последовательности, в которых они были написаны, с помощью атрибутов каждого сообщения. При получении последовательности сообщений, классифицируются отдельные комментарии с помощью классификатора отдельных рецензий, ос-

нованного на CodeBERT, потом из списка классов составляются n-grams и те, которые есть в списке шаблонов классификации последовательностей отвечают за их метку, их может быть несколько.

Для проведения экспериментов был собран датасет из трех популярных проектов с открытым исходным кодом, использующих для разработки сервисы GitHub и Gerrit: LibreOffice, TensorFlow, Matplotlib. Общий размер набора данных составил около 30 тысяч последовательностей. В результате алгоритм покрывает 80% последовательностей, только пятая часть помечается как «unclassified».

В ходе выполнения дипломной работы была успешно решена задача разработки модели классификации последовательностей рецензий исходного кода программ. Были решены подзадачи, обозначенные в начале исследования, собрана статистика и предоставлены данные для других прикладных задач.

Была улучшена модель классификации отдельных рецензий, основанная на CodeBERT. Очищено 9% набора данных, набор расширен до 11 009 комментариев к исходному коду, реализован специализированный препроцессинг. В результате показатель f1-масго вырос на 15% и составил 0.60.

Проведено исследование и собрана статистика по комментариям и их последовательностям. Был опробован набор алгоритмов поиска часто встречающихся подпоследовательностей и выявлена классификация последовательностей рецензий.

Подготовлен набор данных, размером 29 642 последовательности рецензий и разработана модель классификации последовательностей, основанная на методе n-grams. Реализованный классификатор имеет следующие метрики: правильно или допустимо классифицирует 75% последовательностей рецензий в общей сложности, имеет точность 92% на коротких последовательностях.

С помощью разработанной классификации получены данные для прикладной задачи: обучение модели автоматических ответов на вопросы рецензентов.

По теме данной работы было выступление на международной конференции «Интеллектуализация обработки информации, 2024», где исследование вызвало интерес у жюри и участников конференции. По итогам выступления были опубликованы тезисы доклада [4] и принята на публикацию расширенная версия статьи.

Задача классификации отдельных комментариев успешно доработана, а задача классификации последовательностей комментариев решена и получены приемлемые результаты для решения прикладных задач.

ЛИТЕРАТУРА

- [1] Bosu A., Greiler M., Bird C. Characteristics of Useful Code Reviews: An Empirical Study at Microsoft // 2015 IEEE/ACM 12th Working Conference on Mining Software Repositories, Florence, Italy, 2015, P. 146–156, doi: 10.1109/MSR.2015.21.

- [2] CodeBERT: A Pre-Trained Model for Programming and Natural Languages / Z. Feng, D. Guo, D. Tang, N. Duan, X. Feng, M. Gong, L. Shou, B. Qin, T. Liu, D. Jiang, M. Zhou // Conference «Findings of the Association for Computational Linguistics: EMNLP», 2020
- [3] Towards Automated Classification of Code Review Feedback to Support Analytics / A. Turzo, F. Faysal, O. Poddar, J. Sarker, A. Iqbal, A. Bosu // Proceedings of the 17th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), 2023
- [4] Петрова П. А., Марков С. И., Качанов В. В. Создание набора данных для комбинированной классификации рецензий исходного кода // Сборник тезисов международной конференции «Интеллектуализация обработки информации, 2024» С. 83–85, eLIBRARY ID: 80268337

Оценки безрискового эквивалента для различных классов распределений

Работа удостоена диплома 3 степени
Пилушенюк Артур Александрович

Кафедра системного анализа
e-mail: pilyushenok.arthur@gmail.com

Научный руководитель — д.ф.-м.н. проф. Смирнов Сергей Николаевич

Одной из классических задач математических финансов является задача оптимального инвестирования в одношаговой модели рынка [1]. Она состоит в максимизации ожидаемой полезности конечной стоимости портфеля по его структуре ξ при бюджетном ограничении. Однако практическая применимость данной модели существенно ограничена: для корректного расчёта математического ожидания необходимо полное знание распределения вектора цен S , хотя на практике исследователь обладает неполной информацией. Для устранения недостатков этой модели рассматривается робастная формулировка задачи: необходимо оценить

$$\max_{\xi \in \mathbb{R}_+^n} \min_{\pi \in \mathcal{Q}} u^{-1} \left(\mathbb{E}_{\pi} \left[u \left(\xi^T S \right) \right] \right) \quad (1)$$

где оператор $u^{-1}(\mathbb{E}_{\pi}[\cdot])$ называется безрисковым эквивалентом случайной выплаты для функции полезности u и распределения π из допустимого класса $\mathcal{Q}$. Минимизация учитывает наиболее неблагоприятный сценарий поведения рынка, а максимизация заключается в поиске наилучшей стратегии для инвестора. При описании риска безрисковый эквивалент обладает тем преимуществом, что он легче интерпретируется, чем ожидаемая полезность.

Построение интерпретируемых оценок, учитывающих эмпирические характеристики, является важным шагом для усовершенствования методов управления риском и прогнозирования. Современные результаты показывают, что в робастных моделях для безрискового эквивалента с постоянным абсолютным коэффициентом отвращения к риску существуют экспоненциально-конические реформулировки [2]. Численные примеры демонстрируют, что получаемые стратегии превосходят по качеству решения стандартные стохастические подходы, особенно при высокой степени неприятия к риску.

В настоящей работе исследуется одномерная задача минимизации, вспомогательная для (1), при предположении выпуклости класса допустимых распределений $\mathcal{Q}$:

$$CE(\pi) = u^{-1} \left(\int u(x) \pi(dx) \right) \rightarrow \min_{\pi \in \mathcal{Q}},$$

Формализация задачи. Необходимо оценить наихудший случай: найти значение и экстремальное распределение для $CE(\pi)$ среди абсолютно непрерывных распределений $\pi \in \mathcal{Q}$, удовлетворяющих моментным ограничениям

$q = (q_0, q_1, \dots, q_d), q_0 = 1$, т. е.

$$\int x^k \pi(dx) = q_k, \quad k = 0, 1, \dots, d.$$

Класс вероятностных распределений $\mathcal{Q}$ предполагается замкнутым выпуклым конусом и представляется конической оболочкой множества распределений $\mathcal{T} = \{\pi_t \mid t \in I_{\mathcal{T}}\}$, где $I_{\mathcal{T}}$ — конечный интервал. Классы унимодальных с фиксированной модой распределений и симметричных с фиксированным центром распределений являются выпуклыми [3].

Функция полезности $u(x) = \sum_{r=0}^d z_r x^r$ предполагается строго возрастающим многочленом степени не выше d .

Указанные предположения позволяют сформулировать вспомогательную задачу полуопределённого программирования (semidefinite program, SDP), допускающую численное решение.

Двойственная задача. Разрешимость исходной задачи и структура экстремального распределения анализируются через двойственную задачу:

$$y^T q \rightarrow \min_{y \in \mathbb{R}^{d+1}},$$

$$y^T f(\cdot) + u(\cdot) \in \mathcal{Q}^*,$$

где $f(x) = (1, x, x^2, \dots, x^d)$, каждая компонента вектора y соответствует ограничению исходной задачи и множество функций

$$\mathcal{Q}^* = \left\{ h(\cdot) \mid \int h(x) \pi_t(dx) \geq 0, \forall t \in I_{\mathcal{T}} \right\}$$

называется полярной множества $\mathcal{Q}$.

В работе [3] указано, что достаточным условием сильной двойственности является принадлежность вектора q внутренности множества достижимых моментов. Там же выявлено, что если полярный конус $\mathcal{Q}^*$ состоит из многочленов степени не выше d , то двойственная задача эффективно решается как SDP, а экстремальное распределение является комбинацией $d+1$ мер из порождающего класса $\mathcal{T}$. Различные диапазоны цен приводят к различным SDP-реформулировкам [4].

Вопрос о достижимости вектора q на некотором распределении $\pi \in \mathcal{Q}$ восходит к проблеме моментов, постановка которой была впервые поставлена Т. Стилтесом в 1894-1895 гг. для положительной полуоси. В 2005-2007 годах развитие методов внутренней точки, используемых для численного решения нелинейных задач выпуклой оптимизации возобновило интерес к обобщенным чебышевским неравенствам, ранее развитие которых связано с чебышевскими системами [5]. В 2014 году было показано, что многомерные аналоги неравенств Чебышева и Гаусса укладываются в единые теоретические рамки и соответствующие задачи максимизации вероятности образуют параметрическое семейство оптимизационных задач, каждая из которых допускает вычислительно эффективную SDP-реформулировку.

Для унимодальных на $[L_1, L_2]$ распределений с модой m двойственная задача выписывается в виде

$$\begin{aligned} \min_{y \in \mathbb{R}^{d+1}} \quad & y^T q, \quad \text{т.ч.} \\ & 0 \leq \sum_{r=0}^d (y_r + z_r) m^r, \\ & 0 = \text{diag}_{2l-1}(U), \quad l = 1, \dots, d+1. \\ & 0 = \text{diag}_{2l-1}(V), \quad l = 1, \dots, d+1. \\ & \sum_{n=0}^l \sum_{r=n}^{d+n-l+1} C_r^m C_{d-r+1}^{l-n} a_r^{(-)} L_1^{r-n} m^n = \text{diag}_{2l}(U), \quad l = 0, \dots, d+1. \\ & \sum_{n=0}^l \sum_{r=n}^{d+n-l+1} C_r^m C_{d-r+1}^{l-n} a_r^{(+)} m^{r-n} L_2^n = \text{diag}_{2l}(V), \quad l = 0, \dots, d+1. \end{aligned} \quad (2)$$

$$U_{(d+2) \times (d+2)}, V_{(d+2) \times (d+2)} \succeq 0,$$

где $\text{diag}_b(X) = \sum_{i+j=b} x_{ij}$ — антидиагональная сумма порядка b матрицы $X = \{x_{ij}\}_{i,j=0}^n$ и коэффициенты $a_r^{(-)}, a_r^{(+)}, r = \overline{1, d+1}$ равны

$$\begin{aligned} a_0^{(+)} &= \sum_{r=0}^d \frac{-y_r - z_r}{r+1} m^{r+1}, & a_0^{(-)} &= -a_0^{(+)}, \\ a_r^{(+)} &= \frac{1}{r} (y_{r-1} + z_{r-1}), & a_r^{(-)} &= \frac{(-1)^r}{r} (y_{r-1} + z_{r-1}). \end{aligned}$$

Аналогичным образом для симметричных на $[M-L, M+L]$ распределений с центром M двойственная задача выписывается в виде

$$\begin{aligned} \min_{y \in \mathbb{R}^{d+1}} \quad & y^T q, \quad \text{т.ч.} \\ & 0 = \text{diag}_{2l-1}(U), \quad l = 1, \dots, d+1. \\ & \sum_{n=0}^l \sum_{r=n}^{d+n-l} C_r^m C_{d-r}^{l-n} h_r(M-L)^{r-n} (M+L)^n = \text{diag}_{2l}(U), \quad l = 0, \dots, d+1. \\ & U_{(d+1) \times (d+1)} \succeq 0. \end{aligned}$$

Численные решения двойственных задач выполнены на языке Python с использованием интерактивной среды Jupyter Notebook и пакета программного обеспечения MOSEK. Работоспособность программы проверена на синтетических данных. Для улучшения работы метода была добавлена нормализация переменных $\begin{pmatrix} y_0 & y_1 \\ y_1 & y_2 \end{pmatrix} \succeq 0$. Выявлено, что при её отсутствии алгоритм не всегда сходится.

Вектор переменных y выписанных двойственных задач соответствует коэффициентам многочлена, прилегающего сверху к функции $-u(x)$. На рис. 1 показаны геометрическая интерпретация оптимального численного решения задачи (2) и достижимость оценки.

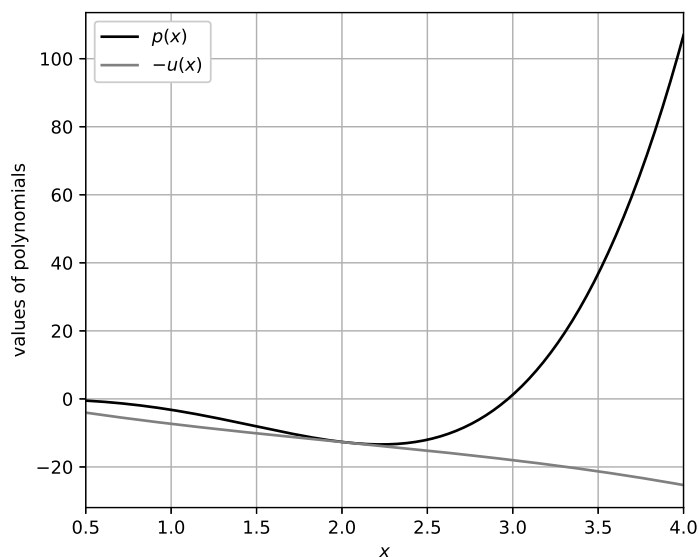


Рис. 1: Результат работы программы для унимодальных распределений.

Дальнейшие исследования сосредоточатся на строгом решении задачи на максимин, сопоставлении с задачей на минимакс, исследовании их равенства. Параллельно предстоит теоретически обосновать выявленные недостатки текущего SDP-алгоритма, а также проверить полученные выводы на реальных рыночных данных.

ЛИТЕРАТУРА

- [1] Föllmer H., Schied A. Stochastic finance. An introduction in discrete time, 4th edition. New York: Walter de Gruyter, 2016. 608 p.
- [2] Chen L., Sim M. Robust CARA optimization // Operations Research, 2025. Vol. 73. N 3. P. 1459-1478.
- [3] Popescu I. A semidefinite programming approach to optimal moment bounds for convex classes of distributions // Mathematics of operations research, 2005. Vol. 30. N 3. P. 632-657.
- [4] Bertsimas D., Popescu I. Optimal inequalities in probability theory: a convex optimization approach // SIAM Journal on Optimization, 2005. Vol. 15. N 3. P. 780-804.
- [5] Карлин С., Стадден В. Чебышевские системы и их применение в анализе и статистике. М. : Наука, 1975. 568 с.

Разработка подхода к обучению сверточных и полносвязных нейронных сетей на основе параллелизма данных

Работа удостоена диплома 3 степени

Пономаренко Алексей Сергеевич

Кафедра суперкомпьютеров и квантовой информатики

e-mail: s02210518@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. доц. Попова Нина Николаевна

Введение. Фреймворки машинного обучения предоставляют исследователям мощные высокоуровневые интерфейсы для проектирования, обучения и валидации нейронных сетей с произвольной архитектурой. Эти инструменты значительно упрощают процесс разработки, абстрагируя пользователя от низкоуровневых деталей реализации и оптимизации вычислительных процессов, таких как управление памятью, распределение вычислений и аппаратное ускорение. Благодаря этому исследователи могут сосредоточиться на создании моделей и экспериментировании с архитектурами без необходимости глубокого погружения в детали реализации.

Однако подобная автоматизация и абстракция зачастую сопряжены с ограничениями в возможностях тонкой настройки распределения вычислительных ресурсов, что приводит к неэффективному использованию аппаратного обеспечения. Особенно это становится заметно в условиях масштабируемости, когда требуется эффективно использовать многопроцессорные и кластерные системы для ускорения обучения моделей. Ограниченная гибкость стандартных инструментов затрудняет реализацию адаптивных стратегий распараллеливания и балансировки нагрузки, что ведет к субоптимальной производительности.

Разработка алгоритма с полной настройкой процесса обучения сверточных и полносвязных нейронных сетей, способного эффективно масштабироваться как на кластерных вычислительных системах, так и на однопроцессорных с поддержкой многопоточности, открывает новые возможности для повышения производительности. Такой подход позволяет не только существенно ускорить обучение на современных многоядерных и распределённых системах, но и сделать однопроцессорные решения конкурентоспособными за счёт эффективного использования параллелизма. В рамках данной работы рассматривается реализация и анализ такого алгоритма с использованием возможностей современного стандарта языка C++23 и технологии OpenMP, что обеспечивает высокую производительность и гибкость в управлении вычислительными ресурсами.

Цель и постановка задачи. Целью работы является исследование эффективности подходов к распараллеливанию обучения сверточных и полносвязных нейронных сетей для многоядерных вычислительных систем.

Постановка задачи:

1. Провести анализ существующих подходов к распараллеливанию алгоритмов обучения нейронных сетей
2. Разработать параллельные алгоритмы обучения сверточных и полносвязных нейронных сетей для многоядерных вычислительных систем
3. Разработать программную реализацию предложенных методов обучения сверточных и полносвязных нейронных сетей на языке C++ с использованием технологий параллельного программирования
4. Исследовать масштабируемость и эффективность предложенных алгоритмов

Предложенные алгоритмы многопоточного обучения сверточных и полносвязных нейронных сетей на основе параллелизма данных.

Синхронный алгоритм начинается с инициализации всех параметров модели — весов и смещений — а также гиперпараметров обучения, включая скорость обучения, количество эпох и стратегию раннего останова. Далее обучающая выборка делится на непересекающиеся батчи фиксированного размера, которые складываются в пул $\mathcal{P}$ — очередь, доступную для конкурентного доступа потоков.

Каждая эпоха состоит из последовательной обработки всех батчей в пуле. На каждом шаге из пула извлекается следующий доступный батч, который параллельно обрабатывается каждым потоком. Обработка включает в себя прямое распространение (вычисление выхода модели), обратное распространение ошибки (вычисление градиентов локально в потоке), а затем синхронную блокировку всех потоков до завершения этого этапа.

После завершения локальных вычислений в каждом потоке происходит централизованное агрегирование градиентов. Главный поток (или выделенный мастер-поток) усредняет градиенты по всем потокам, получая глобальный градиент для текущего шага. Модель обновляется синхронно с использованием этого усреднённого значения.

Цикл повторяется до тех пор, пока пул батчей не будет исчерпан. После каждой эпохи проверяется условие раннего останова: если метрика качества на валидационном наборе не улучшается в течение заданного количества эпох, то обучение прерывается досрочно, и текущие параметры модели сохраняются в файл.

Такой подход обеспечивает высокую степень согласованности параметров модели в ходе обучения, поскольку обновления происходят исключительно на основе усреднённого градиента, что исключает гонки данных и обеспечивает стабильность сходимости. Алгоритм адаптирован под архитектуры с общей памятью (многопоточная реализация на CPU) и в будущем может быть обобщён на распределённые системы с использованием коллективных операций (например, MPI AllReduce).

Асинхронный алгоритм начинается с инициализации всех параметров модели, включая веса и гиперпараметры, такие как скорость обучения, количество эпох и интервал синхронизации. Затем обучающая выборка делится на непересекающиеся батчи фиксированного размера, которые помещаются в пул $\mathcal{P}$ для параллельной обработки.

Каждая эпоха состоит из обработки всех батчей в пуле. На каждом шаге из пула извлекается доступный батч, который обрабатывается каждым потоком. Каждый поток выполняет прямой проход через модель, вычисляет ошибку и локальную функцию потерь, затем выполняет обратное распространение для вычисления локальных градиентов. Веса модели в каждом потоке обновляются локально с использованием собственного градиента.

После обработки всех батчей в текущей эпохе проверяется, если номер эпохи кратен заданному параметру *sync_interval*. В случае совпадения выполняется синхронизация: усредняются веса всех потоков, и полученные значения обновляют глобальные параметры модели.

Цикл повторяется до тех пор, пока пул батчей не будет исчерпан. После каждой эпохи проверяется условие ранней остановки: если метрика качества на валидационном наборе не улучшилась в течение заданного числа эпох, обучение прерывается досрочно, и параметры модели сохраняются.

Этот подход позволяет эффективно использовать параллельные вычисления, обеспечивая быстрые обновления весов при минимальной задержке на синхронизацию. Однако для достижения хороших результатов важно правильно настроить интервал синхронизации, чтобы обеспечить баланс между параллельной производительностью и качеством сходимости модели. Алгоритм может быть адаптирован для распределённых систем с частичной синхронизацией, где обновления происходят локально с периодическими коллективными операциями.

Программная реализация разработанных алгоритмов. Система реализована на языке C++ с соблюдением принципов объектно-ориентированного проектирования. Параллельная обработка данных основана на многопоточности, реализованной через технологию OpenMP.

Для повышения удобства использования, гибкости настройки и обеспечения устойчивости процесса обучения нейронной сети в разработанной системе реализован ряд вспомогательных механизмов:

1. Раннее завершение обучения (Early Stopping)
2. Аугментация и регуляризация входных данных
3. Визуализация архитектуры рассматриваемой нейросетевой модели
4. Конфигурирование модели через файл настроек
5. Контрольные точки обучения (Checkpointing)
6. Автоматическая фиксация лучших весов модели
7. Выбор оптимальной модели по Парето-фронту

Основные классы и их взаимосвязи представлены в виде UML-диаграммы (рис. 1)

Ядром системы выступает класс `Network`. Он инкапсулирует вектор объектов типа `NetworkLayer`, а также предоставляет методы для конфигурации архитектуры (`AddLayer`), запуска обучения (`Train`), инференса (`GetOutput`), сериализации модели (`Save`) и др.

Подсистемы, ответственные за хранение данных (`DataLoader`), вычисление градиентов (`LossFunction`), оптимизацию параметров (`Optimizer`) и преобразования данных, реализованы в виде независимых классов. Их взаимодействие с `Network` организовано через агрегацию: эти компоненты инжектируются в процесс обучения в качестве параметров (например, `Optimizer` и `LossFunction` передаются в метод `Train`)

Для расширения архитектуры предусмотрен абстрактный класс `NetworkLayer`. У него есть виртуальные методы `Forward`, `Backward` и `UpdateWeights`, которые переопределяются в его наследниках.

Унификация работы с многомерными данными достигнута за счет классов `Volume` и `VolumeSize`. Их использование всеми компонентами системы гарантирует согласованность вычислений.

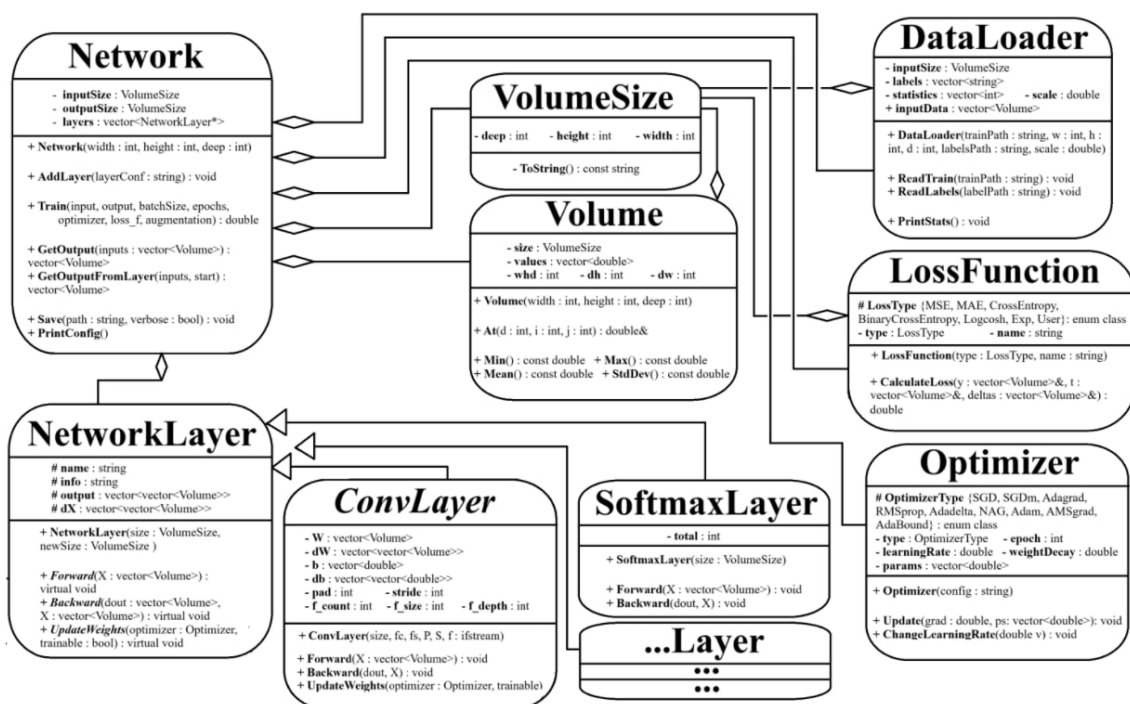


Рис. 1: UML- диаграмма классов системы

В реализации также выполнена векторизация вычислений через `#pragma omp simd`. Также используются директивы `OpenMP #pragma omp atomic` — атомарное сложение локальных значений потерь или счётчиков к общим переменным; `#pragma omp barrier` — барьерная синхронизация, гарантирующая завершение всех потоков перед объединением параметров или переходом к следующей итерации.

Экспериментальное исследование масштабируемости и эффективности разработанных алгоритмов. Экспериментальное исследование проводилось на современных вычислительных платформах кластерных узлов "МГУ-270" и "IBM Polus", обладающих высокой степенью масштабируемости и полноценной поддержкой параллельных вычислений. Эти ресурсы обеспечили необходимую вычислительную производительность и гибкость для всестороннего тестирования разработанных алгоритмов в условиях значительной и разнообразной вычислительной нагрузки.

В качестве прикладной задачи была выбрана задача классификации изображений, основанная на датасете CIFAR-10. Этот датасет представляет собой стандартный корпус из 60,000 цветных изображений размером 32×32 пикселя, равномерно распределённых по 10 классам. Выбор CIFAR-10 обоснован его сбалансированной сложностью: с одной стороны, он содержит достаточно данных для обучения глубоких моделей, с другой — его компактность позволяет эффективно исследовать влияние масштабирования и архитектурных решений. Такое сочетание делает CIFAR-10 особенно подходящей тестовой площадкой для оценки методов оптимизации обучения, включая техники, основанные на параллелизме данных и ускорении вычислений.

Для проведения экспериментов была выбрана архитектура VGG-11, адаптированная под формат входных данных $32 \times 32 \times 3$. Она включает в себя три сверточных блока с функциями активации ReLU и операциями субдискретизации, а также полносвязную выходную голову. Общая численность обучаемых параметров модели составляет приблизительно 3,064,000, что делает её репрезентативной для оценки вычислительной эффективности при сохранении приемлемого времени обучения и стабильной сходимости.

Результаты экспериментальных запусков подтвердили масштабируемость предложенных алгоритмов. Особенно значимый прирост производительности был достигнут в асинхронной версии реализации, которая показала устойчиво более высокую скорость исполнения на обеих платформенных архитектурах. Наибольшее ускорение — 92.74 раза по сравнению с базовой однопоточной реализацией — было достигнуто при запуске на кластере "МГУ-270" при использовании 128 нитей исполнения. Это наблюдение полностью согласуется с теоретическими предпосылками эффективности асинхронного параллелизма при ограниченном взаимодействии между потоками и локализации данных.

Проведено сравнение предложенного алгоритма с реализацией средствами фреймворка PyTorch. Сравнение осуществлялось при заранее подобранных оптимальных гиперпараметрах, обеспечивающих справедливость оценки результатов. Согласно полученным данным, разработанная реализация показала уверенное превосходство над PyTorch по времени обучения, достигнув ускорения на 31% при сохранении сопоставимого уровня точности на тестовом множестве.

Основные результаты работы.

1. На основе анализа существующих подходов к распараллеливанию обучения разработаны параллельные синхронный и асинхронный алгоритмы обучения нейронных сетей с использованием многопоточности
2. Реализованы последовательная и параллельные версии алгоритма на C++ общим объемом около 6000 строк; исходный код размещён в открытом доступе на GitHub (github.com/AlexxPonomarenko/CMC_diploma)
3. Проведено экспериментальное исследование разработанных алгоритмов на платформах вычислительного узла MSU-270 И IBM Polus; показана масштабируемость, получено ускорение до 93 раз по сравнению с последовательной версией алгоритма
4. При оптимальной настройке алгоритм обучается на 31% быстрее по сравнению с реализацией на PyTorch при сопоставимой точности

ЛИТЕРАТУРА

- [1] Goyal P., Dollár P., Girshick R., Noordhuis P., Wesolowski L., Kyrola A., Tulloch A., Jia Y., He K. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour
- [2] Niu F., Recht B., Ré C., Wright S. J. Hogwild!: A Lock-Free Approach to Parallelizing Stochastic Gradient Descent // arXiv preprint arXiv:1106.5730, 2011.
- [3] Zinkevich M., Weimer M., Li L., Smola A. J. Parallelized Stochastic Gradient Descent // Advances in Neural Information Processing Systems (NeurIPS), Vol. 23, 2010. С. 2595–2603.
- [4] Lian X., Huang Y., Li Y., Liu J. Can Decentralized Algorithms Outperform Centralized Algorithms? A Case Study for Decentralized Parallel Stochastic Gradient Descent // Advances in Neural Information Processing Systems (NeurIPS), Vol. 30, 2017
- [5] Frusu A., Arlitt M. Stochastic Gradient Descent on Modern Hardware: Multi-core CPU or GPU? Synchronous or Asynchronous? // Proceedings of the 33rd IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2019

Моделирование динамики веществ в сосудистой системе человека с учетом детоксикационной функции печени

Попко Федор Дмитриевич

Кафедра вычислительных методов

e-mail: popkofd@msu.ru

Научный руководитель — к.ф.-м.н. доц. Хруленко Александр Борисович

В современной медицине и фармакологии, в связи с необходимостью разработки новых лекарственных препаратов, всё более востребованными становятся исследования в области анализа динамики распределения веществ в крови человека. Их распространение существенно зависит от функционального состояния печени и особенностей гемодинамики. По данным Всемирной организации здравоохранения [1], заболевания печени остаются одной из наиболее распространённых причин смертности в мире. Актуальным является изучение различных поражений печени, таких как фиброз и цирроз, которые сопровождаются портальной гипертензией и вызывают серьёзные изменения в гемодинамике.

В данной работе рассматривалась задача моделирования динамики распространения индоцианина зеленого (ИЦЗ) по сердечно-сосудистой системе человека, учитывая детоксикационную функцию печени. Для решения задачи был использован программный комплекс CVSS [2], основанный на решении квазиодномерных уравнений гемодинамики. Он позволяет моделировать кровеносную систему человека [3], численно рассчитывать концентрацию вещества в крови и оценивать периоды его полувыведения. Результаты моделирования сравнивались с экспериментальными данными, полученными в ходе клинических исследований и представленными в работах [4, 5]. Моделирование было проведено для трех сценариев с учетом критериев тяжести поражения печени по шкале Чайлд—Пью (классы А, В и С).

На основании выполненных расчетов были определены численные значения периодов полувыведения ИЦЗ для различных состояний печени: здоровой и с различными степенями тяжести цирроза согласно классификации Чайлд—Пью (классы А, В и С). Для получения корректных расчетов был предварительно рассчитан оптимальный параметр экстракции, характеризующий способность печеночных сегментов к выведению вещества с учетом реальных экспериментальных данных. Полученные численные результаты согласуются с опубликованными клиническими данными, подтверждая адекватность предложенной модели и используемого подхода.

Расчеты показали, что с увеличением тяжести цирроза печени по классификации Чайлд—Пью периоды полувыведения ИЦЗ значительно возрастают, отражая снижение функциональной активности печени. Эти результаты могут быть полезны для дальнейших исследований оценки состояния печени и прогнозирования динамики веществ в организме, что, в свою очередь,

способно помочь врачам в принятии решений по индивидуализации терапии пациентов с заболеваниями печени.

ЛИТЕРАТУРА

- [1] World Health Organization. SDR, chronic liver disease and cirrhosis, all ages per 100 000
- [2] Борзов А.Г., Мухин С.И., Соснин Н.В. Консервативные схемы переноса вещества по системе сосудов, замкнутых через сердце // Дифференц. уравнения, 2012, т.48, №7, с.935–944.
- [3] Жалеев Т.Р., Кубышкин В.А., Мухин С.И., Рубина А.Ф., Хруленко А.Б. Математическое моделирование кровотока в системе сосудов печени // Прикладная математика и информатика: Факультет ВМК МГУ имени М.В. Ломоносова / Под ред. В.И. Дмитриева. — М.: МАКС Пресс, 2019. — №61. — С. 42–56.
- [4] Дзидзава И.И., Котив Б.Н., Кашкин Д.П., Кочаткова А.А., Бугаев С.А., Смородский А.В., Слободяник А.В. Количественная оценка функции печени методом клиренс-теста с индоцианином зеленым // Трансплантология. — 2010. — №1. — С. 29–36.
- [5] Al Mahtab M., Noor E. Alam S.M., Rahim M.A., Alam M.A., Khondaker F.A., Moben A.L., Mohsena M., Akbar S.M.F. Hepatic Venous Pressure Gradient Measurement in Bangladeshi Cirrhotic Patients: A Correlation with Child's Status, Variceal Size, and Bleeding // Euroasian Journal of Hepato-Gastroenterology. — 2017. — Vol. 7, No. 2. — P. 142–145.

Классификация потоков по требованиям к качеству сервиса

Попова Полина Дмитриевна

Кафедра автоматизации систем вычислительных комплексов

e-mail: polinppv@mail.ru

Научный руководитель — к.ф.-м.н. асс. Степанов Евгений Павлович

Актуальность. Некоторые приложения (потокосная передача мультимедиа, голосовая связь, видеосвязь, мониторинг транспортного трафика, умный дом) требуют обеспечения определенного уровня качества сервиса (минимальной пропускной способности). Применение методов балансировки нагрузки позволяет повысить уровень качества сервиса и свести к минимуму частоту перегрузок в работе сети.

При балансировке потоков необходимо учитывать, какой объем данных будет передан на каждом из них. В научных статьях неформально вводятся понятия «поток-слон» и «поток-мышь» [1, 2]. «Потоки-слоны» определяются как потоки, занимающие непропорционально большую долю общей пропускной способности в течение определенного периода времени. Потоки внутри класса «потоков-слонов» могут значительно отличаться по требованиям к пропускной способности. Выделение подклассов внутри данного класса позволяет повысить эффективность балансировки нагрузки, точность приоритизации и уровень качества сервиса [3].

Класс потока необходимо определять по первым пакетам, чтобы успеть применить методы балансировки и приоритизации.

Цель и задачи. Цель выпускной квалификационной работы – разработать метод классификации транспортных потоков по их требованиям к качеству сервиса по первым пакетам.

Для достижения поставленной цели в рамках данной работы необходимо решить следующие задачи:

1. Составление математической постановки задачи классификации потоков по их требованиям к качеству сервиса;
2. Обзор методов классификации транспортных потоков;
3. Разработка и реализация методов классификации транспортных потоков по их требованиям к качеству сервиса на основе выбранных в обзоре методов;
4. Проведение экспериментального исследования для сравнения разработанных методов с точки зрения точности определения класса транспортного потока и эффективности по числу пакетов, необходимых для классификации.

Постановка задачи. Задача классификации транспортных потоков по их требованиям к качеству сервиса формулируется с помощью конечного автомата $M = (A, B, Q, \phi, \psi)$, где:

- A – входной алфавит, представляющий собой значения координат потока $x_i = \{IPs_i, IPr_i, Ps_i, Pr_i, Prt_i, l_{1i}, l_{2i}, \dots, l_{mi}, t_{1i}, t_{2i}, \dots, t_{mi}\}$ из последовательности транспортных потоков $X = \{x_1, x_2, \dots, x_c\}$, где IPs_i, IPr_i – IP-адреса отправителя и получателя, Ps_i, Pr_i – порты отправителя и получателя, Prt_i – протокол транспортного уровня, l_{ji} – длина полезной нагрузки j -го пакета, t_{ji} – время поступления j -го пакета на интерфейс маршрутизатора;
- B – выходной алфавит, представляющий собой множество v классов транспортных потоков $H = \{h_1, h_2, \dots, h_v\}$. Класс h_f задается вектором: $h_f = \{b_{f-1}, b_f\}$, где $b_{f-1}, b_f \geq 0$ ($b_{f-1} < b_f$) – минимальное и максимальное требование по пропускной способности соответственно;
- $Q = \{q_y\}$ – множество состояний автомата M , состоящее из одного элемента q_y – состояния вычисления признаков;
- $\psi : A \times Q \rightarrow Q$ – функция переходов, отображающая входные данные в признаки: $\psi(A) = q_y$;
- $\phi : A \times Q \rightarrow B$ – функция выходов, отображающая признаки в класс потока: $\phi(x_i, q_y) = h_f$.

Необходимо определить функцию ϕ и построить конечный автомат, который по входным данным классифицирует каждый поток x_i в один из классов h_f на основе вычисленных признаков.

Обзор литературы и существующий решений. Целью обзора является выбор метода классификации транспортных потоков. Для сравнения методов, решающих задачу классификации транспортных потоков, предлагается использовать следующие критерии:

1. *Точность работы классификатора*

Данный критерий определяется как доля верно классифицированных потоков из всех транспортных потоков;

2. *Классы потоков*

В научных работах, решающих задачу классификации транспортных потоков, используются разные классы потоков. В настоящей выпускной квалификационной работе необходимо задать классы по требованиям к качеству сервиса (минимальной пропускной способности);

3. *Необходимость данных для обучения*

Для некоторых методов классификации транспортных потоков необходимы сбор данных трассировки пакетов и подготовка датасетов для обучения модели-классификатора, включающая в себя трудоемкий процесс извлечения признаков.

В рамках обзора схожих работ на основании перечисленных выше критериев будут рассмотрены следующие методы, решающие задачу классификации транспортных потоков:

1. Машинное обучение с учителем: метод *Decision Tree* («решающего дерева») [4] и метод *K Nearest Neighbours* («K ближайших соседей»);
2. Машинное обучение без учителя: *Gaussian Mixture Model* («Модель смеси Гаусса») [1] и метод *K-means* («K средних») [2];
3. Ансамблевые методы машинного обучения: метод *Random Forest* («случайного леса») [5];
4. Фрактальный анализ [3].

Исходя из результатов проведенного обзора, можно сделать вывод, что в большей степени удовлетворяющими описанным критериям являются ансамблевые методы машинного обучения в силу высокой точности и фрактальный анализ в силу высокой точности и отсутствия необходимости в обучающих датасетах. Для решения задачи классификации транспортных потоков в данной выпускной квалификационной работе выбран метод машинного обучения «Случайный лес», а также анализ фрактальных размерностей.

Определение «потоков-мышей» и «потоков-слонов». Затруднительно ввести определение, инвариантное относительно физических характеристик канала и типа сетевого трафика. В данной выпускной квалификационной работе в рамках проведения экспериментального исследования использовался канал с пропускной способностью 0.5 Гбит в секунду, исследовался *UDP*-, *TCP*- (незашифрованный) и *TLS*- (зашифрованный) трафик. *Поток данных* назовем «потоком-слоном», если объем передаваемых данных на единицу времени превышает 35 Кбайт в секунду. В противном случае речь идет о «потоке-мышь».

Разбиение «потоков-слонов» на подклассы. Потоки внутри класса «потоков-слонов» могут значительно отличаться по требованиям к пропускной способности, поэтому среди данного класса выделим $v - 1$ подклассов. В настоящей выпускной квалификационной работе на основе собранного трафика исследована зависимость количества транспортных потоков от их коэффициента веса для пропускной способности. Здесь коэффициент веса определяется как отношение объема данных, передаваемых потоком в единицу времени (в Кбайт в секунду), к 35 Кбайт в секунду, то есть к границе между «потоками-мышьями» и «потоками-слонами».

Дискретно найдем площадь под кривой, описывающей зависимость количества транспортных «потоков-слонов» от их коэффициента веса для пропускной способности:

$$S = \sum_i (b'_i - b'_{i-1})g_i, \text{ где}$$

- b'_i ($b'_0 = 0$) – коэффициенты веса для пропускных способностей «потоков-слонов»;
- g_i – количество «потоков-слонов» с коэффициентом b'_i веса для пропускных способностей.

Разделим найденную площадь на $v - 1$ равных частей площадью:

$$s = \frac{\sum_i (b'_i - b'_{i-1})g_i}{v-1}, \text{ где}$$

- b'_i ($b'_0 = 0$) – коэффициенты веса для пропускных способностей «потокослонов»;
- g_i – количество «потокослонов» с коэффициентом b'_i веса для пропускных способностей;
- v – число классов транспортных потоков (включая класс «потокимыши»).

Отсортируем последовательность «потокослонов» по возрастанию их коэффициентов веса для пропускных способностей и будем постепенно добавлять потоки с данными коэффициентами в текущий класс. Как только площадь под кривой, описывающей зависимость количества «потокослонов» текущего класса от их коэффициента веса для пропускной способности, для добавленных потоков превысит s , перейдем к следующему классу.

Выбор признаков для классификации. Определять класс потока можно, основываясь как на индивидуальных пакетных характеристиках, так и на общих характеристиках всего потока.

В качестве общих характеристик потока в научных работах [1, 2], посвященных классификации транспортных потоков, используют количество пакетов, среднюю длину пакета, пропускную способность, длительность транспортного потока, число байт передачи, байтовую и пакетную скорости. К сожалению, упомянутые признаки лишают нас возможности определить класс потока по первым пакетам.

Чтобы успеть применить стратегии балансировки сетевого трафика, необходимо проводить классификацию транспортных потоков по индивидуальным пакетным характеристикам [5]. Среди них такие статистические характеристики потока трафика, как среднеквадратичное отклонение, а также суммарное, среднее, минимальное, максимальное и медианное значения длины полезной нагрузки пакета и межинтервального времени прихода пакетов на интерфейс маршрутизатора, вычисленные для первых m пакетов заданного потока.

Вычисление фрактальных размерностей. Используется следующая формула [3]:

$$d = \frac{\ln N(r)}{\ln \frac{1}{r}}, \text{ где}$$

- d – искомая фрактальная размерность;
- $r = 2^{-n}$ при $n \in \mathbb{N}$ – размер ячейки, которой покрывается фрактал;
- $N(r) = \frac{1}{r}(\max_j(t_j) - \min_j(t_j))$ – количество ячеек размера r , необходимых для полного покрытия фрактала, где t_j – межинтервальное время прихода пакетов.

Экспериментальное исследование. Цели экспериментального исследования:

1. исследование зависимости точности классификации методами на основе «Случайного леса» и на основе анализа фрактальных размерностей от числа первых m пакетов, информация о которых необходима для классификации, для разного числа классов;
2. определение числа m первых пакетов, информации о которых достаточно для классификации транспортного потока с наивысшей возможной точностью.

Данные трассировки *TCP*-трафика предоставлены кафедрой АСВК. Данные трассировки *TLS*-трафика собраны самостоятельно путем просмотра видео при 360p, 480p, 720p, 1080p и захвата пакетов с помощью утилиты *Wireshark*. Данные трассировки *UDP*-трафика собраны самостоятельно путем перебрасывания видео с использованием инструментария *ffmpeg* с помощью *RTP*-протокола с одной виртуальной машины на разные порты другой виртуальной машины. На принимающий виртуальной машине было осуществлено прослушивание входящих пакетов с использованием *tshark*.

В ходе проведения экспериментального исследования определены основные оценки работы классификатора: общая точность *accuracy* – доля верно предсказанных потоков из всех, точность каждого класса *precision* – доля верно предсказанных потоков класса из всех потоков, которые были отнесены к этому классу, полнота каждого класса *recall* – доля верно предсказанных потоков из всех, принадлежащих этому классу, *f1*-мера – гармоническое среднее между полнотой и точностью каждого класса.

С точки зрения точности классификации лучше показал себя разработанный метод на основе «Случайного леса». Полученные в настоящей выпускной квалификационной работы результаты показали, что информации о первых 10 пакетах заданного транспортного потока достаточно для классификации методом на основе «Случайного леса» с точностью не менее 99% и *f1*-мерой не менее 97%.

Заключение. В рамках данной выпускной квалификационной работы получены следующие результаты:

1. Составлена математическая постановка задачи классификации транспортных потоков по их требованиям к качеству сервиса;
2. По результатам проведенного обзора методов классификации транспортных потоков в данной выпускной квалификационной работе выбраны ансамблевый метод «Случайный лес» и анализ фрактальных размерностей;
3. Разработаны и реализованы методы классификации транспортных потоков по их требованиям к качеству сервиса на основе выбранных методов, анализирующие длины пакетов и время между приходами пакетов;
4. Проведено экспериментальное исследование для сравнения разработанных методов с точки зрения точности определения класса потока и эффектив-

ности по числу пакетов, необходимых для классификации, для 2, 3, 4 и 5 классов. С точностью 99% и $f1$ -мерой 97% лучше показал себя метод классификации на основе «Случайного леса», использующий данные трассировки первых 10 пакетов потока.

Для дальнейшего изучения проблемы классификации транспортных потоков представляют интерес следующие направления:

1. классификация транспортных потоков по требованиям к качеству сервиса с использованием методов прогнозирования рисков;
2. классификация транспортных потоков по типам приложений.

ЛИТЕРАТУРА

- [1] Chhabra A., Kiran M. Classifying elephant and mice flows in high-speed scientific networks // Proceedings of INDIS 2017. 2017.
- [2] Khan A., Kelefouras V., Walker D. J., Al-Saadi B., Al-Saadi M. SDN-based routing framework for elephant and mice flows using unsupervised machine learning // Network 3. 2017. P. 218–238.
- [3] Tang P., Dong Y., Jin J., Mao S. Fine-grained classification of internet video traffic from QoS perspective using fractal spectrum // IEEE Transactions on Multimedia. 2020. P. 2579–2596.
- [4] Anantavasilp I. Automatic flow classification using machine learning // 15th International Conference on Software, Telecommunications and Computer Network. 2013. P. 1–6.
- [5] Краснова И. А. Динамическая классификация потоков трафика на основе машинного обучения для обеспечения качества обслуживания в мультисервисной программно-конфигурируемой сети : дис. ... канд. тех. наук : 05.12.13 / Краснова Ирина Артуровна. Москва, 2021. 191 с.

Разработка компилятора операций нейронных сетей для тензорного процессора

Работа удостоена диплома 1 степени

Романов Михаил Максимович

Кафедра системного программирования

e-mail: mmromanov@ispras.ru

Научный руководитель — к. ф.-м. н. доц. Соловьев Михаил Александрович

Научный консультант — к. т. н. ст. преп. Кулагин Иван Иванович

Введение. В настоящее время искусственные нейронные сети внедряются во все сферы человеческой жизни. Для ускорения их исполнения в числе прочего применяются специализированные вычислители. Одним из их видов являются тензорные (нейронные) процессоры [1], ускоряющие выполнение перемножения матриц и векторных операций. Среди особенностей тензорных процессоров можно выделить:

- наличие отдельных АЛУ (арифметико-логических устройств) для сложных операций (например, матричного умножения),
- использование простейших форм параллелизма (напр., SIMD),
- многоуровневую организацию памяти,
- параллельную работу АЛУ.

Для эффективного использования данных особенностей необходим оптимизирующий компилятор нейронных сетей. Примерами таких компиляторов являются XLA, Glow, OpenVINO и другие.

В данной работе разработан компилятор операций нейронных сетей для архитектуры DaVinci, на которой основано семейство процессоров Ascend. Особое внимание уделено эффективной трансляции матричного умножения, так как оно составляет большую часть времени выполнения многих современных нейронных сетей. В частности, используются *блочная стратегия* [2], так как множители могут, вообще говоря, не поместиться во внутреннюю память процессора, и *двойная буферизация* — техника разделения обрабатываемых данных на две части, над каждой из которых параллельно проводятся разные этапы обработки. Выполнен анализ архитектуры DaVinci.

Компилятор основан на фреймворке MLIR. Основное его понятие — диалект, являющийся совокупностью операций, типов и оптимизирующих трансформаций. Он позволяет использовать подход, при котором крупноблочные операции нейронных сетей транслируются в более низкоуровневые представления, близкие к машинному коду целевой архитектуры.

В работе также проведён обзор предметно-ориентированных языков и фреймворков, которые можно использовать для компиляции нейронных сетей. Большинство актуальных решений (Halide, TVM) основаны на идее описания алгоритма в виде гнезда циклов, в телах которых находятся скалярные операции, и последующих оптимизаций данного гнезда. Данный подход тяже-

ло применить к архитектуре DaVinci, поскольку её операции обрабатывают большие массивы данных за раз и обладают сложной семантикой.

Архитектура DaVinci. Рассмотрим функциональную структуру архитектуры DaVinci [3]. Вычислительное ядро на её основе содержит:

- сравнительно медленную общую память GM;
- сравнительно быстрые, но небольшие буферы: UBuf, L1, L0A, L0B, L0C;
- вычислительные устройства Cube Unit, Vector Unit и Scalar Unit;
- устройства MTE1..3, перемещающие данные между буферами.

Все устройства могут работать независимо друг от друга, что даёт параллелизм на уровне инструкций.

Cube Unit выполняет матричное умножение. Множители должны находиться в буферах L0A (левый) и L0B (правый), результат окажется в L0C. При этом инструкция умножения требует, чтобы множители были представлены в особых блочных форматах, изображённых на рис. 1.

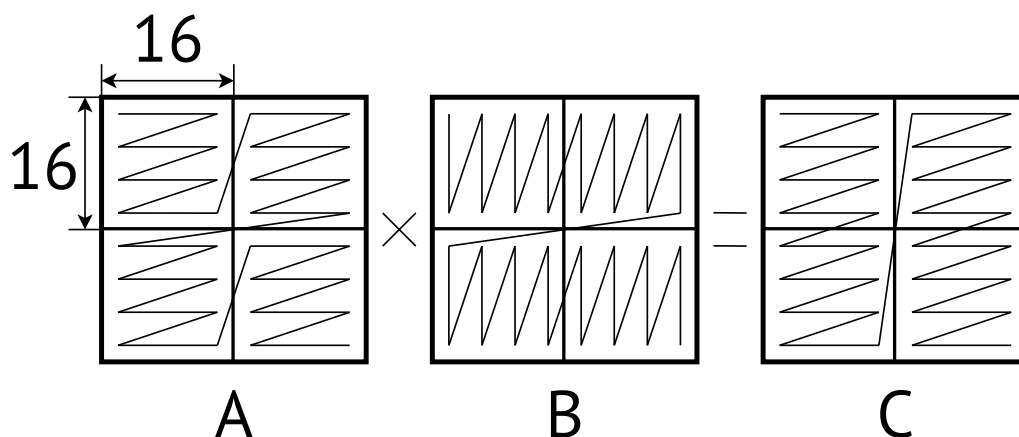


Рис. 1: Формат хранения сомножителей и результата матричного умножения Cube Unit

Формат хранения левого сомножителя (A) назовём Zz , правого (B) — Zn , а формат, в котором записывается результат (C) — Nz . Процесс упаковки построчно хранящихся матриц в такие форматы будем называть *фрактали-зацией*, а обратный, соответственно, *дефрактализацией*. Инструкции, с помощью которых можно осуществить данные преобразования, исполняются устройством Vector Unit над операндами в буфере UBuf.

Описание компилятора. Созданный в данной работе компилятор операций нейронных сетей основан на инфраструктуре MLIR. Заданы 3 диалекта и трансформации, преобразующие код нейронной сети из диалекта StableHLO (взяты из открытых источников) через разработанные диалекты во внутреннее представление LLVM IR. Результат его работы передаётся в закрытый компилятор *ссес*, который генерирует бинарный модуль для архитектуры DaVinci. Помимо этого, разработанный компилятор автоматически генерирует код на языке C, запускающий модель на устройстве.

Среди разработанных диалектов основными являются **ascend** и **cse**. Операции диалекта **ascend** представляют собой высокоуровневые действия, такие как векторные поэлементные операции, фрактализация, матричное умножение и т. д. Диалект **cse** же приближен к ассемблеру архитектуры DaVinci. В трансляции из **ascend** в **cse** сосредоточена основная логика кодогенерации. Все входные данные и результаты операций **ascend** хранятся в общей памяти GM, а операции диалекта **cse** могут находиться в любом буфере.

«Традиционный» подход к матричному умножению состоит в выделении фрактализации и дефрактализации в отдельные операции. Они присутствуют в диалекте **ascend**: **ascend.fractalize** и **ascend.defractalize** соответственно. Фрактализованные матрицы, соответственно, хранятся в общей памяти GM. В данной работе же особое внимание уделено трансляции операции **ascend.matmul2d**, представляющей собой матричное умножение, совмещённое с фрактализацией множителей и дефрактализацией произведения. В ней фрактализации подвергаются не матрицы целиком, а лишь блоки непосредственно перед их перемножением, что исключает накладные расходы на перемещение промежуточного результата в GM и обратно.

Реализованные оптимизации. Проведено экспериментальное исследование эффективности разработанных оптимизаций. В частности, исследована производительность различных вариантов организации фрактализации блоков. Для матриц, ширина или высота которых не кратны 16, во время фрактализации необходимо дополнять недостающие столбцы или строки соответственно нулями. Исследованы способы реализации такого дополнения. Показано, что они существенно различаются по производительности. Разработаны 4 версии фрактализации, отличающиеся друг от друга поэтапным применением оптимизаций к трансляции фрактализации и дополнения нулями по ширине. В таблице 16 показано время выполнения оттранслированного матричного умножения с использованием соответствующих версий фрактализации.

№ версии	Всего	Vector	Cube	Scalar	MTE1	MTE2	MTE3
1	7 369	1 904	186	856	157	908	3 990
2	6 600	1 188	186	822	157	815	3 946
3	6 174	957	186	597	157	720	3 946
4	4 418	616	183	276	147	385	3 029

Таблица 16: время работы функциональных устройств ядра архитектуры DaVinci в микросекундах при выполнении **ascend.matmul2d** с различными оптимизациями фрактализации; матрица A имеет размер 128×128 , B — 128×30522 (взято из BERT-Tiny)

В случаях, когда блок множителя занимает всю его ширину или высоту, возможна оптимизация, состоящая в устранении избыточных загрузок этого блока. В работе исследована её эффективность. В таблице 17 показано время

выполнения оттранслированного матричного умножения без данной оптимизации и с ней.

Оптимизация	Всего	Vector	Cube	Scalar	MTE1	MTE2	MTE3
нет	4 418	616	183	276	147	385	3 029
есть	1 828	797	183	569	89	604	487

Таблица 17: время работы функциональных устройств ядра архитектуры DaVinci в микросекундах при выполнении `ascend.matmul2d` с оптимизацией устранения избыточных загрузок и без неё; матрица A имеет размер 128×128 , B — 128×30522 (взято из BERT-Tiny)

Разные этапы фрактализации и загрузки блоков левого и правого множителя можно подвергнуть двойной буферизации. В случае, когда устранение избыточных загрузок не применено, она состоит в параллелизации фрактализации и загрузок блоков левого и правого множителей. В противном случае один из блоков делится пополам, тогда параллелизуется фрактализация и умножение получившихся половин. В работе исследована эффективность данной оптимизации. В таблице 18 показано время выполнения оттранслированного матричного умножения без двойной буферизации и с ней.

Оптимизация	Всего	Vector	Cube	Scalar	MTE1	MTE2	MTE3
нет	8 281	488	183	200	47	4 080	2 976
есть	1 828	797	183	569	89	604	487

Таблица 18: время работы функциональных устройств ядра архитектуры DaVinci в микросекундах при выполнении `ascend.matmul2d` с двойной буферизацией и без неё; матрица A имеет размер 128×128 , B — 128×30522 (взято из BERT-Tiny)

Было проведено сравнение разработанного алгоритма трансляции операции `ascend.matmul2d` (с фрактализацией блоков «на лету») с «традиционным» матричным умножением (с вынесением фрактализации в отдельную операцию). В таблице 19 показано время выполнения матричного умножения, транслированного «традиционным» и предложенным в данной работе подходом.

Подход	Всего	Vector	Cube	Scalar	MTE1	MTE2	MTE3
«традиционный»	2 273	152	186	157	119	1 005	945
предложенный	1 828	797	183	569	89	604	487

Таблица 19: время работы функциональных устройств ядра архитектуры DaVinci в микросекундах при выполнении матричного умножения «традиционным» подходом и предложенным; матрица A имеет размер 128×128 , B — 128×30522 (взято из BERT-Tiny)

Таким образом, предложенный в работе подход на 30 % эффективнее «традиционного», а реализованные оптимизации оказались эффективными.

Апробация результатов. Доклад о принципах устройства разработанного компилятора был представлен на Открытой конференции ИСП РАН, соответствующая статья опубликована в журнале «Труды ИСП РАН» [4]. Результаты работы докладывались на конференции «Ломоносовские чтения» [5].

ЛИТЕРАТУРА

- [1] Hennessy J. L., Patterson D. A. Computer Architecture: A Quantitative Approach. Cambridge : Elsevier. 2019.
- [2] Goto K., Geijn R. A. van de. Anatomy of high-performance matrix multiplication // ACM Transactions on Mathematical Software. 2008. Vol. 34, N 3. P. 12:1–12:25.
- [3] Xiaoyao Liang. Ascend AI Processor Architecture and Programming. Amsterdam : Elsevier. 2020. P. 18–98.
- [4] Подход к построению компиляторов нейронных сетей с использованием инфраструктуры MLIR / И. И. Кулагин, Р. А. Бучацкий, М. В. Пантилимонов, А. В. Вязовцев, М. М. Романов, Д. М. Мельник // Труды ИСП РАН. 2025. Т. 37, № 1. С. 87–106.
- [5] Кулагин И. И., Романов М. М. Оптимизация генерации кода матричного умножения для тензорного процессора // Ломоносовские чтения. М. : «МАКС Пресс», 2025. С. 158–159.

Спектральный анализ нелокальной краевой задачи для дифференциального оператора с инволюцией и его применения

Работа удостоена диплома 3 степени

Романова Юлия Антоновна

Кафедра общей математики

e-mail: amor.avon@yandex.ru

Научный руководитель — к.ф.-м.н. доц. Крицков Леонид Владимирович

Спектральный анализ дифференциальных операторов, включая неклассические случаи с преобразованиями аргумента, в частности, инволютивными, представляет собой важное направление математической физики.

В данной работе изучены спектральные свойства одного дифференциального оператора с инволюцией – отражением. Найдены спектр и набор собственных функций. Построена сопряженная задача, не совпадающая с исходной. Для системы собственных и присоединенных функций рассматриваемого оператора доказана безусловная базисность в пространстве $L_2[-1, 1]$.

Как приложение спектрального анализа оператора в работе рассмотрена нелокальная начально-краевая задача для уравнения параболического типа. Исследована корректная разрешимость данной задачи и найден класс функций, для которого такая разрешимость имеет место, решение задачи построено в виде ряда по собственным (собственным и присоединенным) функциям дифференциального оператора.

Постановка задачи. Рассматривается спектральная задача:

$$-u''(-x) = \lambda u(x), \quad -1 < x < 1, \quad u(-1) = 0, \quad u(1) = \alpha u(0), \quad (1)$$

где $u(x) \in C^2(-1, 1) \cap C[-1, 1]$, параметр $\alpha \in \mathbb{R}$.

Спектр оператора. Для нахождения спектра задачи (1) построим характеристическое уравнение задачи, используя общее решение дифференциального уравнения в (1): $u(x) = A \cos(\sqrt{\lambda}x) + B \operatorname{sh}(\sqrt{\lambda}x)$ при $\sqrt{\lambda} \neq 0$ и $u(x) = Ax + b$ при $\lambda = 0$ (значение $\sqrt{\lambda}$ выбирается так, чтобы его действительная часть была неотрицательна). Характеристическое уравнение задачи (1) имеет вид:

$$\operatorname{sh}(\sqrt{\lambda})(2 \cos \sqrt{\lambda} - \alpha) = 0.$$

Его решения – собственные значения оператора (1):

$$1. \lambda_n^{(1)} = -\pi^2 n^2;$$

$$2.1. |\alpha| \neq 2:$$

$$\lambda_n^{(2)} = (2\pi n + \arccos(\frac{\alpha}{2}))^2,$$

$$\lambda_n^{(3)} = (2\pi n - \arccos(\frac{\alpha}{2}))^2.$$

При $|\alpha| > 2$ под значением $\arccos(\frac{\alpha}{2})$ понимается аналитическое продолжение вещественного арккосинуса на комплексную плоскость.

2.2. $\alpha = 2$:

$$\lambda_n^{(2)} = 4\pi^2 n^2;$$

$$\lambda_0 = 0;$$

2.3. $\alpha = -2$:

$$\lambda_n^{(2)} = (2\pi n + \pi)^2.$$

Можно заметить, что $\lambda_n^{(i)} \neq \lambda_n^{(j)}, i \neq j$ при $|\alpha| \neq 2$. Однако при $\alpha = \pm 2$ происходит «слипание», так как $\sqrt{\lambda}$ – двукратный корень уравнения $2 \cos \sqrt{\lambda} = \alpha$. Таким образом, каждому собственному значению соответствует одна собственная функция, а при $\alpha = \pm 2$ дополнительно еще одна присоединенная. Для прямой задачи она является решением задачи (1) с неоднородным уравнением $-u''(-x) = \lambda u(x) + u_n(x)$, где $u_n(x)$ – собственная функция.

Сопряженный оператор. Соответствующий этой задаче оператор не является самосопряженным, поэтому для исследования спектральных разложений необходимо сформировать биортогональную пару с собственными функциями сопряженной задачи. Сопряженная задача к задаче (1) находится из равенства $(Lx, y)_{L_2} = (x, L^*y)_{L_2}$ и выглядит следующим образом:

$$\begin{cases} -v''(-x) = \lambda v(x), & x \in (-1, 0) \cup (0, 1), \\ v(1) = v(-1) = 0, & v(0+0) = v(0-0), \\ v'(0+0) - v'(0-0) + \alpha v'(-1) = 0. \end{cases} \quad (2)$$

В силу нелокальности задачи (1) решения сопряженной задачи ищутся в классе разрывных функций $C^2(-1, 0) \cap C^2(0, 1) \cap C[-1, 1]$.

Собственные и присоединенные функции прямой задачи.

При $\alpha \neq \pm 2$:

$$u_n^*(x) = \sin(\pi n x);$$

$$u_n^\pm(x) = \cos((2\pi n \pm \arccos(\frac{\alpha}{2}))x) + \frac{\alpha \operatorname{sh}((2\pi n \pm \arccos(\frac{\alpha}{2}))x)}{2 \operatorname{sh}(2\pi n \pm \arccos(\frac{\alpha}{2}))};$$

При $\alpha = 2$:

$$u_n^*(x) = \sin(\pi n x);$$

$$u_0(x) = x + 1;$$

$$u_n(x) = 2 \cos(2\pi n x) + \frac{2 \operatorname{sh}(2\pi n x)}{\operatorname{sh}(2\pi n)};$$

$$u_n^{(1)}(x) = \frac{1}{4\pi n} \left(2 \operatorname{cth}(2\pi n) \cos(2\pi n x) - 2x \sin(2\pi n x) + \frac{2x}{\operatorname{sh}(2\pi n)} \operatorname{ch}(2\pi n x) \right);$$

При $\alpha = -2$:

$$u_n^*(x) = \sin(\pi n x);$$

$$u_n(x) = 2 \cos((2\pi n + \pi)x) + \frac{2 \operatorname{sh}((2\pi n + \pi)x)}{\operatorname{sh}(2\pi n + \pi)};$$

$$u_n^{(1)}(x) = \frac{1}{2(2\pi n + \pi)} \left(2 \operatorname{cth}(2\pi n + \pi) \cos((2\pi n + \pi)x) - \right. \\ \left. - 2x \sin((2\pi n + \pi)x) - \frac{2x \operatorname{ch}((2\pi n + \pi)x)}{\operatorname{sh}(2\pi n + \pi)} \right).$$

Собственные и присоединенные функции сопряженной задачи.

При $\alpha \neq \pm 2$:

$$v_n^*(x) = \sin(\pi n x) + \frac{(-1)^n}{\left(\frac{2}{\alpha} \operatorname{ch}(\pi n) - 1\right)} \operatorname{sh}(\pi n(1 - |x|));$$

$$v_n^\pm(x) = \cos\left((2\pi n \pm \arccos(\frac{\alpha}{2}))x\right) \mp \frac{\alpha}{\sqrt{4 - \alpha^2}} \sin\left((2\pi n \pm \arccos(\frac{\alpha}{2}))|x|\right);$$

При $\alpha = 2$:

$$v_n^*(x) = \sin(\pi n x) + \frac{(-1)^n}{\left(\frac{2}{\alpha} \operatorname{ch}(\pi n) - 1\right)} \operatorname{sh}(\pi n(1 - |x|));$$

$$v_0(x) = 1 - |x|;$$

$$v_n(x) = -4\pi n \sin(2\pi n|x|);$$

$$v_n^{(1)}(x) = |x| \cos(2\pi n x);$$

При $\alpha = -2$:

$$v_n^*(x) = \sin(\pi n x) + \frac{(-1)^n}{\left(\frac{2}{\alpha} \operatorname{ch}(\pi n) - 1\right)} \operatorname{sh}(\pi n(1 - |x|));$$

$$v_n(x) = -2(2\pi n + \pi) \sin((2\pi n + \pi)|x|);$$

$$v_n^{(1)}(x) = |x| \cos((2\pi n + \pi)x);$$

Для данных наборов функций сформулированы и доказаны следующие утверждения.

Лемма 1. Система собственных и присоединённых функций задачи (1) полна и минимальна в пространстве $L_2[-1, 1]$.

Лемма 2. Пары собственных и присоединённых функции задач (1) и (2) можно нормировать так, чтобы получилась пара биортогонально сопряжённых равномерно минимальных систем функций.

Лемма 3. Система корневых функций задачи (2) удовлетворяет неравенству типа Бесселя, т.е.

$$\exists C > 0 : \quad \forall f \in L_2(-1, 1) \quad \sum_{n=1}^{\infty} \frac{|(f, v_n)|^2}{\|v_n\|^2} \leq C \|f\|_{L_2(-1, 1)}^2.$$

Лемма 4. Система корневых функций задачи (1) удовлетворяет неравенству типа Бесселя и неравенству типа Гильберта, т.е. $\exists B_1, B_2 > 0$:

$$\forall f \in L_2(-1, 1) \quad B_1 \|f\|_{L_2(-1, 1)}^2 \leq \sum_{n=1}^{\infty} \frac{|(f, u_n)|^2}{\|u_n\|^2} \leq B_2 \|f\|_{L_2(-1, 1)}^2.$$

Из доказанных лемм и теоремы Н.К. Бари [1] получается следующий результат.

Теорема 1. Системы корневых (собственных и присоединенных) функций задач (1) и (2) образуют биортогональную пару и являются безусловными базисами в пространстве $L_2[-1, 1]$ для всех вещественных значений параметра α .

Применение к начально-краевой задаче. Рассматривается начально-краевая задача для уравнения параболического типа с инволюцией:

$$\begin{cases} \frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(-x,t)}{\partial x^2}, & -1 < x < 1, \quad t > 0, \\ u(x, 0) = \varphi(x), & -1 \leq x \leq 1, \\ u(-1, t) = 0, \quad u(1, t) = \alpha u(0, t), & t \geq 0. \end{cases} \quad (3)$$

Решение ищется в классическом смысле, т.е.

$$u(x, t) \in C^{2,1}((-1, 1) \times (0, +\infty)) \cap C([-1, 1] \times [0, +\infty)).$$

Теорема 2. Задача (3) корректно разрешима для любого α на множестве четных начальных функций $\varphi \in W_2^1(-1, 1) : \varphi(-1) = \varphi(1) = 0$. Это значит, что:

1. решение существует $\forall \varphi \in W_2^1(-1, 1) : \varphi(x) \equiv \varphi(-x)$,
 $\varphi(-1) = \varphi(0) = \varphi(1) = 0$;
2. решение единственно;
3. выполняются априорные оценки:

$$\begin{aligned} \max_{t \geq 0} \|u(x, t)\|_{L_2[-1, 1]} &\leq C \|\varphi(x)\|_{L_2}, \\ \max_{t \geq 0, x \in [-1, 1]} |u(x, t)| &\leq C \|\varphi(x)\|_{W_2^1}. \end{aligned}$$

Решение задачи (3) для четных $\varphi(x)$ построено в виде биортогонального ряда:

$\alpha \neq \pm 2$:

$$u(x, t) = \sum_{n=1}^{\infty} (\varphi, v_n^+) u_n^+(x) e^{-\lambda_n^+ t} + \sum_{n=1}^{\infty} (\varphi, v_n^-) u_n^-(x) e^{-\lambda_n^- t}$$

$\alpha = -2$:

$$u(x, t) = \sum_{n=1}^{\infty} \left((\varphi, v_n^{(1)}) u_n(x) + (\varphi, v_n) (u_n^{(1)}(x) + t u_n(x)) \right) e^{-\lambda_n t}$$

$\alpha = 2$:

$$\begin{aligned} u(x, t) = \sum_{n=1}^{\infty} \left((\varphi, v_n^{(1)}) u_n(x) + (\varphi, v_n) (u_n^{(1)}(x) + t u_n(x)) \right) e^{-\lambda_n t} + \\ + (\varphi, v_0) u_0(x). \end{aligned}$$

ЛИТЕРАТУРА

- [1] Бари Н. К. Биортогональные системы и базисы в гильбертовом пространстве // Математика. Т. 4: Уч. записки Моск. гос. ун-та. Вып. 148 М., 1951 С. 69–107. No 3 (153). С. 113–174.

Алгоритм группового управления в неопределенной среде с динамическими препятствиями

Работа удостоена диплома 1 степени

Саврасов Геннадий Александрович

Кафедра нелинейных динамических систем и процессов управления

e-mail: gengensav@gmail.com

Научный руководитель — к.ф.-м.н. проф. Ильин Александр Владимирович

В последние десятилетия коллективные системы автономных агентов (многоагентные робототехнические комплексы, беспилотные летательные аппараты, наземные транспортные средства) находят применение в различных прикладных областях: от мониторинга критически важных объектов и разведки территорий до поисково-спасательных операций и сельскохозяйственной автоматизации. Эффективная координация группы агентов при ограниченной и неточной информации о среде остаётся нетривиальной задачей, особенно в условиях динамических препятствий.

Основные трудности управляемости многоагентных систем заключаются в:

- Высокой степени неопределённости окружающей среды
- Наличии динамических препятствий (движущиеся объекты, внезапно появляющиеся завалы)
- Погрешностях сенсорных измерений и задержках передачи данных

Существующие подходы (RRT, методы на основе глубокого обучения, графовые алгоритмы) либо слишком вычислительно затратны для реального времени, либо не гарантируют безопасность при сильных возмущениях среды.

Цель работы: разработка алгоритма группового управления на основе метода потенциальных полей [1], обеспечивающего:

1. Безопасное движение в среде с динамическими препятствиями
2. Сохранение групповой формации
3. Преодоление локальных минимумов
4. Минимизацию времени выполнения задачи

Постановка задачи: Рассматривается ограниченная плоская область $\Omega \subset \mathbb{R}^2$ с множеством агентов $\mathcal{V} = \{1, 2, \dots, n\}$. Каждый агент описывается уравнениями второго порядка:

$$\begin{cases} \dot{\mathbf{q}}_i(t) = \mathbf{p}_i(t) \\ \dot{\mathbf{p}}_i(t) = \mathbf{u}_i(t) \end{cases}$$

где $\mathbf{q}_i \in \mathbb{R}^2$ - позиция i -го агента, $\mathbf{p}_i \in \mathbb{R}^2$ - скорость i -го, $\mathbf{u}_i \in \mathbb{R}^2$ - управляющее воздействие.

Ограничения:

1. *Достижение цели:* $\mathbf{q}_i(t^*) \in G$, $\|\mathbf{p}_i(t^*)\| \leq p_{\text{stop}}$, $\forall i \in \mathcal{V}(t^*)$.
2. *Безопасность движения:*

$$\|\mathbf{q}_i(t) - \mathbf{q}_j(t)\| > 2d^a, \quad \text{dist}(\mathbf{q}_i(t), O_k(t)) > d^a$$

где $G \subset \Omega$ — целевая область, $t^* < \infty$ — время завершения задачи, p_{stop} — скорость остановки, $\mathcal{V}(t)$ — множество нестолкнувшихся агентов до момента времени t , d^a — радиус агента.

Критерии качества:

- Минимизация времени t^* .
- Минимизация интеграла отклонений от формации:

$$C = \int_0^{t^*} \sum_{i=1}^n \sum_{j \in \mathcal{N}_i(t)} (\|\mathbf{q}_i(t) - \mathbf{q}_j(t)\| - d)^2 dt$$

где d — заданное расстояние для строя.

Дискретная модель агента При предположении, что управляющее воздействие на каждом отрезке $[kT, (k+1)T)$, где T — шаг дискретизации, остается постоянным: $\mathbf{u}(t) = \mathbf{u}(kT)$, $t \in [kT, (k+1)T)$. Дискретная модель агента представима в следующем виде:

$$\begin{cases} \mathbf{q}((k+1)T) = \mathbf{q}(kT) + T \mathbf{p}(kT) + \frac{T^2}{2} \mathbf{u}(kT) \\ \mathbf{p}((k+1)T) = \mathbf{p}(kT) + T \mathbf{u}(kT) \end{cases}$$

Формирование строя Множество соседей агента i :

$$\mathcal{N}_i(t) = \{j \neq i : \|\mathbf{q}_i(t) - \mathbf{q}_j(t)\| \leq R_c, j \in \mathcal{V}(t)\}$$

где R_c — радиус сенсорного взаимодействия. Конфигурация является α -решёткой [6] масштаба d , если:

$$\|\mathbf{q}_i(t) - \mathbf{q}_j(t)\| = d \quad \forall j \in \mathcal{N}_i(t)$$

Модель препятствий Рассмотрены три типа препятствий:

1. **Тип А:** Статические круги фиксированного радиуса
2. **Тип В:** Динамические круги с движением центра:

$$\dot{\mathbf{o}}_j(t) = \mathbf{v}_j(t), \quad \dot{\mathbf{v}}_j(t) = \mathbf{w}_j(t), \quad \mathbf{w}_j(t) \sim \mathcal{N}(\mathbf{0}, \sigma_p^2 I)$$

3. **Тип С:** Дуговые структуры из кругов:

$$\begin{aligned} \dot{\mathbf{o}}_j(t) &= \mathbf{v}_j(t), & \dot{\mathbf{v}}_j(t) &= \mathbf{w}_{1j}(t), \\ \dot{\theta}_j(t) &= \omega_j(t), & \dot{\omega}_j(t) &= w_{2j}(t) \end{aligned}$$

где $\mathbf{w}_{1j} \sim \mathcal{N}(\mathbf{0}, \sigma_p^2 I)$, $w_{2j} \sim \mathcal{N}(0, \sigma_\omega^2)$.

Обработка препятствий Каждый агент i обнаруживает препятствия в пределах сенсорного радиуса R_s :

1. Испускает M лучей для определения точек пересечения
2. Группирует точки в кластеры $N_{i,\ell}^\beta$
3. Если агент находится внутри выпуклой оболочки кластера, то строится виртуальное препятствие $\mathcal{H}(N_{i,\ell}^\beta)$, представляющее эту выпуклую оболочку. В противном случае виртуальным препятствием $\mathcal{H}(N_{i,\ell}^\beta)$ является сам кластер.
4. Определяет ближайшие точки к виртуальным препятствиям $\hat{\mathbf{q}}_{i,\ell}$ и проекции своей скорости на виртуальные препятствия $\hat{\mathbf{p}}_{i,\ell}$

Алгоритм управления: Управление $\mathbf{u}_i$ формируется как сумма шести компонент:

$$\mathbf{u}_i = \mathbf{u}_i^\alpha + \mathbf{u}_i^\beta + \mathbf{u}_i^\gamma + \mathbf{u}_i^\delta + \mathbf{u}_i^{gyro} + \mathbf{u}_i^{imp} \quad (1)$$

Компоненты управления

- $\mathbf{u}_i^\alpha$ - поддержание формации (α -решётки) [2]

$$\mathbf{u}_i^\alpha = c_1^\alpha \sum_{j \in N_i} \phi_\alpha(\|\mathbf{q}_j - \mathbf{q}_i\|_\sigma) \mathbf{n}_{ij} + c_2^\alpha \sum_{j \in N_i} a_{ij}(\mathbf{p}_j - \mathbf{p}_i)$$

- $\mathbf{u}_i^\beta$ - избегание препятствий [2]

$$\mathbf{u}_i^\beta = c_1^\beta \sum_{k \in N_i^\beta} \phi_\beta(\|\hat{\mathbf{q}}_{ik} - \mathbf{q}_i\|_\sigma) \hat{\mathbf{n}}_{ik} + c_2^\beta \sum_{k \in N_i^\beta} b_{ik}(\hat{\mathbf{p}}_{ik} - \mathbf{p}_i)$$

- $\mathbf{u}_i^\gamma$ - движение к цели [2]

$$\mathbf{u}_i^\gamma = -c_1^\gamma(\mathbf{q}_i - \mathbf{q}_r) - c_2^\gamma \mathbf{p}_i$$

- $\mathbf{u}_i^\delta$ - межагентное отталкивание

$$\mathbf{u}_i^\delta = c_1^\delta \sum_{j \in N_i} \phi_\delta(\|\mathbf{q}_j - \mathbf{q}_i\|_\sigma) \mathbf{n}_{ij} + c_2^\delta \sum_{j \in N_i} c_{ij}(\mathbf{p}_j - \mathbf{p}_i)$$

- $\mathbf{u}_i^{imp}$ - импульсный компонент (преодоление локальных минимумов)

$$\mathbf{u}_i^{imp} = \begin{cases} c_1^{imp}(\mathbf{u}_i^\beta \times \mathbf{z}), & \|\mathbf{p}_i\| < p^{imp} \\ c_2^{imp} \mathbf{u}_i^{imp}, & \text{иначе} \end{cases}$$

- $\mathbf{u}_i^{gyro}$ - гироскопический компонент (преодоление локальных минимумов) [3]

$$\mathbf{u}_i^{gyro} = \begin{bmatrix} 0 & -\omega(\mathbf{q}_i, \mathbf{p}_i) \\ \omega(\mathbf{q}_i, \mathbf{p}_i) & 0 \end{bmatrix} \mathbf{p}_i$$

Оптимизация параметров Для настройки 18 параметров управления использован метод дифференциальной эволюции [4]. Целевая функция:

$$J = \frac{1}{m} \sum_{i=1}^m (k_1 L_i + k_2 R_i + k_3 C_i + k_4 D_i)$$

где:

- m - количество различных карт для оптимизации параметров
- k_j - веса, определяющие значимость каждой метрики
- $L_i = K_i T$ - время выполнения задачи
- $R_i = \frac{1}{nK_i} \sum_{k=1}^{K_i} \sum_{j \in \mathcal{V}(kT)} \|\mathbf{q}_j(kT) - \mathbf{q}_r\|$ - среднее расстояние до цели
- $C_i = \frac{1}{nK_i} \sum_{k=1}^{K_i} \sum_j \sum_\ell ||\|\mathbf{q}_j - \mathbf{q}_\ell\| - d|$ - отклонение от формации
- $D_i = |\mathcal{V}(K_i)|/n$ - доля не столкнувшихся агентов

Результаты моделирования Для подбора параметров с весовыми коэффициентами, приведенными в Табл. 20, использовались четыре стандартных карты для первичного подбора параметров, а также две карты для анализа поведения в локальных минимумах. Карты для анализа поведения в локальных минимумах выбраны так, чтобы группа агентов застревала в этих областях при оптимальных параметрах при отсутствии сил $\mathbf{u}_i^{gyro}$, $\mathbf{u}_i^{imp}$. Были получены первичные оптимальные параметры управления после 50 поколений. В Табл. 21 приведены метрики работы алгоритма для пяти случайных карт при движении группы из 5 агентов. После первичной оптимизации параметров после 10 поколений были получены параметры для $\mathbf{u}_i^{gyro}$, $\mathbf{u}_i^{imp}$.

k_1	k_2	k_3	k_4
100	20	400	400

Таблица 20: Параметры оптимизации

Карта	Тип препятствий	L_i	R_i	D_i	C_i
1	A	23.62	111.01	0	0.98
2	B	11.36	103.18	0	1.1
3	B	16.02	109.61	0	1.25
4	C	15.9	83.44	0	1.27
5	C	16.98	99.31	0	1.12

Таблица 21: Результаты моделирования для случайных карт

Заключение Разработан композитный алгоритм группового управления, обеспечивающий:

1. Приемлимое время выполнения задачи
2. Безопасное движение в среде с динамическими препятствиями

3. Сохранение групповой формации (α -решётки)
4. Преодоление локальных минимумов за счёт импульсного и гироскопического управления

Перспективные направления:

- Учёт коммуникационных задержек и потерь
- Адаптация к сенсорным шумам
- Реализация на реальных робототехнических платформах
- Расширение для 3D-сред

ЛИТЕРАТУРА

- [1] Khatib O. Real-time obstacle avoidance for manipulators and mobile robots // Proceedings. 1985 IEEE International Conference on Robotics and Automation. 1985. Vol. 2, P. 500–505.
- [2] Olfati-Saber R. Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory // IEEE Transactions on Automatic Control. 2006. Vol. 51, № 3 P. 401–420.
- [3] Chang D. E., Marsden J. E. Gyroscopic Forces and Collision Avoidance with Convex Obstacles // New Trends in Nonlinear Dynamics and Control and their Applications. 2003. P. 145–159.
- [4] Storn R., Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces // Journal of Global Optimization. 1997. Vol. 11, № 4. P. 341–359.

Планирование потока задач с зависимостями по данным между несколькими вычислительными кластерами

Санников Тимофей Владимирович

Кафедра автоматизации систем вычислительных комплексов

e-mail: timohaj1@yandex.ru

Научный руководитель — к.ф.-м.н. в.н.с. Санников Алексей Николаевич

Введение. Работа посвящена проблеме эффективного планирования вычислительных задач с зависимостями между несколькими кластерами. Современные научные задачи требуют значительных ресурсов, для обработки которых используются кластеры. Доступ к кластерам предоставляется преимущественно через SSH. Цель работы — создание инструмента, автоматизирующего планирование задач с минимизацией времени выполнения потока.

Мат. постановка задачи планирования.

Цель: минимизировать общее время выполнения потока задач:

$$\min_{j \in \mathcal{J}} (T_{\text{total}}^j)$$

$$T_{\text{total}}^j = \max_{i \in \mathcal{T}} (t_{\text{all}}^i),$$

где для каждой задачи i :

$$t_{\text{all}}^i = t_{\text{start}} + t_{\text{tr}}^i + t_{\text{que}}^i + t_{\text{exec}}^i,$$

$$t_{\text{tr}}^i = \frac{S_i}{v} + t_{\text{tr_prev}}^i.$$

- Порядок выполнения, определяемый зависимостями: $j \prec i \Rightarrow t_{\text{all}}^j \leq t_{\text{start}}^i$.
- Запрет на превышение ресурсов кластера: $\sum_{i \in C_k} r_i \leq R_k, \forall k$.
- Запрет гранулярности выполнения задачи: $x_{ik} \in \{0, 1\}$.

Обозначение	Описание
$\mathcal{J}$	Множество расписаний
$\mathcal{T}$	Множество всех задач
T_{total}^j	Общее время выполнения потока при расписании j
t_{all}^i	Полное время выполнения задачи i
t_{tr}^i	Время передачи данных для задачи i
t_{exec}^i	Время выполнения задачи i на кластере
t_{que}^i	Время ожидания задачи i в очереди
S_i	Объем данных задачи i (бит)
v	Скорость передачи данных (бит/с)
$t_{\text{tr_prev}}^i$	Время окончания передачи данных предыдущих задач на кластер
$j \prec i$	Задача i зависит от задачи j
R_k	доступные ресурсы кластера k
r_i	ресурсы, требуемые задачей i
x_{ik}	Задача i выполняется на вычислительном кластере k

Выбор алгоритмов для модификации Для достижения поставленной цели были выделены требования к алгоритмам. Federated и Bucket не учитывают время передачи данных. Spear и GRAPHENE, игнорируют очередь на кластере, а мод. Backfill не работает с зависимостями. Поэтому были выбраны и доработаны те алгоритмы, которые либо имели потенциал для модификации (Spear, GRAPHENE), либо уже частично решали поставленные задачи (Backfill).

Доработка алгоритмов Backfill. Базовый алгоритм Backfill был модифицирован для работы с задачами, имеющими зависимости и требующими передачи данных между кластерами. Основная доработка заключается во введении минимального времени начала исполнения задачи, которое определяется как сумма времени передачи файлов и времени ожидания в очереди. Также была реализована логика выбора кластера на основе минимального ожидаемого времени старта, что позволяет более эффективно использовать ресурсы.

Принцип работы алгоритма основан на поиске «окон» в расписании, куда можно вставить задачу без нарушения текущего порядка задач. Алгоритм проходит по интервалам свободных ресурсов кластера, проверяя, может ли задача быть размещена с учётом всех ограничений.

Дополнительной модификацией стало изменение расчета минимального времени старта – время передачи файлов теперь оканчивалось только после передачи файлов, от которых зависит текущая задача.

GRAPHENE. Алгоритм GRAPHENE был доработан за счёт добавления оценки времени передачи данных. Вес задачи при этом рассчитывается с учётом вычислительной нагрузки, объёма передаваемых данных и количества зависимостей, что позволяет лучше распределять задачи между кластерами.

В основе алгоритма лежит идея виртуального ресурсно-временного пространства, в которое задачи размещаются с учётом зависимостей и доступных ресурсов. Большие задачи размещаются первыми, чтобы минимизировать фрагментацию, а малые задачи заполняют оставшиеся промежутки, обеспечивая более равномерную загрузку кластеров. Выбор кластера осуществляется по коэффициенту загрузки.

$$w_i = \begin{cases} \alpha \cdot (t_{\text{exec}_i} - \frac{S_i}{v}) + \beta \cdot R_i, & \text{большая} \\ \gamma \cdot (\frac{S_i}{v} + \delta \cdot D_i), & \text{малая} \end{cases}$$

Критерии разделения:

— Большая задача:

$$\alpha \cdot t_{\text{exec}_i} + \beta \cdot R_i > T_{\text{threshold}}$$

— Малая задача: иначе

Переменная	Описание
w_i	Вес задачи i
t_{exec_i}	Время выполнения (с)
S_i	Объём данных (бит)
v	Скорость передачи (бит/с)
D_i	Число зависимостей
R_i	Ресурсы (ядра/память)

Spear. В Spear был добавлен учет времени передачи данных и времени в очереди. Модель глубокого обучения с подкреплением (DRL) была доработана: ей подаётся вектор времён передачи данных по каждому кластеру, текущее состояние очередей кластеров. Это сделало выбор кластера более чувствительным к реальной ситуации в системе.

Алгоритм сочетает в себе Monte Carlo Tree Search (MCTS) и DRL. MCTS исследует возможные варианты расстановки задач, а DRL направляет поиск, сокращая количество бессмысленных ветвей. Каждое действие оценивается на основе текущего состояния кластеров, загруженности, зависимостей и прогнозируемого времени выполнения. Такой подход позволяет динамически адаптироваться к изменениям в системе и минимизировать общее время выполнения потока задач.

Программная реализация Для реализации предложенных алгоритмов был разработан программный комплекс, состоящий из сервера управления, модулей планирования и компонентов для взаимодействия с вычислительными кластерами. Основным языком реализации — Python, который использовался для работы с базой данных, отправки файлов и управления процессами через протокол SSH. Планировщики были реализованы на C++ для повышения производительности, и вызывались из Python через библиотеку `ctypes`.

Хранение всей информации о задачах, их статусах и кластерах производилось в базе данных PostgreSQL. Каждая задача имела атрибуты, такие как необходимые ресурсы, максимальное время выполнения, список зависимостей и путь к входным файлам. После поступления задачи на сервер, запуск происходил в два этапа: сначала определялся кластер с минимальным временем начала выполнения, затем происходила передача необходимых файлов и постановка задачи в очередь на соответствующем кластере.

В модифицированном Backfill реализован механизм учёта передачи данных и времени в очереди, а также стратегия выбора кластера с наименьшим стартовым временем. В GRAPHENE реализован расчёт веса задач и динамический анализ загрузки кластеров для выбора наиболее подходящего. В Spear встроены механизмы обучения с подкреплением (DRL) и поиска по дереву (MCTS), адаптированные для работы в распределённой среде, где каждый кластер описывается через своё состояние и очередь задач.

Тестирование Для оценки эффективности реализованных алгоритмов был разработан генератор тестовых данных, создающий потоки задач с различ-

ными параметрами: количеством задач, зависимостями, объёмами передаваемых данных и требованиями к ресурсам. Тесты проводились в идентичных условиях на идентичных потоках данных с изменением алгоритма. Тестирование проходило на вычислительном кластере ЦКП «Информатика» ФИЦ ИУ РАН.

Основным критерием оценки эффективности алгоритмов выступало общее время выполнения потока задач с учётом всех зависимостей, времени передачи данных и ожидания в очереди. Тестирование проводилось на одинаковых потоках задач в контролируемой среде, где варьировалось количество задач и их параметры.

Результаты показали, что при большом числе задач в потоке (от 20) наилучшее время выполнения демонстрировал алгоритм **Spear**. Его способность учитывать сложные зависимости и текущее состояние кластеров через методы обучения с подкреплением позволила достигать высокой степени оптимизации. Однако при меньших потоках лучшим оказался **GRAPHENE**, благодаря своей низкой сложности и быстрой адаптации к изменению загрузки кластеров.

Модифицированный **Backfill** показал стабильные, сбалансированные результаты на всех наборах задач. Он уступал **Spear** на крупных потоках и **Graphene** на малых, но при этом не требовал значительных вычислительных затрат на построение расписания. Таким образом, выбор планировщика следует осуществлять с учётом размера потока задач: для малых потоков целесообразно использовать **Graphene**, для крупных — **Spear**, а **Backfill** может служить универсальным решением в смешанных или ограниченных по ресурсам сценариях.

Результаты работы В ходе выполнения работы была решена задача планирования потока задач с зависимостями по данным между несколькими вычислительными кластерами. Проведён обзор существующих алгоритмов планирования, из которых на основе критериев (учёт зависимостей, времени передачи, очереди и целостности задач) были выбраны и доработаны три: **Backfill**, **GRAPHENE** и **Spear**. Для каждого из них были реализованы модификации, учитывающие специфические особенности распределённой вычислительной среды, включая ограниченный доступ к кластерам и необходимость предварительной передачи данных.

Разработан программный комплекс, включающий сервер управления задачами, реализацию планировщиков на C++ и взаимодействие с вычислительными узлами по SSH. Все алгоритмы были протестированы в одинаковых условиях на сгенерированных потоках задач с различными характеристиками.

Полученные результаты показали, что каждый из алгоритмов имеет свою область эффективности. **Spear** оказался наиболее подходящим для больших объёмов задач, **GRAPHENE** — для небольших потоков с быстрым принятием решений, а **Backfill** показал стабильную работу на всех конфигурациях. Та-

ким образом, разработанная система позволяет адаптивно подбирать планировщик под характер вычислительной нагрузки и эффективно использовать ресурсы кластера.

ЛИТЕРАТУРА

- [1] Hu Z., Tu J., Li B. Spear: Optimized Dependency-Aware Task Scheduling with Deep Reinforcement Learning // Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS). Dallas, TX, USA: IEEE, 2019. P. 201–211.
- [2] Grandl R., Kandula S., Rao S. и др. GRAPHENE: Packing and Dependency-Aware Scheduling for Data-Parallel Clusters // 13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16). USENIX Association, 2016. С. 81–97.
- [3] Mu'alem A. W., Feitelson D. G. Utilization, predictability, workloads and user runtime estimates in scheduling the IBM SP2 with backfilling // IEEE Transactions on Parallel and Distributed Systems. 2001. Vol. 12, № 6. P. 529–543.
- [4] Санников Т. В., Сальников А. Н. Двухуровневое планирование потока задач между несколькими вычислительными кластерами // Программные системы и инструменты. 2023. № 2. С. 60–73.

Математическое моделирование влияния слабой четвёртой гармоники на процесс утроения частоты лазерного излучения

Работа удостоена диплома 3 степени

Сапрончев Андрей Михайлович

Кафедра математической физики

e-mail: s02210537@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. асс. Харитонов Дмитрий Михайлович

Постановка задачи. В выпускной квалификационной работе исследовался процесс утроения частоты лазерного излучения в среде с квадратичным нелинейным откликом с учётом влияния слабой четвёртой гармоники. Исследования процессов утроения частоты в нелинейных средах активно развиваются благодаря их значимости для оптических технологий. Существующие работы [1, 2, 3] подчёркивают роль фазового синхронизма, но часто в них отсутствует анализ высших гармоник. Настоящая работа восполнила этот пробел, впервые было рассмотрено влияние четвёртой гармоники на процесс утроения частоты.

Основной целью работы был анализ влияния четвёртой гармоники на эффективность генерации третьей гармоники, а также сравнение результатов моделирования для систем с учётом и без учёта четвёртой гармоники. Исследование проводилось для различных параметров, включая интенсивность основного излучения и угол падения, в условиях, близких к фазовому синхронизму.

Исходная система. Взаимодействие волн в среде с квадратичным нелинейным откликом описывается следующей системой безразмерных нелинейных дифференциальных уравнений в частных производных относительно медленно меняющихся комплексных амплитуд, далее называемой исходной системой:

$$\begin{aligned}\frac{\partial A_1}{\partial z} + iD_1 \frac{\partial^2 A_1}{\partial t^2} + i\gamma (A_1^* A_2 + A_2^* A_3 + A_3^* A_4) &= 0, \\ \frac{\partial A_2}{\partial z} + \nu_{21} \frac{\partial A_2}{\partial t} + iD_2 \frac{\partial^2 A_2}{\partial t^2} + i\gamma (A_1^2 + 2A_1^* A_3 + 2A_2^* A_4) + i\Delta_{21} k A_2 &= 0, \\ \frac{\partial A_3}{\partial z} + \nu_{31} \frac{\partial A_3}{\partial t} + iD_3 \frac{\partial^2 A_3}{\partial t^2} + 3i\gamma (A_1 A_2 + A_1^* A_4) + i\Delta_{31} k A_3 &= 0, \\ \frac{\partial A_4}{\partial z} + \nu_{41} \frac{\partial A_4}{\partial t} + iD_4 \frac{\partial^2 A_4}{\partial t^2} + 2i\gamma (2A_1 A_3 + A_2^2) + i\Delta_{41} k A_4 &= 0.\end{aligned}$$

со следующими граничными и начальными условиями:

$$\begin{aligned}A_j(z, 0) &= A_j(z, L_t) = 0, \quad j = 1, 2, 3, 4, \\ A_1(0, t) &= A_{10}(t), \quad A_j(0, t) = 0, \quad j = 2, 3, 4.\end{aligned}$$

Консервативность системы. Были получены три инварианта исходной системы. Первый инвариант — аналог закона сохранения энергии, третий инвариант — аналог Гамильтониана системы:

$$I_1 = \int_0^{L_t} \left(|A_1|^2 + |A_2|^2 + |A_3|^2 + |A_4|^2 \right) dt = \text{const.}$$

$$I_2 = \int_0^{L_t} \left[12A_1 \frac{\partial A_1^*}{\partial t} + 6A_2 \frac{\partial A_2^*}{\partial t} + 4A_3 \frac{\partial A_3^*}{\partial t} + 3A_4 \frac{\partial A_4^*}{\partial t} \right] dt = \text{const.}$$

$$I_3 = \int_0^{L_t} \left[- \sum_{j=1}^4 \left(p_j D_j \left| \frac{\partial \bar{A}_j}{\partial t} \right|^2 \right) - \sum_{j=2}^4 \text{Im} \left(p_j \nu_{j1} \bar{A}_j \frac{\partial \bar{A}_j^*}{\partial t} \right) + \sum_{j=2}^4 \left(p_j \Delta_{j1} k |\bar{A}_j|^2 \right) + 2\gamma \text{Re} \left(6\bar{A}_1^{*2} \bar{A}_2 + 12\bar{A}_1^* \bar{A}_2^* \bar{A}_3 + \right. \right. \\ \left. \left. + 12\bar{A}_1^* \bar{A}_3^* \bar{A}_4 + 6\bar{A}_2^{*2} \bar{A}_4 \right) \right] dt = \text{const.}$$

Модифицированная система. В предположении, что $|\Delta_{21}k| \gg 1$, комплексные амплитуды были разложены в степенной ряд по малому параметру: $\mu = \frac{1}{\Delta_{21}k}$. Введённый параметр s отвечает за связь между фазовыми расстройками: $\Delta_{21}k = s \cdot \Delta_{41}k$. После преобразований было получено приближённое решение исходной задачи в виде модифицированной системы:

$$\begin{aligned} \frac{\partial U}{\partial z} + iD_1 \frac{\partial^2 U}{\partial t^2} - \frac{i\gamma^2}{\Delta_{21}k} \cdot (s \cdot 4U|W|^2 + (U|U|^2 + 2U|W|^2 + 3WU^{*2})) &= 0, \\ \frac{\partial W}{\partial z} + v_{31} \frac{\partial W}{\partial t} + iD_3 \frac{\partial^2 W}{\partial t^2} - \frac{3i\gamma^2}{\Delta_{21}k} (s \cdot 4|U|^2 W + (U^3 + 2|U|^2 W)) + i\Delta_{31}kW &= 0, \\ \frac{\partial v_1}{\partial z} + v_{21} \frac{\partial v_1}{\partial t} + iD_2 \frac{\partial^2 v_1}{\partial t^2} &= 0. \end{aligned}$$

с нулевыми граничными условиями и следующими начальными условиями:

$$U|_{z=0} = A_{10}(t), \quad W|_{z=0} = 0, \quad v_1|_{z=0} = \gamma A_{10}^2(t).$$

Инварианты для модифицированной системы были получены аналогичными преобразованиями, что и для исходной системы.

Численное решение. Подобранный на равномерной сетке аппроксимации для нелинейных слагаемых в виде комбинации разностных функции на двух слоях: $A_l(z_k, t_j)$, и $A_l(z_{k+1}, t_j)$, позволили построить консервативную разностную схему для исходной системы уравнений. Полученная схема имеет второй порядок погрешности аппроксимации и сохраняет разностный аналог Гамильтониана:

$$\begin{aligned}
I_3 = \sum_{j=1}^{N_t-1} \tau \Bigg[& - \sum_{l=1}^3 p_l D_l |A_{l\bar{\tau}}|^2 + 6v_{21} \text{Im}(A_2^* A_{2\bar{\tau}}) + 4v_{31} \text{Im}(A_3^* A_{3\bar{\tau}}) + \\
& + 3v_{41} \text{Im}(A_4^* A_{4\bar{\tau}}) + \sum_{j=2}^4 \left(p_j \Delta_{j1} k |\bar{A}_j|^2 \right) + \\
& + 2\gamma \text{Re} \left(6\bar{A}_1^{*2} \bar{A}_2 + 12\bar{A}_1^* \bar{A}_2^* \bar{A}_3 + 12\bar{A}_1^* \bar{A}_3^* \bar{A}_4 + 6\bar{A}_2^{*2} \bar{A}_4 \right) \Bigg].
\end{aligned}$$

Разностный первый инвариант не сохраняется, однако удовлетворяет следующему равенству:

$$I_1 = \sum_{j=1}^{N_t-1} \tau \sum_{l=1}^4 |A_l|^2, \quad \hat{I}_1 = \sum_{j=1}^{N_t-1} \tau \sum_{l=1}^4 |\hat{A}_l|^2, \quad \hat{I}_1 - I_1 = O(h^3).$$

Аналогичное построение разностной схемы было проведено и для модифицированной системы. Последующие численные эксперименты подтвердили сохранение разностных инвариантов и близость решений двух систем.

Приближение длинных импульсов. В приближении длинных импульсов можно считать, что функции U , W зависят только от переменной z . Используя инварианты модифицированной системы, она сводится к дифференциальному уравнению относительно неизвестной амплитуды третьей гармоники p_3 :

$$\begin{aligned}
\frac{dp_3}{dz} &= \pm 7.5 \frac{\gamma^2}{\Delta_{21} k} \chi \sqrt{p_3 f(p_3)}, \\
f(p_3) &= \frac{1}{\chi^2} \frac{16}{25} \left((1-p_3)^3 - \frac{1}{144} p_3 (s \cdot 24(p_3-1) + 9p_3 + 2q - 6)^2 \right), \\
\text{где } \chi &= \sqrt{\frac{16}{25} \left(1 + \frac{(24s+9)^2}{144} \right)}, \quad q = \frac{\Delta_{21} k \Delta_{31} k}{\gamma^2}.
\end{aligned}$$

Процесс генерации определяется параметрами q и s , которые влияют на значения корней уравнения $f(p_3) = 0$. Исследовав систему Штурма [4], пространство интересующего множества параметров было разделено на области, в которых тип корней многочлена сохраняется. В каждой из областей было получено аналитическое решение, описывающее зависимость интенсивности третьей гармоники от поперечной координаты. Было показано совпадение численного решения модифицированной системы и аналитических формул.

Результаты моделирования. Было проведено сравнение результатов аналитического решения, полученного для системы из четырёх уравнений, с численными решениями как для системы из четырёх уравнений, так и для системы из трёх уравнений. Для анализа были выбраны параметры трёх нелинейных кристаллов: KDP, BBO и KBBF. Расчёты проводились в области углов, близких к фазовому синхронизму и таких параметрах, чтобы

нелинейная восприимчивость второго порядка $\chi^{(2)}$ достигала максимального значения [5]. На Рис. 1 цветом обозначена максимальная интенсивность третьей гармоники достигнутая за время симуляции. Аналитическое решение для системы из четырёх уравнений демонстрирует хорошее совпадение с численными результатами для той же системы. Совпадение результатов всех решений выявлено только при начальных значениях интенсивности основного излучения, когда эффект перекачки недостаточно выражен. В остальных случаях при сравнении с системой из трёх уравнений выявлены следующие ключевые различия:

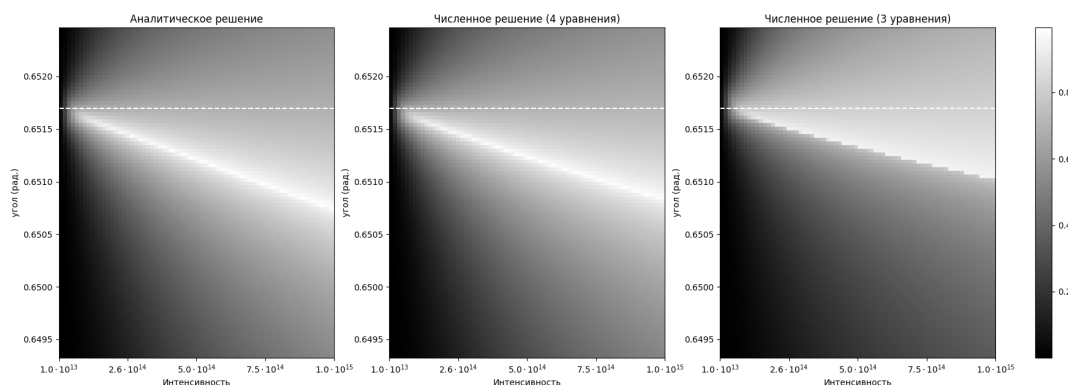


Рис. 1: Сравнение результатов моделирования для кристалла ВВО.

1. Наклон линии максимальной перекачки энергии

В численном решении для четырёх уравнений линия, соответствующая максимальной перекачке энергии в третью гармонику, имеет больший наклон относительно зависимости от угла падения и интенсивности основного излучения. Из-за этого зона оптимальной генерации третьей гармоники оказывается смещена.

2. Область максимальной генерации третьей гармоники

Для системы трёх уравнений линия максимальной интенсивности третьей гармоники имеет резкий переход при приближении снизу к линии максимальной перекачки. В нижней зоне интенсивность третьей гармоники остаётся малой, затем резко возрастает до пика и после этого начинает спадать. При учёте четвёртой гармоники область эффективной генерации получается более гладкой, а зона ниже линии оказывается тоже эффективной.

3. Максимальная интенсивность третьей гармоники

Система четырёх уравнений местами предсказывает большее значение максимальной эффективности третьей гармоники. Одновременно с этим зона в которой можно достичь максимальную интенсивность получается более широкой. Однако для системы из четырёх уравнений при росте значения начальной интенсивности не наблюдается столь большое увеличение

ширины области, в которой может наблюдаться менее слабый режим перекачки.

Выводы. Полученные результаты показывают, что учёт четвёртой гармоники позволяет более точно описать физические процессы в нелинейных кристаллах, таких как KDP, ВВО и КВВФ, особенно в условиях близких к фазовому синхронизму. Использование аналитического решения, основанного на методе многих масштабов и теории Штурма, оказывается достаточным в широком диапазоне начальных интенсивностей в области фазового синхронизма для того, чтобы точно предсказать эффект генерации третьей гармоники. Разработанные модели и методы имеют практическую ценность для оптимизации процессов преобразования частоты в лазерной оптике, позволяя повысить эффективность работы нелинейных оптических устройств.

ЛИТЕРАТУРА

- [1] Theoretical investigations of efficient cascaded third-harmonic generation in quasiphasematched and -mismatched configurations / Y.-Q. Qin, Y.-Y. Zhu, C. Zhang, N.-B. Ming // Journal of the Optical Society of America B. 2003. T. 20, № 1. С. 73–82.
- [2] Kim M. S., Yoon C. S. Theoretical analysis of third-harmonic generation via direct third-order and cascaded second-order processes in CsLiB₆O₁₀ crystals // Phys. Rev. A. 2002. T. 65, № 3. С. 033831.
- [3] Tomov I. V., Van Wonerghem B., Rentzepis P. M. Third-harmonic generation in barium borate // Applied Optics. 1992. T. 31, № 18. С. 3471–3474.
- [4] Prasolov V. V. Polynomials : Springer Science+Business Media, 2009. Т. 11.
- [5] Nikogosyan D. N. Nonlinear optical crystals: a complete survey : Springer Science & Business Media, 2006.

Подход на основе линейного программирования к минимизации пикового использования ресурса в однопроцессорных системах

Работа удостоена диплома 1 степени

Сенюшкина Алина Константиновна

Кафедра автоматизации систем вычислительных комплексов

e-mail: s02210543@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. с.н.с. Балашов Василий Викторович

Введение. В вычислительных системах с ограниченным количеством некоторого ресурса (например, оперативной памяти) при построении расписания вычислений важным критерием является пиковое (максимальное по всему расписанию) использование ресурса. В данной работе рассматривается задача построения однопроцессорного расписания с минимизацией этого критерия. Результат решения задачи может использоваться для оценки объема оперативной памяти, необходимого в проектируемой системе для выполнения заданного набора работ.

В работе [1] для данной задачи предложен детерминированный алгоритм, строящий близкие к оптимальным расписания в случае, если входной граф является последовательно-параллельным. По мере добавления в граф ребер, нарушающих его последовательно-параллельную структуру, получаемые алгоритмом расписания по качеству (т.е. значению целевой функции) приближаются к результатам жадного алгоритма, что указывает на рост отклонения от оптимума.

В работе [2] для данной задачи предложен муравьиный алгоритм, превосходящий по качеству решения жадный алгоритм, однако в силу рандомизированности не гарантирующий качество решения даже на уровне верхней оценки отклонения от оптимума.

Актуальна разработка метода построения оптимального расписания, который не был бы ограничен конкретным классом структуры входного графа.

Сведение к задаче целочисленного линейного программирования. Опишем предложенную в данной работе схему сведения поставленной задачи к задаче ЦЛП.

Для представления расписания, т.е. перестановки вершин графа $G = (V, E)$, воспользуемся подходом на основе переменных предшествования, описанном в [3] и первоначально предложенном в [4].

Поскольку этот подход использует длительности работ, формально положим длительность каждой работы равной 1. В рамках указанного подхода используются бинарные переменные предшествования:

$$\forall i, j \in V, i \neq j :$$

$$m_{ij} = \begin{cases} 1, & \text{если вершина } i \text{ стоит в перестановке раньше вершины } j \\ 0, & \text{в противном случае} \end{cases}$$

Также используются целочисленные переменные $s_j, j \in V$, которые задают время старта соответствующей работы j , в нашем случае равное ее позиции в расписании, считая с 0.

Чтобы совокупность переменных m_{ij} и s_j задавала линейный порядок вершин графа с соблюдением заданного графом частичного порядка, на нее накладываются ограничения (1) – (5) [7, Appendix C].

$$s_j = \sum_{i \in V} m_{ij}, \forall j \in V \quad (1)$$

$$m_{ij} + m_{ji} = 1, \forall i, j \in V, i \neq j \quad (2)$$

$$m_{ij} = 1, \forall (i, j) \in E \quad (3)$$

$$s_i + 1 \leq s_j + |V|(1 - m_{ij}), \forall i, j \in V, i < j \quad (4)$$

$$s_j + 1 \leq s_i + |V|m_{ij}, \forall i, j \in V, i < j \quad (5)$$

Для удобства будем считать, что

$$m_{ii} = 1, i \in V \quad (6)$$

Ограничение (1) определяет, что позиция вершины в перестановке равна числу вершин, предшествующих ей в перестановке. Ограничение (2) требует, чтобы для каждой пары вершин одна из них стояла раньше другой. Ограничение (3) гарантирует частичный порядок на множестве вершин, заданный графом G . Ограничения (4), (5) вместе с ограничением (1) исключают цикличность перестановки, а именно задают условие: для каждой пары вершин i, j либо j расположена в расписании позже чем i (значит, $s_i + 1 \leq s_j$), либо i расположена в расписании позже чем j (значит, $s_j + 1 \leq s_i$). В правых частях ограничений (4), (5) $|V|$ играет роль достаточно большого положительного числа, так чтобы в зависимости от значения m_{ij} только одно из этих двух ограничений было содержательным, а второе заведомо выполнялось. Вводить ограничения (4), (5) для пар i, j при $i > j$ избыточно, поскольку ограничения уже введены для соответствующих пар j, i .

Теперь необходимо ввести целевую функцию на расписании.

Обозначим через $R_k, k \in V$, объем ресурса, который требуется при выполнении работы k , включая ресурс, требуемый для результата самой работы k . Введем переменную F и ограничения для нее:

$$F \geq R_k, \forall k \in V \quad (7)$$

С учетом этих ограничений, исходная задача сводится к минимизации целевой функции, равной переменной F :

$$\min F$$

Теперь надо выразить R_k с использованием переменных m_{ij} . Для этого достаточно найти все работы i , выполняющиеся до работы k , т.е. $m_{ik} = 1$, результат выполнения которых требуется либо самой работе k , либо хотя бы одной работе j , $(i, j) \in E$, выполняющейся после работы k ; для случая $m_{ik} = 1$ при $i = k$ считаем, что результат работы «требуется ей самой», т.к. ресурс под него отводится до высвобождения ресурса работой.

Введем переменные

$$\forall i, k \in V : y_{ik} = \begin{cases} 1, & \text{если } m_{ik} = 1 \text{ и } \exists j \in V : ((i, j) \in E) \wedge (m_{kj} = 1) \\ 1, & \text{если } i = k \\ 0, & \text{в противном случае} \end{cases}$$

При введенных таким способом y_{ik} верно, что

$$R_k = \sum_{i \in V} r_i y_{ik}, \forall k \in V$$

Зададим линейные ограничения на переменные y_{ik} :

$$y_{ii} = 1, \forall i \in V \quad (8)$$

$$y_{ij} = 1, \forall (i, j) \in E \quad (9)$$

$$y_{ik} \leq m_{ik}, \forall i, k : i \neq k, (i, k) \notin E \quad (10)$$

$$y_{ik} \leq \sum_{(i, j) \in E} m_{kj}, \forall i, k, j : i \neq k, (i, j) \in E, (i, k) \notin E \quad (11)$$

$$y_{ik} \geq m_{kj} + m_{ik} - 1, \forall i, k, j : i \neq k, (i, j) \in E, (i, k) \notin E \quad (12)$$

Ограничение (8) показывает, что результат работы «требуется ей самой», т.к. ресурс под него отводится до высвобождения ресурса работой. Ограничение (9) гарантирует, что результат работы i будет храниться до завершения выполнения всех ее прямых потомков j . Ограничение (10) гарантирует, что, если работа k находится в расписании перед работой i , то результат работы i для работы k храниться не будет. Ограничение (11) гарантирует, что, если все прямые потомки j работы i находятся в расписании до работы k , то результат работы i для работы k храниться не будет. Ограничение (12) гарантирует, что результат работы i будет храниться во время выполнения

k , если хотя бы один прямой потомок j работы i стоит в расписании позже k .

Итого, исходная задача сводится к задаче ЦЛП вида:

$$\min F$$

с переменными m_{ij} , s_j , y_{ij} , F и ограничениями (1 – 12).

Число переменных задачи ЦЛП:

$$2|V|^2 + |V| + 1.$$

Число ограничений задачи ЦЛП:

$$\begin{aligned} &|V| + (|V|^2 - |V|) + |E| + (|V|^2 - |V|) + |V| + |V| + |V| + |E| + (|V|^2 - |V| - |E|) + \\ &+ (|V|^2 - |V| * p - |V| + p - |E|) + |V| * |E| - |E| - \sum_{i=1}^{|V|} l_i^2 = \\ &= 4|V|^2 + |V| * |E| - |V| * p - |E| + p - \sum_{i=1}^{|V|} l_i^2, \end{aligned}$$

где p — число вершин без потомков, l_i — число потомков i -ой вершины.

Выбор решателя задач целочисленного линейного программирования. Был проведен обзор решателей задач по критериям доступности, поддержки основных программных платформ и наличия входного языка «lp» для описания задач ЦЛП. Из свободно доступных решателей для сравнения производительности на трех классах графов (случайные, слоистые, треугольные) были выбраны GLPK, SCIP, CBC COIN-OR, поскольку они удовлетворяют всем критериям.

Сравнение производительности на треугольных графах показало, что SCIP в большинстве случаев завершается быстрее (зачастую в ≈ 10 раз) других решателей и, в отличие от них, ни разу не достигает лимита времени выполнения. На случайных и слоистых графах были получены аналогичные результаты.

По совокупности проведенных экспериментов решатель SCIP оказался лучшим по скорости решения задач, поэтому дальнейшие эксперименты проводились на нем.

Запуск решателя при выполнении экспериментов. В связи с существенной зависимостью времени работы решателя от упорядочения входных данных и наличия/отсутствия транзитивных ребер, для основной серии экспериментов была выбрана следующая схема выполнения эксперимента с каждым конкретным набором входных данных (графом работ):

— на 10 процессорных ядрах параллельно запускаются 10 экземпляров решателя, с 5 разными упорядочениями переменных и линейных ограничений в каждой секции входного файла решателя, как при наличии так и при отсутствии добавленных в граф транзитивных ребер;

- на выполнение каждого экземпляра решателя установлен лимит по времени, являющийся параметром эксперимента;
- по достижении лимита времени, все экземпляры решателя, не завершившиеся до этого момента, останавливаются искусственно;
- если хотя бы один экземпляр решателя завершился с обнаружением оптимального решения (статус «optimal solution found» для SCIP), то принудительно завершается выполнение остальных экземпляров.

Экспериментальное исследование масштабируемости подхода. Экспериментальное исследование показало быстрый рост времени работы решателя с увеличением размера входного графа, что характерно для NP-трудных задач. При одинаковом количестве вершин в графе, существенное влияние на время работы решателя оказывает как количество потребляемого работами ресурса (видно на треугольных графах), так и структура графа (видно на слоистых графах с различным количеством пропускаемых ребрами слоев).

Выявлена значительная зависимость времени поиска решения от упорядочения входных данных и от наличия или отсутствия добавленных в граф транзитивных ребер. Зависимость не носит систематического характера, поэтому рекомендуется использовать схему с параллельным запуском экземпляров решателя для различных сочетаний упорядочения и наличия/отсутствия транзитивных ребер при одном и том же наборе входных данных задачи.

Ограничения применимости метода по размеру входного графа зависят от структуры графа. В проведенных экспериментах, для треугольных графов максимальное число вершин графа, для которого было найдено решение, составляло 55; для слоистых – 80 (при отсутствии ребер, пропускающих слои). При этом для треугольного графа с 55 вершинами и 90 ребрами поиск решения занял двое суток, а для слоистого графа с 75 вершинами и 161 ребрами (включая ребра, пропускающие слои) решение не было найдено за 20 суток. Отметим, что в области цифровой обработки сигналов присутствуют графы работ с количеством вершин, не превышающим выявленные пределы: в статье [5] приведены графы работ для операций LU-разложения (55 вершин), QR-разложения (55 вершин), разложения Холецкого (35 вершин).

В ряде случаев время работы решателя можно значительно (в десятки раз, до 100 раз) сократить, если остановить поиск при обнаружении решения, отклоняющегося на 10-15% от оптимального по значению целевой функции.

Результаты. В данной работе получены следующие результаты:

1. Разработана схема сведения задачи построения расписания к задаче ЦЛП;
2. На основе сравнения производительности выбран решатель ЦЛП — SCIP;
3. С учетом выявленной зависимости времени работы решателя от упорядочения входных данных, предложена и реализована многопроцессная схема запуска решателя;
4. Проведено экспериментальное исследование масштабируемости предложенного подхода, показавшее:

- значительную зависимость времени работы решателя ЦЛП от упорядочения вершин и ребер графа работ;
- применимость подхода для входных графов, содержащих до 97 вершин и 143 ребер (один из случайных графов);
- возможность сокращения времени работы решателя на 0,5–99% (в зависимости от входных данных) в случае приемлемости решения, отклоняющегося от оптимального на 5–20%.

ЛИТЕРАТУРА

- [1] Г.А. Шутков. Алгоритм построения расписания для минимизации пикового использования ресурса в однопроцессорной системе // Выпускная квалификационная работа. 2023. С. 41.
- [2] Муравьиный алгоритм для построения однопроцессорного расписания с минимизацией пикового использования ресурса / В.В. Балашов, А.В. Абрамов, А.А. Чупахин и др. // Дискретный анализ и исследование операций. 2024. Т. 31, № 2. С. 5–26.
- [3] Principles of Sequencing and Scheduling / K.R. Baker, D. Trietsch. John Wiley and Sons. 2009. P. 510.
- [4] A.S. Manne. On the Job-Shop Scheduling Problem // Operations Research. 1960. Vol. 8, N. 2. P. 219–223.
- [5] Computing the expected makespan of task graphs in the presence of silent errors / H. Casanova, J. Herrmann, Y. Robert // Parallel Computing. 2017. P. 28.

Сегментация строк в изображениях рукописных документов

Смирнова Виктория Станиславовна

Кафедра математических методов прогнозирования

e-mail: smirnovavs@my.msu.ru

Научный руководитель — д.т.н. проф. Местецкий Леонид Моисеевич

Исторические рукописные документы содержат ценные сведения, которые важно сохранить в цифровом виде для дальнейшего анализа и поиска информации. Одним из ключевых этапов обработки рукописей является сегментация текста на отдельные строки, которая необходима для корректного распознавания текста с помощью современных OCR-систем. Сложности при сегментации рукописей обусловлены разными направлениями линий письма, неоднородностью почерка и межстрочных интервалов.

Цель работы заключается в разработке метода построчной сегментации рукописных документов, основанного на построении векторного поля направлений компонент текста на плоскости страницы рукописи, который будет устойчив к указанным выше сложностям.

Постановка задачи состоит в том, чтобы по каждому изображению восстановить набор осевых линий строк, каждая из которых представляет собой ломаную, проходящую через центры текстовых сегментов и аппроксимирующую интегральную кривую векторного поля локальных направлений сегментов текста. Осевые линии должны отражать как глобальную кривизну, так и локальный наклон строки. Предложенный метод опирается на следующие предположения:

1. Каждая строка имеет линейную структуру и может быть аппроксимирована ломаной с углами перегиба, не превышающие 45° .
2. Преобладающий наклон соседних строк считается постоянным.
3. В пределах одной строки локальные участки текста в малой окрестности имеют схожие углы наклона.
4. Строки преимущественно имеют горизонтальный наклон, то есть угол наклона строк, для которых возможна корректная сегментация составляет 45° .

Основные понятия. Пусть D — бинаризованное исходное изображение страницы.

Определение. Точка $g \in D$ называется узловой, если она лежит в центре локального скопления текстовых пикселей.

Определение. Множество всех узловых точек $G = \{g_1, g_2, \dots, g_N\} \subset D$ называется строковой решёткой. Решётка обеспечивает дискретное представление страницы через узловые точки, по которым восстанавливаются осевые линии строк.

Определение. *Отображение $d: G \rightarrow S$, $S = \{v \in \mathbb{R}^2 \mid \|v\| = 1\}$, где $d(g_i)$ — единичный вектор, указывающий наиболее вероятное направление текста в окрестности узловой точки g_i , называется полем направлений строк.*

Определение. *Непрерывная ломаная $\ell = (v_1, v_2, \dots, v_K)$, $v_k \in G$ упорядоченная по возрастанию абсциссы и проходящая через узловые точки одной строки, называется осевой линией соответствующей строки.*

Подход к решению. Предложенный подход предполагает преобразование исходного изображения в дискретную строковую решётку и построение на ней векторного поля направлений компонент текста. Узловые точки решётки соответствуют центрам локальной концентрации текстовых пикселей. Затем формируется граф потенциальных связей между элементами строковой решётки, на котором задаются операции склейки ребер и склейки фрагментов. Из этого графа, с учётом критерия согласованности рёбер и ориентации векторного поля, выделяются связные цепи, соответствующие осевым линиям строк, аппроксимирующим интегральные кривые векторного поля.

Метод решения состоит из последовательных этапов:

Предобработка изображения включает медианное сглаживание для подавления шумов, адаптивную бинаризацию для учёта вариативности освещения и структуры бумаги, а также морфологические операции для устранения шумовых элементов и улучшения формы текстовых компонент.

Построение строковой решётки осуществляется путём анализа вертикального профиля плотности текстовых пикселей. Изображение делится на равные по ширине вертикальные полосы, в каждой из которых выявляются локальные максимумы данного профиля. Вокруг максимумов формируются области, в которых вычисляются центры тяжести текстовых пикселей, которые принимаются за узловые точки строковой решётки.

Для построения векторного поля направлений используется метод главных компонент [1]. Для каждой узловой точки вычисляется ковариационная матрица координат текстовых пикселей, а локальное направление текста определяется собственным вектором, соответствующим наибольшему собственному значению этой матрицы.

Далее формируется модель соседства узловых точек на основе триангуляции Делоне [2]. То есть строится граф, вершинами которого являются узловые точки, а рёбрами — потенциальные связи между ними. Из рёбер триангуляции выбираются только те, которые максимально согласованы с направлением локального векторного поля в каждой узловой точке, тем самым образуя фрагменты осевых линий.

На следующем этапе выполняется постобработка возможных локальных артефактов: разрывов цепочек и шумовых вершин. Склейка рёбер осуществляется объединением близких по направлению рёбер для устранения разрывов. Для обработки шумовых вершин, создающих ветвления в подгра-

фах, используется критерий отбора инцидентных узловых точек рёбер, максимально согласованных с локальными направлениями данной точки и её соседей. Затем проводится операция склейки близко расположенных фрагментов в непрерывные осевые линии, что обеспечивает восстановление структуры строк.

Эксперимент. Для оценки качества работы метода использовалась выборка из 120 наиболее сложных для сегментации страниц исторических рукописей XIX века, характеризующихся отсутствием строковой структуры, различным межстрочным интервалами и разнообразием почерков. Таким образом, тестовый набор включал 1546 строк, из которых:

- 1482 были сегментированы корректно,
- 181 выделенный алгоритмом элемент не соответствовал отдельным строкам,
- 64 строки не были обнаружены.

Метод продемонстрировал высокую эффективность: подавляющее большинство строк верно обнаружено ($Recall = 0.96$), при этом число лишних фрагментов невелико ($Precision = 0.89$). Гармоничное сочетание этих показателей ($F1-score = 0.92$) подтверждает сбалансированность метрик *Recall* и *Precision* и высокое качество сегментирования строк.

Таким образом, предложенный метод экспериментально подтвердил высокое качество построения сегментации в изображениях рукописных документов, продемонстрировав устойчивость к различным видам почерков и кривизне строк.

ЛИТЕРАТУРА

- [1] Местецкий Л., М. Непрерывная морфология бинарных изображений: фигуры, скелеты, циркуляры // Л., М. Местецкий. М.: Физматлит, 2009. 400, с.
- [2] Jolliffe I., T. Principal Component Analysis // I., T. Jolliffe. 2nd ed. Springer, 2002. 487, p.

Методы машинного обучения с подкреплением в линейно-квадратичных задачах управления

Работа удостоена диплома 1 степени

Смовж Илья Николаевич

Кафедра оптимального управления

e-mail: smovzh.ilya@mail.ru

Научный руководитель — д.ф.-м.н. проф. Мельников Николай Борисович

Введение. Во многих приложениях возникает задача стабилизации линейной системы управления в непрерывном времени

$$\begin{cases} \dot{x} = Ax + Bu, \\ y = Cx, \end{cases} \quad x \in \mathbb{R}^n, \quad y \in \mathbb{R}^r, \quad u \in \mathbb{R}^m, \quad (1)$$

где A , B , C — постоянные матрицы. При заданных матрицах методы стабилизации таких систем с помощью линейно-квадратичного регулятора хорошо известны (см., напр., [1] и цитированную там литературу). Мы рассматриваем задачу стабилизации системы с частично неизвестной динамикой: матрица A неизвестна. Эти методы не предполагают предварительной идентификации системы.

Для случая, когда наблюдается весь вектор состояния x ($C = I$), были разработаны методы V -обучения с подкреплением: методы u -итераций и V -итераций [2]. Для случая, когда наблюдается лишь часть вектора состояния x ($C \neq I$), метод u -итераций был предложен в работе [3] в предположении, что целевая траектория описывается линейной динамикой (в. ч., может быть неподвижной точкой).

В нашей работе для случая, когда наблюдается лишь часть вектора состояния x , предложен метод V -итераций и исследована его сходимости. Как и в работе [3], предполагается, что целевая траектория описывается линейной динамикой. Чтобы не накладывать дополнительных ограничений на целевую траекторию, в функционал добавлен дисконтирующий множитель, который часто используется в алгоритмах обучения с подкреплением (см., напр., [1]).

Линейно-квадратичный регулятор. Рассматриваем линейную систему управления (1), предполагая, что вектор состояния x и выход y не совпадают. Решаем задачу стабилизации данной системы в нуле. Для этого используем квадратичный критерий качества

$$J(u(\cdot)) = \frac{1}{2} \int_t^\infty e^{-\gamma(\tau-t)} [(Cx - y_d)^T Q (Cx - y_d) + u^T R u] d\tau \rightarrow \min_{u(\cdot)}, \quad (2)$$

где $Q, R > 0$ — известные симметричные матрицы. Предполагаем, что целевая траектория y_d удовлетворяет уравнению

$$\dot{y}_d = F y_d, \quad (3)$$

где F — постоянная матрица.

Приводим задачу (1)–(2) с условием (3) к более удобному виду. Вводим обозначения $X = (x^T, y_d^T)^T$ и $Y = (Cx - y_d)$. Объединяя уравнения (1) и (3), получим расширенную систему

$$\begin{cases} \dot{X} = \begin{pmatrix} A & 0 \\ 0 & F \end{pmatrix} X + \begin{pmatrix} B \\ 0 \end{pmatrix} u \equiv MX + B_1 u, \\ Y = C_1 X, \end{cases} \quad (4)$$

где $C_1 = (C \ -I)$. Систему считаем управляемой и наблюдаемой. Критерий качества (2) примет вид

$$J(u(\cdot)) = \frac{1}{2} \int_t^\infty e^{-\gamma(\tau-t)} [X^T C_1^T Q C_1 X + u^T R u] d\tau \rightarrow \min_{u(\cdot)}. \quad (5)$$

Динамическое программирование. При полностью известной динамике системы задача решается с помощью динамического программирования. Для задачи с расширенной системой (4)–(5) функция цены имеет вид квадратичной формы

$$V(X) = \frac{1}{2} X^T P X,$$

где матрица $P > 0$ является решением уравнения Риккати

$$Ric(P) \equiv M^T P + P M - \gamma P + C_1^T Q C_1 - P B_1 R^{-1} B_1^T P = 0,$$

а оптимальное управление имеет вид линейной обратной связи

$$u(X) = -K_1 X, \quad K_1 \equiv R^{-1} B_1^T P. \quad (6)$$

При оптимальном управлении (6) функция цены является функцией Ляпунова для системы

$$\dot{X} = (M - B_1 K_1) X,$$

то есть нулевое решение асимптотически устойчиво.

Обучение с подкреплением. При неизвестной матрице A нет возможности решать уравнение Риккати. Вместо этого итерационно обновляем функцию цены $V_i(X)$ и управление $u_i(X)$ по наблюдениям вектора состояния X на отрезках длины $T > 0$. Таким образом компенсируется отсутствие полной информации о динамике системы (4).

В нашей работе предложен метод V -итераций. На каждом шаге происходит вычисление P_{i+1} из уравнения

$$\begin{aligned} X^T(t) P_{i+1} X(t) &= \int_t^{t+T} e^{-\gamma(\tau-t)} [X^T C_1^T Q C_1 X + u_i^T R u_i] d\tau \\ &+ e^{-\gamma T} X^T(t+T) P_i X(t+T), \end{aligned} \quad (7)$$

после чего обновляется управление

$$u_{i+1} = -R^{-1} B_1^T P_{i+1} X. \quad (8)$$

Для этого метода сформулированы и доказаны следующие теоремы.

Теорема 1. Метод V -итераций (7)–(8) эквивалентен квазиньютоновскому методу решения уравнения Риккати:

$$P_{i+1} = P_i - J_i^{-1} Ric(P_i),$$

где

$$J_i^{-1} P = \left(Ric'_{P_i} \right)^{-1} \left(P - e^{-\gamma T} e^{M_i^T T} P e^{M_i T} \right).$$

Здесь J_i – приближение производной Фреше $Ric'_{P_i}(P) = M_i^T P + P M_i$ оператора Ric в точке P_i , а $M_i = (M - B_1 R^{-1} B_1^T P_i)$.

Теорема 2. Пусть $P_0 > 0$ и $Ric(P_0) \leq 0$, т. е. начальное управление $u_0(X)$ – стабилизирующее. Тогда

1. имеет место монотонная сходимость

$$\begin{cases} \lim_{i \rightarrow \infty} P_i = P^*, \\ 0 < P^* \leq \dots \leq P_{i+1} \leq P_i \leq \dots \leq P_1 \leq P_0, \end{cases}$$

где $Ric(P^*) = 0$;

2. $P_i > 0$ и $Ric(P_i) \leq 0$, т. е. управление $u_i(X)$ – стабилизирующее на каждой итерации.

Применение метода V -итераций продемонстрировано нами на примерах.

ЛИТЕРАТУРА

- [1] Ронжина М.И., Мельников Н.Б. Машинное обучение и оптимальное управление. М.: РГУ нефти и газа (НИУ) имени И.М. Губкина, 2024.
- [2] Lewis F.L., Vrabie D., Vamvoudakis K.G. Reinforcement learning and feedback control // IEEE control systems magazine. 2012. P. 76–105.
- [3] Modares H., Lewis F.L. Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning // IEEE transactions on automatic control. 2014. Vol. 59. P. 3051–3056.

Разработка эффективных методов построения ансамблевых регрессионных линейных моделей, основанных на минимизации ошибки

Работа удостоена диплома 3 степени

Суржанский Александр Сергеевич

Кафедра математических методов прогнозирования

e-mail: alexandr.surzhanskiy@yandex.ru

Научный руководитель — д.ф.-м.н. проф. Сенько Олег Валентинович

Введение. Ансамблевые методы машинного обучения занимают важное место в современной прикладной математике и интеллектуальном анализе данных. Их основная идея — объединение предсказаний множества базовых моделей для повышения точности, устойчивости и обобщающей способности итогового решения. Благодаря этому ансамбли широко применяются в таких критически важных областях, как медицина, финансы, промышленная диагностика и прогнозирование сложных процессов. Классические подходы, такие как случайный лес и градиентный бустинг, эффективны, но не позволяют явно управлять весами базовых моделей, что снижает гибкость и интерпретируемость решений.

Одним из перспективных направлений является построение ансамблей с использованием *выпуклой оптимизации* — то есть линейного объединения базовых предсказаний с весами, найденными путём минимизации ошибки прогноза при заданных ограничениях. Такой подход позволяет явно контролировать структуру ансамбля, добиваться интерпретируемости результатов и использовать более широкие классы базовых моделей [1].

Целью данной работы является разработка эффективных методов построения ансамблевых регрессионных моделей, в которых итоговое предсказание формируется как выпуклая комбинация базовых решений, оптимизированная по критерию точности. Предлагаемые методы направлены на улучшение устойчивости ансамблей и адаптацию к особенностям конкретных задач.

Актуальность работы обусловлена необходимостью создания универсальных и одновременно интерпретируемых моделей, способных конкурировать с традиционными ансамблями в условиях ограниченного объёма данных, высокой размерности признакового пространства и разнообразия источников шума. Подобные модели находят применение в задачах, где важно не только получить точный прогноз, но и обосновать его структуру и надёжность.

Постановка задачи. Пусть на валидационной выборке $\mathcal{D}_{\text{val}} = \{(x_i, y_i)\}_{i=1}^n$ известны отклики K базовых моделей $F_{i,k} = f_k(x_i)$, где $F \in \mathbb{R}^{n \times K}$. Требуется подобрать неотрицательные веса $\alpha \in \mathbb{R}^K$, суммирующиеся к единице, чтобы

минимизировать среднеквадратичную ошибку ансамбля

$$\begin{aligned} \min_{\alpha} \quad & \|F\alpha - y\|_2^2 \\ \text{subject to} \quad & \alpha_k \geq 0, \quad \sum_{k=1}^K \alpha_k = 1 \end{aligned} \quad (1)$$

Обозначив $G = F^\top F \in \mathbb{R}^{K \times K}$ и $b = F^\top y$, задачу (1) можно свести к эквивалентной квадратичной программе

$$\min_{\alpha \in \Delta} \frac{1}{2} \alpha^\top G \alpha - b^\top \alpha, \quad \Delta = \left\{ \alpha \mid \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1 \right\}. \quad (2)$$

Матрица G симметрична и положительно полуопределённа, а линейные ограничения описывают $(K - 1)$ -мерный симплекс. Функция (2) уже содержит встроенную регуляризацию: условия $\alpha_k \geq 0$ и $\sum_k \alpha_k = 1$ ограничивают эффективную сложность ансамбля, поэтому явный штраф (Lasso или Ridge) не требуется.

Решение задачи оптимизации. Для реализации задачи (2) применена библиотека `cvxpy` [2], позволяющая описывать конвекс-программы на декларативном уровне. Переход от первоначального критерия $\|F\alpha - y\|_2^2$ к квадратичной форме (2) позволяет сформулировать задачу ансамблирования как строго выпуклую программу на симплексе, гарантирующую единственность решения без дополнительных регуляризаторов. Для её решения *наиболее надёжным* оказался пакет `SCS` [3], тогда как `ECOS` страдает от численных сбоев на вырожденных матрицах, а `OSQP` требует дополнительной проекции, снижающей точность.

Модели на основе выпуклой оптимизации. Далее будет представлено подробное описание *пяти схем ансамблирования*, основанных на линейно-выпуклом объединении базовых предсказаний. Во всех случаях веса итогового ансамбля вычисляются путём решения задачи (2); различия между методами заключаются в способе построения матрицы откликов F и в стратегии генерации базовых моделей. Для каждой схемы представлены концептуальная идея и краткий анализ ключевых свойств модели.

Single-Feature Convex Ensemble (SF-CE). Для каждого признака x_j независимо обучается *одномерная модель* регрессии $f_j: \mathbb{R} \rightarrow \mathbb{R}$. Финальный ансамбль представляет собой выпуклую комбинацию этих d моделей, веса которой подбираются на валидационной выборке.

Особенности:

- Интерпретируемость. Веса α_j имеют прямую семантику: каждый из них количественно отражает вклад соответствующего признака x_j в итоговое предсказание.
- Масштабируемость. Обучение базовых моделей выполняется независимо для каждого признака, что обеспечивает линейную сложность по размер-

ности пространства признаков и допускает эффективную параллелизацию.

Per-Feature Convex Ensemble (Per-SF-CE). Одномерный регрессор $f_j: \mathbb{R} \rightarrow \mathbb{R}$ обучается на каждом признаке x_j , при этом тип модели (например, линейная регрессия или дерево решений) *выбирается отдельно* для каждого признака на основе качества предсказаний на валидационной выборке. Итоговое предсказание формируется как выпуклая комбинация этих d моделей.

Особенности:

- Адаптивный выбор модели. Использование разнородных моделей повышает гибкость ансамбля и позволяет точнее подстраиваться под структуру данных на уровне отдельных признаков.
- Стабилизация за счёт выпуклой оптимизации. Несмотря на потенциальный риск переобучения при повышенной гибкости, итоговая модель стабилизируется благодаря выпуклым ограничениям на веса.

Single-Feature Convex Residual Ensemble (SF-CRE). Ансамбль формируется итеративно: на каждом шаге добавляется одномерный регрессор $f^{(t)}$, обученный на одном из признаков x_j , отобранном по критерию наибольшего прироста качества на валидационной выборке. Для увеличения разнообразия в ансамбле используется дивергентная корректировка [4]: текущее предсказание сравнивается не с истинным ответом напрямую, а с отклонением от уже накопленного ансамблевого ответа. Это позволяет уменьшить корреляцию между базовыми моделями и повысить устойчивость к переобучению. Финальное предсказание формируется как выпуклая комбинация T шагов.

Особенности:

- Дивергентность ансамбля. Параметр $\mu \in (0, 1]$ управляет степенью различия между шагами ансамбля: большие значения μ способствуют формированию более разнообразных, некоррелированных предсказателей.

Stochastic Convex Residual Ensemble (Stoch-CRE). Модель формируется в два уровня. На первом уровне независимо обучаются N ансамблей типа SF-CRE, каждый из которых строится на случайном подмножестве признаков и использует дивергентную итеративную схему с локальной адаптацией. Это позволяет получить разнообразные предсказания, устойчивые к шуму и переобучению. На втором уровне предсказания всех N моделей объединяются в выпуклую линейную комбинацию, где веса оптимизируются по валидационной выборке аналогично задаче (2).

Особенности:

- Стохастическая диверсификация. Аналогично классическим методам бэггинга, обучение нескольких независимых SF-CRE на случайных подмножествах признаков снижает взаимную корреляцию моделей и уменьшает дисперсию ансамбля.

- Двухуровневая структура. Дивергентные SF-CRE первого уровня создают устойчивые частичные представления, а выпуклая оптимизация на втором уровне объединяет их, подстраивая итоговый прогноз под валидационные данные.

Full-Feature Convex Residual Ensemble (FF-CRE). Ансамбль строится итеративно: на каждом шаге регрессионная модель $f^{(t)}: \mathbb{R}^d \rightarrow \mathbb{R}$ обучается на всём множестве признаков, при этом с заданной вероятностью отдельные признаки могут быть исключены, а объекты — заменены случайным подмножеством выборки с возвращением (бутстреп). В качестве целевой переменной используется скорректированный отклик, отражающий дивергентность относительно предыдущих шагов. Итоговое предсказание формируется как выпуклая комбинация T моделей.

Особенности:

- Глобальность модели. Каждая базовая модель обучается на всём множестве признаков, что позволяет улавливать сложные межпризнаковые зависимости и взаимодействия, недоступные одномерным схемам.
- Стохастическое разрежение. Возможность случайного исключения признаков и применения бутстрапа по объектам повышает разнообразие между шагами ансамбля и снижает переобучение.
- Контроль избыточности. Дивергентное обновление через параметр μ уменьшает корреляцию между базовыми моделями, обеспечивая устойчивость итогового предсказания при сохранении выразительности.

Вычислительные эксперименты. Разработанные методы были протестированы на двенадцати табличных датасетах из области химии. Для оценки качества использовался коэффициент детерминации R^2 , рассчитываемый на тестовой подвыборке. Все данные предварительно делились на обучающую, валидационную и тестовую части в пропорции 60 % / 20 % / 20 %.

Для сопоставления результатов использовались стандартные реализации моделей регрессии: линейные (Lasso, Ridge), а также ансамблевые (RandomForestRegressor, GradientBoostingRegressor, CatBoostRegressor, XGBRegressor). Все модели применялись с параметрами по умолчанию.

В процессе вычислительных экспериментов выполнялась настройка гиперпараметров, управляющих структурой и поведением предложенных моделей ансамблирования. Особое внимание уделялось параметрам, влияющим на устойчивость, разнообразие и аппроксимационные свойства ансамбля. В рассмотрение включались следующие гиперпараметры:

- коэффициент дивергентности $\mu \in [0.05, 0.95]$ — определяет степень отличия текущей базовой модели от накопленного предсказания ансамбля;
- тип базовой модели $\in \{\text{линейная регрессия, дерево решений}\}$ — указывает форму регрессора, используемого на каждом шаге;
- число итераций ансамблирования $T \in \{2, \dots, 10\}$ — задаёт глубину построения ансамбля;

- вероятность случайного исключения признаков $p_{\text{drop}} \in [0, 0.5]$ — отвечает за стохастическое разрежение входных данных;
- флаг применения бутстрэпа по объектам $\in \{0, 1\}$ — определяет, используется ли выборка с возвращением при обучении базовых моделей.

Все параметры, за исключением коэффициента μ , подбирались автоматически с помощью библиотеки `Optuna` на валидационной подвыборке. Такой подход позволил эффективно исследовать пространство допустимых конфигураций и адаптировать архитектуру модели к конкретной задаче.

Коэффициент дивергентности μ настраивался отдельно для каждой задачи с использованием процедуры 5-кратной перекрёстной проверки на обучающей выборке. Это обеспечило устойчивость полученного значения и позволило снизить риск переобучения, связанный с прямой оптимизацией по валидации.

Анализ подобранных гиперпараметров. Анализ параметров показал, что при малом числе шагов качество модели существенно колеблется: минимумы наблюдаются около $\mu \approx 0.6$ – 0.7 , что указывает на высокую чувствительность к параметру. При увеличении глубины ансамбля зависимость сглаживается, а оптимальная зона смещается к средним значениям $\mu \in [0.4, 0.6]$. Для длинного ансамбля кривая становится почти плоской в интервале $\mu \in [0.15, 0.75]$, подтверждая, что увеличение глубины стабилизирует модель и снижает чувствительность к выбору параметра μ . Резкое падение качества при $\mu \rightarrow 1$ связано с избыточной «дивергентностью», в результате которой базовые модели теряют связь с истинным откликом.

Характерные закономерности, выявленные по результатам настройки моделей, можно обобщить следующим образом:

- *Тип базовой модели.* В двух третях задач предпочтение отдано деревьям решений (8 выборов из 12), что свидетельствует о нелинейной природе зависимостей в исходных признаках.
- *Число шагов T .* Большинство конфигураций сконцентрировалось в диапазоне 9–10 шагов, подтверждая, что умеренно глубокий ансамбль обеспечивает наилучший баланс между смещением и дисперсией.
- *Вероятность исключения признаков p_{drop} .* Оптимальные значения лежат в интервале 0.10–0.25, то есть умеренное стохастическое разрежение повышает разнообразие базовых моделей, не ухудшая информативность.
- *Число активных компонент в ансамбле.* В итоговых моделях число базовых предсказателей с ненулевыми весами обычно не превышает пяти, что свидетельствует о естественной склонности решения к компактности даже без явной регуляризации.
- *Бутстрэп по объектам.* Во всех двенадцати задачах оптимальный набор параметров включал использование бутстрэпа, что подчёркивает его роль в уменьшении дисперсии и повышении устойчивости модели.

Основные результаты.

- Разработанные методы демонстрируют высокое качество *на всех тестируемых наборах данных*, что подтверждает их эффективность на табличных данных с ограниченным числом объектов и высоким уровнем шума.
- Алгоритм Full-Feature Convex Residual Ensemble (FF-CRE) достигает наилучшего значения R^2 *на шести* из двенадцати датасетов, что делает его лидером среди предложенных моделей.
- Модификация Stoch SF-CRE также показывает хорошие результаты, особенно на небольших наборах данных с выраженной вариативностью. Она обеспечивает наилучшее качество на двух датасетах, подтверждая свою устойчивость при наличии шума.
- Базовая схема SF-CE продемонстрировала наивысшее качество на двух датасетах, что указывает на её эффективность при малом объёме данных и относительно простой структуре признаков.
- Схема Per SF-CE хотя и не занимает лидирующие позиции, но в отдельных задачах (например, на данных с умеренной размерностью) показывает результат, близкий к лучшим методам.

Заключение. В данной работе предложены новые схемы построения ансамблевых регрессионных моделей, основанные на выпуклом объединении предсказаний базовых алгоритмов. Были разработаны и реализованы несколько модификаций, различающихся по способу генерации базовых моделей и структуре их агрегации.

Для оценки эффективности предложенных моделей проведён ряд вычислительных экспериментов с использованием современных инструментов машинного обучения. Полученные результаты демонстрируют преимущество разработанных методов по сравнению с классическими алгоритмами, особенно в задачах с небольшим числом объектов и высоким уровнем шума.

ЛИТЕРАТУРА

- [1] Stephen Boyd, Lieven Vandenberghe Convex Optimization // Cambridge University Press, 2004.
- [2] Steven Diamond, Stephen Boyd CVXPY: A Python-Embedded Modeling Language for Convex Optimization // Journal of Machine Learning Research, 2016.
- [3] Conic Optimization via Operator Splitting and Homogeneous Self-Dual Embedding / Brendan O'Donoghue, Eric Chu, Neal Parikh, Stephen Boyd // Journal of Optimization Theory and Applications, 2016.
- [4] Two-Level Regression Method Using Ensembles of Trees with Optimal Divergence / Yu. I. Zhuravlev, O. V. Senko, A. A. Dokukin, N. N. Kiselyova, I. A. Saenko // Doklady Mathematics, 2021.

Методы приближения и предсказания канала связи на основе канонического представления малого ранга

Таумурзаев Ахматали Канаматович

Кафедра вычислительных технологий и моделирования

e-mail: akhmat.taumurzaev@yandex.ru

Научный руководитель — к.ф.-м.н. доц. Замарашкин Николай Леонидович

Введение. Тензоры и тензорные разложения находят всё более широкое применение в решении вычислительных задач из различных областей. Одним из важнейших свойств малоранговых приближений является свойство фильтрации шума [1]. Каноническое разложение может быть использовано для решения задач в области беспроводной связи [2]. Канал связи можно моделировать в виде суммы двух тензоров: полезного сигнала, обладающего малоранговой структурой, и шума. Таким образом, малоранговые представления можно использовать для фильтрации канала от шума. Также важной задачей является предсказание состояния канала в будущем для ускорения его обработки.

Задача фильтрации шума. Рассмотрим тензор $\hat{T} = T + N$, где T – тензор ранга R , а элементы N – это независимые нормально распределённые случайные величины. Пусть $\hat{T}_R$ – каноническое приближение ранга R тензора $\hat{T}$, найденное при помощи алгоритма ALS [3]. В статье [1] показано, что при выполнении аппроксимационного условия $\|\hat{T}_R - \hat{T}\|_F \leq \|T - \hat{T}\|_F = \|N\|_F$ справедлива следующая оценка

$$\|\hat{T}_R - T\| \lesssim \sqrt{\frac{dRM^{\frac{1}{d}}}{M}} \mathbb{E}\|N\|_F, \quad (1)$$

где $M = n_1 \dots n_d$ – количество его элементов. Из этой оценки следует, что при увеличении d точность приближения T будет увеличиваться. Тем самым, искусственно увеличив размерность до максимально возможной, можно добиться оптимальной фильтрации шума. Но при увеличении размерности значительно ухудшается сходимость ALS.

Сходимость ALS в существенной степени зависит от начального приближения. Предполагается, что из канонического представления тензора можно получить хорошее начальное приближение для этого же тензора, но с введёнными искусственными размерностями. Рассмотрим для простоты 3-мерный одноранговый тензор $Z = a \circ b \circ c \in \mathbb{C}^{n_1 \times n_2 \times n_3}$. Не умаляя общности, предположим, что n_1 раскладывается на множители $n_1 = m_1 m_2$. Составим из вектора $a \in \mathbb{C}^{n_1}$ матрицу $A \in \mathbb{C}^{m_1 \times m_2}$. Также предположим, что полученная матрица является одноранговой. Тогда верно $A = a_1 a_2^T$, $a_1 \in \mathbb{C}^{m_1}$, $a_2 \in \mathbb{C}^{m_2}$. Тогда вектор a представим в виде $a = a_1 \otimes a_2$, где $\otimes$ – кронекерово произведение. Из этого следует, что если $\tilde{T} \in \mathbb{C}^{m_1 \times m_2 \times n_2 \times n_3}$ – 4-мерное представление тензора Z , то для него верно $\tilde{Z} = a_1 \circ a_2 \circ b \circ c$. Данный подход позволяет увеличивать размерность канонического разложения. Отметим, что случай $R > 1$

отличается лишь тем, что данные действия проводятся для всех столбцов каждой факторной матрицы. Таким образом, возвращаясь к T и $\tilde{T}$, с помощью данного подхода можно из факторных матриц $A_1, \dots, A_d$ получить $[\tilde{A}_1, \dots, \tilde{A}_s] \approx \tilde{T}$.

Особая структура тензоров Предложенный подход применим к тензорам, для которых выполнены условия:

- их ранг не изменяется при увеличении размерности,
- матрицы, получаемые из столбцов факторных матриц с помощью операции *reshape*, являются одноранговыми.

Примером целого класса, удовлетворяющего данным требованиям, являются тензоры, имеющие вид

$$T_{j_1 \dots j_d} = \sum_{r=1}^R \gamma_r \exp(i\phi_r J), \quad T \in \mathbb{C}^{n_1 \times \dots \times n_d}, \quad (2)$$

где $J = j_1(n_2 \dots n_d) + j_2(n_3 \dots n_d) + \dots + j_{d-1}n_d + j_d$ — линейный индекс, γ_r — коэффициенты, ϕ_r — фазовые параметры. Такие тензоры встречаются в области беспроводных технологий [2].

Предлагаемый алгоритм Таким образом, можно составить алгоритм, который будет принимать на вход тензор и находить каноническое разложение максимально возможной размерности. Итерация алгоритма будет состоять из повышения размерности факторных матриц, полученных на предыдущей итерации описанным ранее способом, и применения небольшого количества итераций ALS для уточнения полученного приближения. Численные эксперименты показали, что данный алгоритм позволяет добиться оптимальной фильтрации шума без потерь в сходимости.

Задача предсказания Рассмотрим тензор $T \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ вида (2), положив для простоты $d = 3$. Пусть теперь фазовые параметры $\phi_1, \dots, \phi_R$ непрерывно зависят от переменной t : $\phi_k = \phi_k(t)$. Введём последовательность зашумленных тензоров $\hat{T}_k = T_k + N$, где $T_k = T(t_k)$, а N — белый шум. Задача предсказания заключается в построении $T_{m+1}, T_{m+2}, \dots$ по известным $\hat{T}_1, \dots, \hat{T}_m$ в предположении, что разница между t_k мала.

Обозначим факторные матрицы тензора $T_k \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ соответственно как $A_k \in \mathbb{C}^{n_1 \times R}$, $B_k \in \mathbb{C}^{n_2 \times R}$, $C_k \in \mathbb{C}^{n_3 \times R}$. В силу структуры тензоров T_k и предположения о малости разности между t_k допустим, элементы матриц $A_1, \dots, A_k, \dots$ связаны следующим образом:

$$\{A_k\}_{j_1, j_2} = \gamma_{j_1, j_2} \cdot \{A_{k-1}\}_{j_1, j_2}$$

где γ_{j_1, j_2} — неизвестная константа, зависящая от j_1, j_2 . Так как T_k нам не известны, а известны лишь зашумлённые $\hat{T}_k$, то и для матриц A_k могут быть известны лишь приближенные $\hat{A}_k$. Таким образом, последовательность $\{\hat{A}_k\}_{j_1, j_2}$ и $\gamma_{i, j}$ можно собрать в модельный вектор $x_k = \begin{bmatrix} \{\hat{A}_k\}_{j_1, j_2} \\ \gamma_{j_1, j_2} \end{bmatrix}$ при фиксирован-

ных j_1, j_2 можно рассмотреть как динамическую систему

$$\begin{cases} x_k = \begin{bmatrix} \{A_k\}_{j_1, j_2} \\ \gamma_{j_1, j_2} \end{bmatrix} = f(x_{k-1}) + w_k, \\ z_k = h(x_k) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T \cdot x_k + v_k = \{\hat{A}_k\}_{j_1, j_2}, \end{cases}$$

где f – нелинейная функция, действие которой определено как $f\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} xy \\ y \end{bmatrix}$,

w_k и v_k – случайные величины, отражающие неточности модели и вычислений соответственно. Применяя аналогичные рассуждения для матриц B_k и C_k , получим набор динамических систем и ряд зашумлённых измерений для каждой из них. Распространённым способом оценки подобных систем является фильтр Калмана.

Результаты применения фильтра Данный подход был применён к тензорам, полученным в процессе моделирования канала связи между станцией антенн и движущимся пользователем при различных скоростях. Эксперименты показали, что данный подход хорошо проявляет себя при небольших скоростях, но при больших скоростях фильтр начинает "разваливаться". Предполагается, что данная проблема возникает из-за неустойчивости малоранговых приближений.

Заключение В рамках данной работы были исследованы две задачи: фильтрация шума и предсказание, рассматриваемые на классе тензоров со специальной структурой. Для решения задачи фильтрации шума разработан алгоритм, основанный на методе ALS и учитывающий особенности структуры тензоров. Предложенный подход обеспечивает оптимальное подавление шума без ухудшения скорости сходимости. Вторая задача – предсказание – решалась с использованием моделирования последовательностей элементов факторных матриц в виде зашумлённых динамических систем. Для их анализа применён фильтр Калмана. Данный подход позволяет прогнозировать состояние канала связи при небольших скоростях пользователей, однако при больших скоростях фильтр становится неустойчивым. Данная проблема требует дальнейшего рассмотрения.

ЛИТЕРАТУРА

- [1] Petrov S. V., Zamarashkin N. L. Theoretical Bounds for Noise Filtration using Low-Rank Tensor Approximations // arXiv preprint arXiv:2312.02088, 2023.
- [2] De Almeida A. Tensor modeling and signal processing for wireless communication systems, 2007.
- [3] Comon P., Luciani X., De Almeida A.L.F. Tensor decompositions, alternating least squares and other tales // Journal of Chemometrics: A Journal of the Chemometrics Society, vol. 23, N 7. P. 393–405.

Построение модели управления мобильным роботом при наличии фазовых ограничений

Работа удостоена диплома 3 степени
Титушин Александр Дмитриевич

Кафедра исследования операций
e-mail: titushkin303@mail.ru

Научный руководитель — к.ф.-м.н. доц. Дарьина Анна Николаевна

В последние десятилетия мобильные роботы активно внедряются в различные сферы человеческой деятельности — от промышленных и сельскохозяйственных применений до логистики и исследований в экстремальных условиях. Этот стремительный рост применения робототехнических систем сопровождается повышенным вниманием к решению задач оптимизации их управления. Современные мобильные роботы могут функционировать в двух основных режимах: под управлением оператора (ручное управление) или автономно, следуя заложенным алгоритмам (автоматическое управление). Особый интерес для исследователей представляет именно автоматизация колесных мобильных платформ, однако, несмотря на значительное количество фундаментальных и прикладных работ в этой области, многие проблемы управления остаются не до конца решенными.

Важнейшей задачей управления мобильным роботом является обеспечение его безопасного и эффективного движения в окружающей среде, полной препятствий и динамических изменений. Актуальность приобретают задачи изучения траекторий движения и синтеза оптимальных алгоритмов управления. Эти исследования обусловлены постоянно растущими требованиями к точности робототехнических систем, необходимостью обеспечения плавных динамических перемещений и оптимизации энергопотребления.

Существует множество методов, разработанных для решения подобных задач. Некоторые из наиболее известных и хорошо изученных методов включают: принцип максимума Понтрягина [1], метод штрафов.

Построение оптимального маршрута в предсказуемой среде со стационарными препятствиями — это базовая задача, имеющая важное практическое значение для применения в условиях, где объекты-ограничения не меняют свое расположение (склады, производственные помещения и др.). Для решения этой задачи могут использоваться алгоритмы, такие как A-star [2] или алгоритм Дейкстры [3].

Цель работы заключается в построении модели управления мобильным роботом при наличии фазовых ограничений, а также применении различных методов решения задачи быстрогодействия к разработанной модели и провести сравнительный анализ эффективности рассмотренных методов на различных тестовых сценариях.

Постановка задачи Рассмотрим объект, поведение которого определяется системой обыкновенных дифференциальных уравнений

$$\dot{x}(t) = u_1 \cos(\theta), \quad (1)$$

$$\dot{y}(t) = u_1 \sin(\theta), \quad (2)$$

$$\dot{\theta}(t) = \frac{u_1}{L} \operatorname{tg}(u_2), \quad (3)$$

где

- $(x, y) \in \mathbb{R}^2$ — вектор траектории объекта;
- $u = u_2 \in U \subseteq \mathbb{R}$ — управление, угол поворота (положительный против часовой стрелки);
- u_1 — линейная скорость (будем рассматривать постоянную скорость, движение равномерное прямолинейное);
- θ — ориентация робота относительно оси Ох;
- L — расстояние между передней и задней осями колес.

Управление $u_2(t)$ ищется в классе непрерывных функций, $u_2 \in [-u_{2\max}, u_{2\max}]$, где $u_{2\max} < \frac{\pi}{2}$ заданный максимальный угол поворота.

Заданы начальные условия $(x_0, y_0) \in \mathbb{R}^2$, задан минимальный и максимальный углы поворота робота $u_{2\max}$ и терминальные условия, заданные соотношением:

$$(x_{tf}, y_{tf}) \in P = \{(x_p, y_p) | x^- < x_p < x^+, y^- < y_p < y^+\} \subseteq \mathbb{R}^2. \quad (4)$$

Заданы границы прямоугольной области $(x^-, y^-), (x^+, y^+)$. Заданы статические ограничения следующего вида:

$$\varphi_j(x, y) \leq 0, \quad (5)$$

где $j = 1, \dots, K$, $K \in \mathbb{N}$ — количество препятствий.

В рассматриваемых условиях ставится задача быстрогодействия. Требуется определить управление u_2 и соответствующий ему вектор оптимальной траектории (x, y) , которые бы за все время достигали бы терминального условия (4):

$$t_f \rightarrow \min. \quad (6)$$

Так как движение равномерное, то задача (1) - (6) сводится к задаче поиска оптимального по расстоянию пути из точки (x_0, y_0) в точку (x_{tf}, y_{tf}) . Не ограничивая общности, в проводимых экспериментах будем рассматривать кинематическую модель мобильного робота как материальную точку с координатами $(x, y) \in \mathbb{R}^2$. Данное допущение оправдано тем, что геометрические размеры робота (в частности, габарит L) могут быть учтены путем соответствующего расширения зон препятствий на величину $\frac{L}{2}$ во все стороны. Формально это эквивалентно преобразованию исходных ограничений. Такое преобразование гарантирует, что при движении точечного объекта по траектории $(x(t), y(t))$, реальный робот конечных размеров не будет пересекать исходные препятствия. Данный подход является стандартным в задачах планирования траекторий и позволяет существенно упростить вычислительные процедуры без потери общности.

Геометрия шасси Аккермана Система (1)-(3) описывает принцип работы шасси с геометрией Аккермана [4]. Геометрия шасси Аккермана — это конструктивная схема рулевого управления, используемая преимущественно в автомобилях и мобильных роботах для повышения маневренности и устойчивости при поворотах.

Основная цель геометрии Аккермана состоит в том, чтобы избежать бокового скольжения шин при движении по кривой.

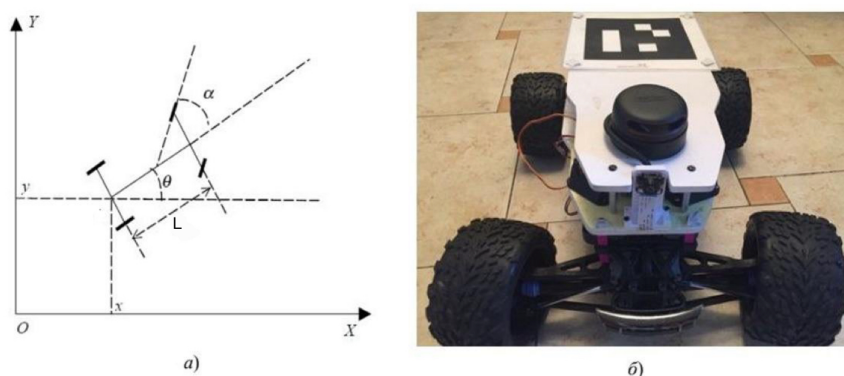


Рис. 1: Кинематическая схема шасси Аккермана управления движением мобильного робота: α — угол поворота руля; θ — угол ориентации мобильного робота относительно оси x ; L — расстояние между передней и задней осями робота.

Причины выбора геометрии Аккермана для мобильного робота: устойчивость при поворотах; эффективность управления: улучшенное сцепление и контроль над движением делает данный тип шасси предпочтительным для динамичного вождения и сложного маневрирования; распространенность и простота.

Выбор шасси геометрии Аккермана диктуется требованиями к маневренности, устойчивости и эффективности управления мобильными роботами в условиях ограниченного пространства и в сценариях, требующих высокой точности выполнения маневров.

Решение задачи с помощью принципа максимума Понтрягина Принцип максимума Понтрягина позволяет определить оптимальные стратегии управления для динамических систем, описываемых обыкновенными дифференциальными уравнениями. Для достижения определенной цели необходимо выбрать управление, которое максимизирует (или минимизирует) гамильтониан системы в каждом моменте времени.

Применение классического принципа максимума к задаче с геометрией Аккермана приведено на рис. 2.

Принцип максимума Понтрягина в рамках стандартной формулировки не учитывает физические ограничения на управление, такие как максимальная скорость и радиус поворота. Например, траектории, предполагающие мгновенные повороты или резкие изменения направления, могут быть невозмож-

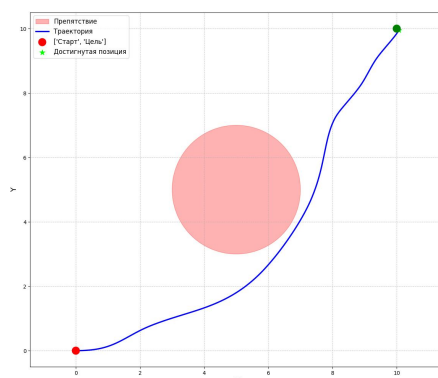


Рис. 2: Траектория движения методом принципа максимума Понтрягина.

ны для выполнения роботом из-за инерции и физических ограничений привода.

Решение задачи методом A-star A-star — это один из самых широко используемых и эффективных алгоритмов поиска кратчайшего пути с минимальными затратами от начальной точки до целевой точки в среде со стационарными препятствиями. Алгоритм A-star представляет собой усовершенствованную версию алгоритма Дейкстры. Особенностью A-star является его способность эффективно обрабатывать большие поисковые пространства, используя эвристическую функцию, которая позволяет предсказывать вероятность того, что узел находится на пути к цели.

Применение метода A-star к задаче с геометрией Аккермана приведено на рис. 3.

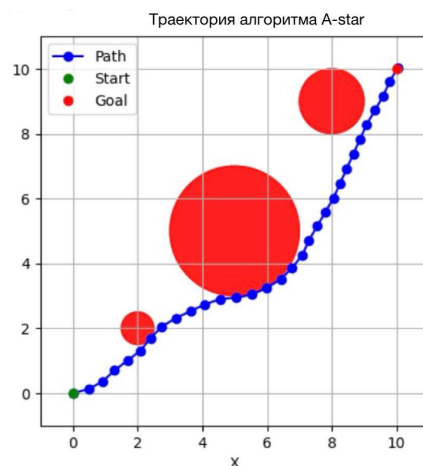


Рис. 3: Пример (метод A-star). Начальная точка $(x, y, \theta) : (0, 0, 0)$. Целевая точка $(x, y) : (10, 10)$. $u_1 : 0.5$. Длина пути: 15.00.

В ходе проведенных экспериментов по применению алгоритма A-star для планирования траектории мобильного робота были выявлены ключевые закономерности, связанные с влиянием параметров управления и дискретизации пространства на качество найденного пути. Наиболее существенные проблемы связаны с дискретизацией пространства состояний и управляющих воздействий.

Решение задачи методом последовательного квадратичного программирования (ПКП) Дискретизируем задачу (1)-(6) при помощи метода Эйлера, который позволяет получить систему разностных уравнений, описывающих состояние системы в дискретном времени. Основной идеей метода последовательного квадратичного программирования [5] является последовательное решение задач квадратичного программирования, аппроксимирующих данную задачу оптимизации.

Применение метода ПКП к задаче с геометрией Аккермана приведено на рис. 4.

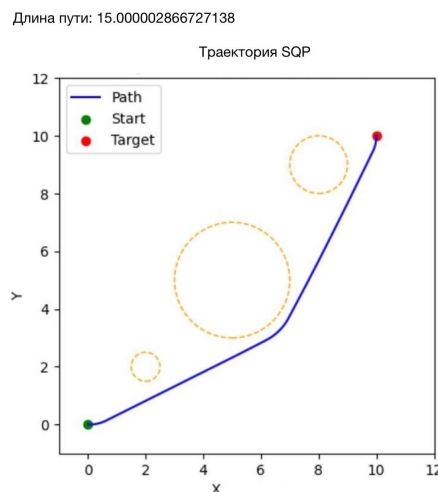


Рис. 4: Пример (метод ПКП). Начальная точка $(x, y, \theta) : (0, 0, 0)$. $u_{2\max} = \frac{\pi}{4}$. Целевая точка $(x, y) : (10, 10)$. $u_1 : 0,5$. $\varepsilon = 0,001$. $\Delta t = 0,1$. Длина пути: 15,000002866727138.

В ходе исследования задачи нахождения оптимальной траектории движения мобильного робота методом ПКП была проведена серия экспериментов, в рамках которых анализировалась зависимость общей длины пути и количества шагов дискретизации N от выбора величины временного шага Δt .

Заключение По итогам выполнения работы можно сделать следующие выводы.

Практическое применение принципа максимума Понтрягина выявило ряд ограничений, связанных с резкими изменениями управления и высокой вычислительной сложностью. Эти особенности делают данный метод неприменимым в реальных условиях работы мобильных роботов.

Модифицированный алгоритм A-star продемонстрировал свою эффективность при поиске субоптимальных траекторий. Разработанная адаптация алгоритма позволила учесть кинематические особенности колесного робота, однако качество всех полученных решений оказалось существенно зависимым от параметров дискретизации. Алгоритм показал себя как хорошее решение для задач, где важна скорость работы и допустимы приближенные решения.

Метод последовательного квадратичного программирования проявил себя как наиболее точный инструмент для построения оптимальных траекто-

рий. Несмотря на высокие вычислительные затраты, этот метод обеспечивает плавность управления и высокую точность попадания в целевую область, что делает его предпочтительным выбором для задач, требующих точного позиционирования.

Применение метода ПКП продемонстрировало высокую точность в поиске оптимальной траектории в непрерывной области, что позволяет учитывать сложные ограничения и достигать плавности движения, которой может быть сложно добиться с использованием дискретных методов, таких как A-star.

Полученные результаты показали, что оба метода (A-star и ПКП) эффективно справляются с поставленной задачей, находя траектории, минимизирующие длину пути, что делает возможным их применение в реальных системах автономной навигации различных мобильных платформ — от складских роботов до беспилотных транспортных средств.

Таким образом, основная цель — построение модели управления мобильным роботом при наличии фазовых ограничений — выполнена полностью. Вычислительный эксперимент показал работоспособность данной модели.

Дальнейшие исследования следует сосредоточить на преодолении выявленных ограничений. Особый интерес представляет разработка адаптивных алгоритмов управления, способных учитывать не только статические, но и динамические препятствия. Это потребует интеграции методов прогнозирования движения объектов и реактивного управления. Также введение управления скоростью как дополнительного параметра оптимизации позволит минимизировать энергопотребление и повысить плавность движения.

ЛИТЕРАТУРА

- [1] Понтрягин Л. С. Принцип максимума в оптимальном управлении. М.: Наука, 1989.
- [2] Hart P. E., Nilsson N. J., Raphael B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths // IEEE Transactions on Systems Science and Cybernetics. 1968. V. 4, № 2. P. 100–107.
- [3] Dijkstra E. W. A note on two problems in connexion with graphs // Numerische Mathematik. 1959. V. 1, № 1. P. 269–271.
- [4] Rajamani R. Vehicle Dynamics and Control. Springer, 2nd ed, 2012.
- [5] Сухарев А. Г., Тимохов А. В., Фёдоров В. В. Курс методов оптимизации. М.: Физматлит, 2005.

Метод определения оксидного состава металла по данным неизотермического газового анализа

Работа удостоена диплома 3 степени

Токарь Дмитрий Сергеевич

Кафедра математической физики

e-mail: dmitriy.s.tokar@gmail.com

Научный руководитель — д.ф.-м.н. проф. Крылов Андрей Серджевич

Актуальность задачи Для обеспечения высокого уровня качества металлических материалов необходимо осуществлять строгий контроль неметаллических включений, которые существенно влияют на механические свойства и износостойкость конечного продукта [1].

Одним из перспективных аналитических методов для оценки оксидного состава металлов является фракционный газовый анализ (ФГА) [2]. Этот метод основан на измерении скорости выделения монооксида углерода (СО) при термическом воздействии на образец в инертной среде, что приводит к восстановлению оксидов углеродом.

Анализ зависимости скорости выделения СО от температуры позволяет охарактеризовать типы и количественное содержание оксидных включений в исследуемом материале. Однако для корректной интерпретации полученных данных требуется разработка и применение специализированных математических моделей и программных алгоритмов.

Существующие методы [3][4] имеют определенные ограничения, связанные со сложностью интерпретации результатов и низкой производительностью процесса анализа. В связи с этим актуальной задачей является разработка новых методов обработки данных фракционного газового анализа, направленных на повышение интерпретируемости и эффективности метода.

Математическая модель Для описания процесса выделения СО из каждого типа оксидных включений используется следующее дифференциальное уравнение, соответствующее модели выделения кислорода из всего объема оксида:

$$\frac{dF(t)}{dt} = F(t) \left(\frac{df(t)}{dt} - e^{f(t)} \right),$$

где $f(t) = K - \frac{E}{T(t)}$, K и E — параметры, зависящие от типа оксида, $T(t)$ — зависимость температуры от времени. Решение задачи Коши приводит к аналитическому выражению функции $F(t)$, которая описывает скорость выделения СО из конкретного оксида. Для определения содержания оксидов в образце кривая, полученная в ходе анализа, аппроксимируется суммой кривых, соответствующих отдельным оксидам.

Предложенный метод Для разделения суммарной кривой выделения окиси углерода на отдельные компоненты, соответствующие различным типам оксидных фаз, предложено использовать численную методику на основе градиентного спуска с применением алгоритма RMSProp [5], реализованного

в сочетании с автоматическим дифференцированием. Подход заключается в параметризации индивидуальных пиков выделения СО с помощью аналитических функций, зависящих от физически значимых параметров, что позволяет учитывать как форму каждого пика, так и их взаимное расположение на температурной шкале. Для каждого типа включений в модели оптимизируются следующие параметры: V_{max} — максимальная скорость выделения СО, характеризующая амплитуду соответствующего пика; T^{beg} — начальная температура начала выделения, определяющая положение переднего фронта пика; T^{max} — температура, при которой достигается максимальная скорость выделения, то есть положение вершины пика на оси абсцисс; E — термодинамический параметр, отвечающий за крутизну и ширину пика. Помимо индивидуальных параметров для каждого компонента, вводится общий параметр T^{global} — глобальный температурный сдвиг, который компенсирует систематическое отклонение всей кривой модели относительно экспериментальных данных и может интерпретироваться как поправка, связанная с неточностью измерения температуры или задержками в регистрации сигнала.

Использование метода RMSProp обеспечивает стабильную и эффективную сходимость процесса оптимизации даже в условиях сложного рельефа целевой функции, поскольку данный метод адаптивно нормирует величину шага по каждому из параметров на основе скользящего среднего квадратов градиентов, что особенно важно при наличии сильно различающихся масштабов параметров и наличия участков с малыми градиентами. Автоматическое дифференцирование позволяет точно и быстро вычислять градиенты целевой функции по всем параметрам без необходимости их ручного вывода или применения численных приближений, что значительно повышает точность результатов. Благодаря использованию физически интерпретируемых параметров, предложенный метод не только обеспечивает высокое качество аппроксимации суммарной кривой, но и позволяет извлекать содержательные выводы о количественном и качественном составе оксидов, присутствующих в образце.

Функция потерь Функция потерь $\mathcal{L}$ представляет собой взвешенную сумму четырёх компонент:

- основная невязка $\mathcal{L}_{ref} = \frac{1}{N_{grid}} \sum_{j=1}^{N_{grid}} \left| F(t_j) - \sum_{i=1}^{N_{oxides}} F^i(t_j) \right|$;
- термодинамический штраф $\mathcal{L}_{beg}^i = \frac{1}{N_{grid}} \sum_{j=1}^{N_{grid}} \frac{F^i(t_j)}{1 + e^{t_j - T_{beg}^i}}$;
- регуляризация сдвига $\mathcal{L}_{shift} = \sum_{i=1}^{N_{oxides}} |\delta T_{beg}^i|$;
- парсимониальная регуляризация $\mathcal{L}_{use} = \sum_{i=1}^{N_{oxides}} w_{use}^i \cdot V_{max}^i$.

Итоговая функция потерь имеет следующий вид:

$$\mathcal{L} = w_{ref} \mathcal{L}_{ref} + w_{beg} \mathcal{L}_{beg} + w_{shift} \mathcal{L}_{shift} + \mathcal{L}_{use}.$$

Особенности алгоритма Алгоритм сочетает использование RMSProp, позволяющего нормализовать шаги обновления параметров за счёт экспоненциально затухающего среднего квадратов градиентов, что особенно эффективно в условиях наличия аномально высоких значений градиента, и метода

момента, обеспечивающего инерционное движение по пространству параметров и способствующего преодолению локальных минимумов. Дополнительно введён механизм отсева незначимых компонентов, основанный на динамическом сравнении текущего значения V_{max} с адаптивным порогом $\hat{V}_{min}$, который позволяет повысить скорость обработки за счёт исключения малозначимых включений и дополнительно снижает вероятность попадания в области с высоким значением градиента.

Реализация и результаты Работа реализована с использованием библиотеки PyTorch для автоматического дифференцирования и оптимизации. Разработан графический интерфейс, позволяющий оператору гибко настраивать параметры анализа, управлять приоритетами и визуализировать результаты. Метод протестирован на реальных данных анализа образцов стали марки ШХ-15. Полученные результаты показали хорошее соответствие с экспертной оценкой содержания оксидов.

Заключение Предложенный метод обеспечивает: автоматизированный анализ данных ФГА, интерпретируемость результатов, учёт физических ограничений, высокую производительность и масштабируемость. Метод легко обобщается на большее число включений а также позволяет задавать дополнительные ограничения внося дополнительные слагаемые в функцию потерь. Тестирование метода показало его применимость на практике.

ЛИТЕРАТУРА

- [1] Григорович К. В., Красовский П. В., Трушникова П. В. Анализ неметаллических включений — основа контроля качества стали и сплавов // Аналитика и контроль. 2002. С. 133–42.
- [2] Григорович К. В. Фракционный газовый анализ—новое направление в контроле качества материалов // Аналитика и контроль. 2000. № 3. С. 244–251.
- [3] Обработка и интерпретация результатов фракционного газового анализа / К. В. Григорович, П. В. Красовский, С. А. Исаков, А. А. Горохов, А. С. Крылов // Заводская лаборатория. Диагностика материалов. 2002. Т. 68, № 9. С. 3–9.
- [4] Neural network method for determining the oxide composition of a metal by the non-isothermal gas evolution curve / D. D. Knorre, A. S. Krylov, K. V. Grigorovich, Y. K. Feldman // Computational Mathematics and Modeling. 2025. P. 1–14.
- [5] Cauchy A. Méthode générale pour la résolution des systèmes d'équations simultanées // C. R. Acad. Sci. Paris. 1847. Vol. 25. P. 536–538.

Модели машинного обучения для обнаружения функциональных паттернов в многомерных временных рядах

Федоров Артем Максимович

Кафедра математических методов прогнозирования

e-mail: fedorov.am@list.ru

Научный руководитель — к.ф.-м.н. доц. Майсурадзе Арчил Ивериевич

Научный консультант — д.ф.-м.н. проф. Волканов Дмитрий Юрьевич

Аннотация Настоящее исследование направлено на разработку нового по своему принципу действия *масштабируемого, интерпретируемого и адаптируемого* инструмента поиска мотивов в многомерных временных рядах, эффективно справляющегося с ростом объемов обучающих данных. Новый метод применяет байесовский вариационный вывод, предлагающий в перспективе гибкий механизм обработки временных рядов разной природы с учетом их особенностей, посредством ядерных методов и вероятностной регуляризации числа находимых паттернов, что будет эффективен как в офлайн анализе крупных архивов, так и в онлайн мониторинге.

Введение Современные задачи прикладных областей, таких как биомедицина, экономика, инженерия, все чаще опираются на выявление *устойчивых локальных закономерностей* во временных рядах, отражающих характерные состояния исследуемых систем, благодаря чему используются в качестве ключевых признаков для прогнозирования, диагностики аномалий и интерпретации больших корпусов временных рядов. Хотя каждое предметное сообщество формулирует собственное определение паттерна и подбирает специализированные методы его поиска, математика и машинное обучение, путем обобщения и формализации, сводят все такие постановки к общей задаче *motif detection*: необходимо (i) восстановить словарь паттернов, присутствующих в выборке, и (ii) локализовать во временных рядах их конкретные реализации — мотивы.

На практике наиболее распространённые алгоритмы motif detection, такие как Matrix Profile[2] и SAX[1], дают приемлемые результаты лишь для коротких одномерных сигналов. При росте объёма выборки, количества каналов и длины рядов они сталкиваются с комбинаторным взрывом сложности, ухудшением точности и потерей интерпретируемости. Между тем прикладные эксперты по-прежнему требуют качественного, прозрачного и быстрого поиска паттернов. Цель настоящего исследования — предложить *масштабируемый, интерпретируемый и дисциплинарно-универсальный детектор*, который обучается без разметки, автоматически адаптирует число паттернов и сохраняет статистическую устойчивость на современных высокоразмерных данных, тем самым устраняя ограничения существующих методов.

Постановка задачи Рассматривается неразмеченная выборка временных рядов $\mathcal{S} = S_{i=1}^N$, где $S_i \in \mathbb{R}^{n_i \times d}$ — многомерный временной ряд, пред-

ставимый в виде конкатенации сегментов $S_i = \bigoplus_{j=1}^{k_i} s_i^j$. Далее основным объектом изучения будет являться именно сегмент временного ряда.

Определение. *Функциональный паттерн — класс эквивалентности сегментов, обладающих общей латентной «структурой», объекты которого имеют устойчивую закономерность возникновения в данных.*

Определение. *Мотив — единичная реализация функционального паттерна в данных, в частности сегмент временного ряда.*

Конечная цель метода motif detection — построить систему классов (обнаружить все функциональные паттерны) и найти в датасете все вхождения элементов в данные классы. Рассмотрим генеративную природу генерации $\mathcal{S}$ (формула 1):

$$p(\mathcal{S}, T, \mathcal{Z}, \mathcal{P}) = p(\mathcal{S}|T, \mathcal{P}) p(T|\mathcal{P}) p(\mathcal{P}) = \prod_{i=1}^N \prod_{j=1}^{k_i} \left\{ \prod_{\ell=1}^{\infty} p(s_i^j | \mu_\ell)^{z_{i,j}^\ell \pi_k} \right\} p(T_i|\mathcal{P}) p(\mathcal{P}) \quad (1)$$

- $\mathcal{S}$ — выборка временных рядов единой природы.
- T — разметки для каждого временного ряда $S_i \in \mathcal{S}$, задающая границы сегментов s_i^j : $T_i = \{0 = t_i^0 \leq t_i^1 \leq \dots \leq t_i^{k_i} = n_i\}$.
- $\mathcal{Z} = \{z_{i,j}^\ell\}$ — индикаторы принадлежности каждого сегмента определенному классу эквивалентности.
- $\mathcal{P} = \{\mu_\ell\}$ — система классов эквивалентности, представимая множеством прообразов сегментов в аналитическом представлении.

Обучение сводится к максимизации маргинального правдоподобия $\mathcal{S}$ (формула 2):

$$\log p(\mathcal{S}) = \log \int p(\mathcal{S}, T, \mathcal{Z}, \mathcal{P}) dT d\mathcal{Z} d\mathcal{P} \rightarrow \max_{p(T, \mathcal{Z}, \mathcal{P})} \quad (2)$$

В результате получаются оценки словаря $\hat{\mathcal{P}}$ ассоциируемые с системой паттернов и апостериорная вероятность $p(T, \mathcal{Z} | \mathcal{S}, \hat{\mathcal{P}})$, позволяющие детектировать мотивы на новых данных.

Компоненты решения Следующим шагом является доопределение компонент вероятностной модели, отвечающих условиям задачи motif detection.

Правдоподобие сегментов при выбранной разметке и классах $\mathcal{P}$ задается изотропным гауссовским распределением $p(s | \mu_\ell) = \mathcal{N}(\varphi(s) | \mu_\ell, I)$, где μ_ℓ — вектор-представитель латентного паттерна. Для этого построим оператор φ , переводящий сегменты различной продолжительности в единое векторное пространство, обеспечивающий инвариантность моделируемых классов эквивалентности к сдвиг-масштабным трансформациям и растяжению по времени (частым требованием со стороны специалистов прикладных областей)

(формула 3). В качестве такого оператора φ будем использовать косинусное дискретное разложение с оставлением первых k старших членов.

$$\forall \alpha, \beta \in \mathbb{R}^+, \forall c \in \mathbb{R} : s(t) \in P \in \mathcal{P} \Rightarrow \alpha \cdot s(\gamma t) + c \in P \quad (3)$$

Число встречаемых паттернов в $\mathcal{S}$ *a priori* неизвестно, мы можем лишь регуляризовать модель через наши представления о «скорости» их появления от объема и вариативности данных. Этого добивается бесконечная смесь Дирихле [3]: на веса π_ℓ накладывается процесс Дирихле с концентрационным параметром α , реализованный в stick-breaking форме (формула 4)

$$\pi_\ell = v_\ell \prod_{m < \ell} (1 - v_m), \quad v_\ell \sim \text{Beta}(1, \alpha), \quad \mu_\ell \sim G_0, \quad (4)$$

$$p(\mathcal{S}, T, \mathcal{Z}, \mathcal{P}) = \prod_{i,j}^{N,k_i} \left[\prod_{\ell}^{\infty} \mathcal{N}(\varphi(s_{ij}) | \mu_\ell, I)^{z_{ij}^\ell \pi_\ell} \right] p(T_i | \mathcal{P}) \prod_{\ell}^{\infty} G_0(\mu_\ell) \text{Beta}(v_\ell | 1, \alpha).$$

Приближение распределения $p(T | \mathcal{P})$ производится посредством метода Монте-Карло, позволяющее добиться линейной алгоритмической сложности $O(|\mathcal{S}|)$ от мощности выборки $\mathcal{S}$ на итерации.

Получаемая задача оптимизации (формула 2) решается вариационным ЕМ (*Expectation Maximization*) алгоритмом. Ввиду отсутствия сопряженности апостериорных распределений на латентные случайные величины, применяется метод Mean-Field approximation [4] для факторизации распределения $p(\mathcal{Z}, \mathcal{P}, v | \mathcal{S}) \approx q(\mathcal{P})q(\mathcal{Z})q(v)$.

Валидация метода Отсутствие данных, обладающих эталонной разметкой, что является особенностью задачи motif detection, делает невозможной прямую оценку метода классическими метриками (precision, recall, F_1 -score). Корректность алгоритма проверяется косвенно: требуется показать, что (i) он извлекает из реальных данных паттерны, признанные предметными экспертами, и (ii) делает это статистически значимым образом.

Эксперимент 1 Первый эксперимент направлен на проверку способности предлагаемого алгоритма находить уже известные аналитикам функциональный паттерны в данных. Работа проводилась на историческом наборе дневных цен открытия более семи тысяч американских акций и ETF за пятьдесят четыре года. Считалось, локальные закономерности развития цен укладываются в интервал от 30 до 150 дней, что послужило ограничением длинны искомых мотивов.

Уже на первых итерациях вариационный функционал вышел на плато, среди элементов $\mathcal{P}$ осталось около ста существенно отличимых друг от друга прототипов. Визуальная экспертиза ста наиболее вероятных прототипов показала, что почти три четверти из них совпадают по форме с классическими фигурами технического анализа — «голова-плечи», «флагами», двойными вершинами и днами. Оставшиеся мотивы оказались редко встречающимися

боковыми консолидациями и всплесками волатильности, что также ожидаемо для реального рынка. Тем самым продемонстрировано, что модель без учителя выделяет содержательные закономерности и ранжирует их по статистической значимости.

Эксперимент 2 Во второй серии экспериментов посредством генерации синтетических данных проводилась проверка способности алгоритма находить явно заложенные в данные мотивы, а так же исследовалась эффективность работы алгоритма от объема обучающих данных. Использовались двухканальные сигналы случайной длины со вставками заранее определенного паттерна: короткий всплеск в первом канале, за которым с некоторым отступом по времени следовал всплеск во втором. Каждый паттерн подлежал сдвиг-масштабной трансформации и растяжению по времени.

С ростом мощности обучающей выборки наблюдается монотонный рост метрик качества и уверенности модели в своих ответах. Метод быстро выходит на плато и достигает полноты в размере 0.84, что говорит о состоятельности алгоритма и соответствии находимых паттернов условиям инвариантности.

Эксперимент 3 Третий эксперимент направлен на оценку конкурентоспособности предлагаемого детектора относительно общеиспользуемых алгоритмов Matrix Profile [2] и SAX [1] в прикладной задаче клинической классификации. Используется публичный набор *HMS Harmful Brain Activity* [5], содержащий 16-канальные фрагменты ЭЭГ, размеченные бинарной меткой *seizure/non-seizure*.

Без потери общности выборка разбивается на обучающую часть детекторов $\mathcal{D}_{\text{unsup}}$ (60 % сегментов) и независимый тест $\mathcal{D}_{\text{test}}$. После обучения каждая модель, работая в режиме инференса, производила разметку $\mathcal{D}_{\text{test}}$. После чего каждый временной ряд из $\mathcal{D}_{\text{test}}$ заменялся бинарным вектором индикаторов, присутствия того или иного паттерна в нем (модель Bag-of-Words). Далее на полученном датасете обучалась Логистическая регрессия. Сравнение проводилось на основе метрик качества Accuracy и F_1 score, достигаемых линейной моделью на отложенных данных, а так же на основе времени, затрачиваемого на обучение детектора мотива.

Метод	Accuracy	F_1 -score	Время обуч.
Matrix Profile	0.83	0.48	3.6 ч
SAX	0.67	0.45	~1 ч
<i>Ours</i>	0.61	0.54	1.3 ч

Табл. 1: Метрики качества методов motif detection для downstream задачи

Предлагаемый метод выигрывает остальные детекторы по метрике F_1 score, благодаря существенно лучшей полноте по миноритарному классу «seizure», при этом затрачивая сопоставимое время обучения с самым быстрым методом SAX [1].

Выводы и перспективы Проведённое исследование показало, что вариационный байесовский детектор, основанный на приведении сегментов к единому латентному представлению посредством дискретного косинусного разложения и имплементирующий процесс Дирихле для отбора функциональных паттернов, автоматически извлекает из многомерных рядов статистически значимые мотивы, позволяющие эффективно решать прикладные задачи. Метод показал свою способность работать с данными основных прикладных областей, таких как экономика и биомедицина, находить существующие паттерны в данных, сохранять интерпретируемость находимых им объектов для дальнейшей работы аналитиков, а так же свою конкурентоспособность относительно основных современных методов motif detection, выигрывая последние в эффективности по времени и по метрикам качества.

Дальнейшее развитие видится в переходе к on-line вариационному выводу, адаптирующему апостериорное распределение на латентные переменные при нестационарной природе сигналов. Обобщение отображения φ на вейвлет- и manifold-базисы расширит спектр детектируемых паттернов, а введение информативной меры G_0 позволит изменять поведение алгоритма исходя из априорных знаний специалистов прикладных областей.

ЛИТЕРАТУРА

- [1] Kevin Feasel. Symbolic aggregate approximation (sax). In Finding Ghosts in Your Data: Anomaly Detection Techniques with Examples in Python // E. Springer, 2022, P. 293–309.
- [2] Hoang Anh Dau and Eamonn Keogh. Matrix profile v: A generic technique to incorporate domain knowledge into motif discovery // In Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017, P. 125–134.
- [3] Yee Whye Teh et al. Dirichlet process. // Encyclopedia of machine learning, Vol. 1063, 2010, P. 280–287.
- [4] Toshiyuki Tanaka. A theory of mean field approximation. // Advances in Neural Information Processing Systems, Vol. 11, 1998.
- [5] Jin Jing et al. HMS - Harmful Brain Activity Classification // Kaggle competition, 2024.

Локальный разрывный метод Галёркина для решения эволюционных систем уравнений в частных производных

Работа удостоена диплома 1 степени

Финенко Никита Алексеевич

Кафедра вычислительных методов

e-mail: s02210579@cs.msu.ru

Научный руководитель — д.ф.-м.н. доц. Савенков Евгений Борисович

В работе исследуется и применяется локальный разрывный метод Галёркина (LDG), впервые предложенный Cockburn и Shu [1], для численного решения одномерных эволюционных систем уравнений в частных производных типа конвекции-диффузии. Метод LDG является естественным обобщением разрывного метода Галёркина (DG), используемого для гиперболических систем уравнений, на задачи, содержащие диффузионные члены второго и более высоких порядков. Ключевая идея метода LDG заключается в преобразовании исходного уравнения в частных производных высокого порядка в эквивалентную систему уравнений в частных производных первого порядка путем введения вспомогательных переменных, аппроксимирующих градиенты искоемых функций. При таком переходе в полученной системе только одно уравнение сохраняет производную по времени, в то время как остальные уравнения, связывающие вспомогательные переменные с производными основных, становятся стационарными. Эти стационарные уравнения могут быть разрешены локально в каждой пространственной ячейке (конечном элементе) расчетной сетки.

Отличительной особенностью разрывных методов Галёркина, включая LDG, является использование базисных функций, которые аппроксимируют решение внутри каждого элемента и могут иметь разрывы на границах между элементами. Это свойство позволяет численному решению воспроизводить резкие изменения или даже разрывы в решении. В данной работе в качестве системы базисных функций на каждом элементе используется ортогональная система полиномов Лежандра произвольного выбираемого порядка N .

Разрывные методы Галёркина, и LDG в частности, обладают рядом преимуществ, делающих их привлекательными для широкого круга задач. Во-первых, они предоставляют единообразный подход для построения аппроксимаций как для гиперболических задач первого порядка, так и для уравнений второго и более высоких порядков. Во-вторых, метод характеризуется высокой локальностью вычислений: операции на каждом элементе зависят только от соседних элементов, что не требует обращения больших разреженных матриц и делает его особенно эффективным для параллельных вычислений. В-третьих, LDG хорошо приспособлен для процедур адаптации сетки, позволяя использовать нерегулярные сетки с локальным сгущением (h -адаптация) и различный порядок базисных функций в различных конечных элементах

(p -адаптация), что дает возможность концентрировать вычислительные ресурсы в областях с наиболее сложным поведением решения. Шаблон метода при этом остается локальным и не зависит от порядка используемых базисных функций.

Для корректной постановки задачи и обеспечения устойчивости схемы на границах между элементами вводятся численные потоки. Для конвективных членов используется поток Лакса-Фридрихса, а для диффузионных членов — центральные разности (среднее арифметическое значений на границе). Интегрирование по времени полученной системы обыкновенных дифференциальных уравнений для коэффициентов разложения решения по базисным функциям осуществляется с помощью явного метода Рунге-Кутты третьего порядка, обладающего свойством сохранения полной вариации (TVD RK3), что способствует подавлению нефизичных осцилляций вблизи разрывов.

Необходимо отметить, что, несмотря на перечисленные преимущества, метод LDG является вычислительно затратным. Это связано с увеличением числа неизвестных за счет введения вспомогательных переменных для аппроксимации градиентов, каждая из которых аппроксимируется независимо.

В рамках работы метод LDG был последовательно применен к ряду модельных задач. Первоначально рассматривалось скалярное уравнение конвекции-диффузии:

$$\partial_t u + \partial_x (f(u) - a(u) \partial_x u) = 0.$$

Для этого уравнения были проведены численные эксперименты как с гладкими начальными данными, где сравнивались результаты с точным аналитическим решением для линейного случая $f(u) = u, a(u) = 1$, так и для случая формирования крутого фронта в уравнении Бюргерса с вязкостью при $f(u) = u^2/2$ и малой вязкости $a(u) = 0.01$. Результаты подтвердили корректность реализации и продемонстрировали ожидаемое повышение точности при увеличении порядка аппроксимации N полиномов Лежандра (от $N = 1$ до $N = 3$). Также было проведено исследование зависимости максимально допустимого шага по времени Δt , при котором схема остается устойчивой, от пространственного шага Δx . Была получена степенная зависимость $\Delta t \approx C \cdot (\Delta x)^{1.98}$, что хорошо согласуется с теоретическими оценками $\Delta t \sim (\Delta x)^2$ для явных схем решения параболических уравнений.

Далее метод LDG был обобщен и применен для решения одномерной системы уравнений конвекции-диффузии общего вида.

$$\partial_t u + \partial_x F_e(u) - \partial_x F_v(u, \partial_x u) = 0.$$

Была рассмотрена система уравнений Навье-Стокса для одномерного течения сжимаемого, вязкого и теплопроводного газа [2]. Эта система, включающая уравнения сохранения массы, импульса и полной энергии, была записана в консервативной форме, и для нее были выведены соответствующие конвективные и вязкие потоки в терминах консервативных переменных. Численные расчеты проводились для задачи о взаимодействии разрывов, результаты которых (профили плотности и скорости) сравнивались с эталонными данными.

Сравнение показало, что повышение порядка аппроксимации полиномов Лежандра ($N = 1, 2, 3$) позволяет существенно точнее воспроизводить структуру течения. В частности, схема первого порядка ($N = 1$) характеризовалась заметным «размыванием» решения из-за численной диссипации, в то время как схемы более высоких порядков обеспечивали лучшее разрешение.

Кроме того, метод LDG был применен к квазигазодинамической (КГД) системе уравнений [3], также в одномерной постановке. КГД система представляет собой модификацию уравнений Навье-Стокса, в которую включены малые физически обоснованные слагаемые диссипативного характера, зависящие от параметра релаксации τ . В работе были детально описаны потоковые члены КГД системы. Численный эксперимент проводился для задачи о распаде произвольного разрыва с начальными данными, соответствующими тестам из работы Т.Г. Елизаровой [3]. Полученные с использованием базисных функций первого порядка ($N = 1$) профили плотности и скорости продемонстрировали плавное сглаживание разрывов без образования осцилляций и ступеней.

Программная реализация метода LDG была выполнена на языке Python. Программа позволяет использовать базисные функции (полиномы Лежандра) произвольного порядка аппроксимации и различные численные потоки.

В результате проведенного исследования было показано, что локальный разрывный метод Галёркина является гибким инструментом для решения широкого круга одномерных эволюционных задач. Продемонстрированы преимущества использования базисных функций высоких порядков для повышения точности. Исследована устойчивость метода. Полученные результаты подтверждают перспективность метода LDG для решения рассматриваемого класса задач, включая системы уравнений газовой динамики.

ЛИТЕРАТУРА

- [1] Cockburn B., Shu C.-W. The Local Discontinuous Galerkin method for time-dependent convection-diffusion systems // SIAM Journal on Numerical Analysis. 1998. Т. 35, № 6. С. 2440–2463.
- [2] Bassi F., Rebay S. High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier–Stokes Equations // Journal of Computational Physics. 1997. Т. 131, № 2. С. 267–279.
- [3] Елизарова Т. Г. Квазигазодинамические уравнения и методы расчета вязких течений. М. : Научный Мир, 2007. 350 с.

Разработка системы управления бортовым оборудованием на отработочном комплексе

Работа удостоена диплома 2 степени

Фомин Дмитрий Сергеевич

Кафедра системного программирования

e-mail: fomindmitriyfoma@mail.ru

Научный руководитель — д.ф.-м.н. проф. Петренко Александр Константинович

Отработочный комплекс — это совокупность программных и аппаратных средств для отладки и тестирования программного обеспечения или аппаратуры.

Рассмотрим отработочный комплекс, который используется в институте системного программирования им. В.П. Иванникова Российской академии наук для отладки операционных систем реального времени. Он включает в свою аппаратную часть бортовое оборудование, для которого разрабатывается операционная система реального времени. Пользователи взаимодействуют с оборудованием с помощью сервера, который является персональным компьютером под управлением операционной системы общего назначения.

Работа посвящена проектированию программного обеспечения, которое запускается на сервере и отвечает за управление отработочным комплексом. Система управления отработочным комплексом предоставляет пользовательский интерфейс для удаленного управления бортовым оборудованием: запуск аппаратуры с помощью подачи электропитания, получение отладочного вывода от бортового оборудования, и т. д.

Разные отработочные комплексы имеют разные сценарии использования. В нашем случае можно говорить о следующих особенностях использования аппаратных ресурсов:

- Несколько пользователей могут одновременно работать с разными экземплярами бортового оборудования отработочного комплекса.
- К аппаратуре отработочного комплекса осуществляется конкурентный доступ (несколько пользователей не могут осуществлять одновременную отладку с использованием одного экземпляра бортового оборудования).
- В аппаратную часть отработочного комплекса входит большое количество разных моделей оборудования (системные платы с процессорами архитектуры PowerPC, RISC-V, x86, и другие).
- Аппаратура отработочного комплекса подвержена сбоям.
- Аппаратная конфигурация отработочного комплекса периодически меняется (добавляется и удаляется бортовое оборудование).

Для управления выбранным отработочным комплексом долгое время использовались программа с открытым исходным кодом и программа, написанная специально для этого отработочного комплекса. Это решение имело ярко

выраженные недостатки, исправление которых было неоправданно сложной задачей.

В ходе исследования были рассмотрены отработочные комплексы, используемые в ракетно-космической промышленности [1, 2, 3], образовании [4, 5], а также в коммерческих кампаниях. Также было найдено программное обеспечение PlatformIO с открытым исходным кодом, часто используемое для создания отработочных комплексов. Однако не было найдено описания такой системы управления оборудованием, которая полностью подходила бы для управления выбранным отработочным комплексом.

Целью данной работы является разработка архитектуры системы управления бортовым оборудованием на отработочном комплексе с учетом особенностей его эксплуатации.

Работа является актуальной в связи с тем, что несколько организаций используют похожие отработочные комплексы, однако в открытых источниках не было найдено программное обеспечение, которое полностью отвечало бы требованиям предметной области и не содержало бы ярко выраженных недостатков.

Разработка системы управления отработочным комплексом состояла из следующих этапов:

- Формулирование требований к архитектуре и реализации системы управления отработочным комплексом.
- Проектирование и реализация системы управления отработочным комплексом в виде программного обеспечения.
- Апробация системы управления на выбранном отработочном комплексе.

Далее приводятся требования, которые были сформулированы на основе изучения предметной области, а также ключевые особенности архитектуры и реализации, благодаря которым разработанная система управления отработочным комплексом удовлетворяет поставленным требованиям.

Проксирование потока данных между бортовым оборудованием отработочного комплекса и пользователем. Пусть пользователь связывается с сервером по протоколу TCP, а бортовое оборудование с сервером — с помощью физических интерфейсов. Проксированием потока данных сервером называется передача данных между физическим портом и TCP-сокетом. Пользователь получает отладочный вывод аппаратной платформы от TCP-сокетов, что делает возможным удаленные отладку и тестирование.

Наличие интерфейса конкурентного доступа к бортовому оборудованию отработочного комплекса. Для обеспечения конкурентного доступа к бортовому оборудованию предлагается использовать интерфейс запросов (в реализации используются HTTP-запросы). Поддерживается как минимум два запроса: «захватить» и «освободить», каждый из которых содержит имя бортового оборудования. Перед началом отладки с помощью экземпляра бортового оборудования пользователь посылает запрос «захватить», а после окончания отладки — запрос «освободить». Можно провести

аналогию с массивом мьютексов, где мьютекс блокируется на все время взаимодействия с оборудованием.

Поддержка различных моделей бортового оборудования. На выбранном отработочном комплексе используется много разных моделей оборудования. Каждая модель имеет уникальный набор инструментов прошивки и отладки. Также могут отличаться сценарии использования аппаратуры. Система управления отработочным комплексом позволяет описывать пользовательские запросы, а также обработчики запросов в виде скриптов взаимодействия с оборудованием. Пользователю предлагается самостоятельно описать запросы и обработчики, подходящие для управления данной моделью оборудования.

Возможность задать аппаратную конфигурацию отработочного комплекса и изменить ее без перезапуска ПО. Для задания аппаратной конфигурации комплекса используется текстовый конфигурационный файл формата `yaml`, в котором содержится список доступного оборудования. При изменении аппаратной конфигурации отработочного комплекса требуется отредактировать конфигурационный файл и послать серверу запрос «пересканировать». Система управления бортовым оборудованием перечитывает конфигурационный файл и изменит свою программную конфигурацию, не прерывая работу большинства компонентов. В процессе обработки запроса сеанс работы пользователей с оборудованием не будет прерван.

Наличие средств мониторинга состояния комплекса. Предлагается использовать веб-интерфейс, на котором можно увидеть состояние отработочного комплекса:

- список заблокированных и свободных аппаратных платформ,
- время последней активности пользователя для данной аппаратной платформы,
- предупреждения об аппаратных сбоях, и т. д.

Кроме того, система управления отработочным комплексом ведет файлы журнала.

Открытость ПО для расширения для поддержки специфических сценариев использования. В процессе эксплуатации к системе управления отработочным комплексом предъявлялись новые требования, предугадать которые заранее было невозможно. В дальнейшем этот список требований может быть еще расширен. Эта проблема также решается с помощью HTTP-запросов, где каждое требование удовлетворяется введением нового запроса.

Утверждается, что такая архитектура ПО подходит для управления любым отработочным комплексом, имеющим описанные выше аппаратную конфигурацию и сценарий использования.

Результатом работы стала программа на языке программирования Python.

Разработанное ПО более года используется в институте системного программирования им. В. П. Иванникова Российской академии наук на указан-

ном отработочном комплексе. По сравнению с ранее использовавшимся решением удалось достигнуть следующих результатов:

- прекратились сбои со стороны системы управления отработочным комплексом (новое решение превосходит старое в надежности реализации),
- появился интерфейс мониторинга аппаратных сбоев со стороны аппаратуры отработочного комплекса,
- для операционных систем реального времени разработаны новые тестовые сценарии, использующие дополнительные возможности новой системы управления оборудованием.

Итоговая версия программы содержит 5000 строк кода на языке программирования Python (считая пустые строки и комментарии). Кроме того, были написаны тесты (1500 строк), имитирующие основные сценарии использования ПО и обеспечивающие покрытие 95% исходного кода программы, а также документация.

ЛИТЕРАТУРА

- [1] Пичкалев А. В. Создание автоматизированного мобильного испытательного комплекса для цеховых испытаний программно-управляемой радиоэлектронной аппаратуры // Решетневские чтения, 2009. С. 528–529.
- [2] Пичкалев А. В. Наземный отладочный комплекс бортовой радиоэлектронной аппаратуры // Решетневские чтения, 2010. С. 515–516.
- [3] Недорезов Д. А. Разработка и внедрение специального программного обеспечения для лабораторных отработочных комплексов испытаний радиоэлектронной аппаратуры // Решетневские чтения, 2011. С. 626–627.
- [4] Angel F. H., Ignacio E., Marisa L. V. A web-based environment providing remote access to FPGA platforms for teaching digital hardware design // IADIS International Conference e-Learning, 2008. P. 5
- [5] Ёхин М. Н., Степанов М. М. Организация многопользовательского удаленного доступа к распределенной гетерогенной системе лабораторного оборудования на основе схем программируемой логики для дистанционных практикумов по цифровой схемотехнике // Современные информационные технологии и ИТ-образование, 2017. С. 65–81.

Вопросно-ответная система для поддержки пользователей на базе открытой языковой модели

Ханин Андрей Дмитриевич

Кафедра алгоритмических языков

e-mail: s02210584n@cs.msu.ru

Научный руководитель — к.ф.-м.н., асс. Кузина Лариса Николаевна

В условиях цифровизации и стремительного роста объёмов внутренней информации компании сталкиваются с проблемой оперативной поддержки пользователей. Ручная обработка запросов приводит к снижению эффективности обслуживания. В данной работе представлена интеллектуальная система поддержки на основе извлекательно-генеративного подхода (RAG — Retrieval-Augmented Generation), позволяющего генерировать качественные ответы с использованием дистиллированных больших языковых моделей (LLM — Large Language Models) и найденных релевантных частей документной базы (чанков).

Целью исследования стала разработка и оптимизация RAG-системы. Для реализации использовался язык программирования Python и библиотеки из экосистемы HuggingFace. Проведён обзор существующих подходов:

- *Шаблонные* – просты в реализации, но ограничены по гибкости;
- *Трансформеры* – дают качественные ответы, но требуют высоких вычислительных ресурсов, особенно при переобучении;
- *RAG-системы* – сочетают преимущества поиска и генерации, позволяя работать с локальными документами.

Разработанная RAG-система коммуницирует с пользователем через сервис "Telegram" и включает две основные части [1]:

1. *Поисковую*, использующую BM25 и/или FAISS с эмбединговыми моделями (all-MiniLM-L6-v2, cointegrated/LaBSE-en-ru и multilingual-e5-large) с последующим объединением результатов через взвешенную сумму или среднее гармоническое для поиска наиболее релевантных вопросу чанков;
2. *Генеративную*, построенную на основе компактных LLM (gemma-2-2b, DeepSeek-R1-Distill-Qwen-1.5B, SmolLM2-1.7B и Qwen-1.5B) с квантированием Q6 и Q8 для оптимизации производительности.

Для проведения эксперимента была взята документная база данных с хакатона РОСАТОМа АтомикХак 2.0 [2], представляющая собой обезличенные документы по работе с системой 1С. Также в рамках хакатона определялся набор вопросов пользователей с ответами специалистов на базе данных документов. В рамках экспериментов набор вопросов-ответов был расширен для формирования обучающей и тестовой выборок.

Пример, вопроса пользователя и ответа специалиста:

Вопрос: «Кто такие дополнительные согласующие?»

Ответ: «Дополнительные согласующие — это пользователи, которые наряду с основным согласующим должны выполнить согласование текущего этапа. Этап будет пройден, когда его согласуют сначала все дополнительные согласующие, а затем основной согласующий.»

Для поиска наиболее оптимальных параметров при данной документной базе данных была разработана отдельная программа, осуществившая Grid Search отдельно для каждой части RAG-системы.

Метрики поисковой части:

- ROUGE-L Recall – оценивает полноту найденных релевантных чанков по отношению к эталонному, отражая долю информации из эталонного ответа, содержащуюся в них.
- ROUGE-L F1-score – оценивает точность и полноту найденных релевантных чанков по отношению к эталонному ответу, отражая общее качество совпадений.
- Load-time-normalized ROUGE-L F1-score – нормализует по времени деления на чанки и при необходимости индексации документной базы данных метрику ROUGE-L F1-score.

Метрики генеративной части:

- ROUGE-L Recall – оценивает полноту сгенерированного ответа по отношению к эталонному, отражая долю информации из эталонного ответа, содержащуюся в нем.
- ROUGE-L F1-score – оценивает точность и полноту сгенерированного ответа по отношению к эталонному ответу, отражая общее качество совпадений.
- BERTScore F1-score оценивает семантическое сходство между сгенерированным и эталонным ответами, учитывая синонимы и перефразирование.
- ROUGE-L-BERT F1-score – среднее гармоническое метрик ROUGE-L F1-score и BERTScore F1-score.
- Answer-time-normalized ROUGE-L-BERT F1 – нормализует по времени генерации ответа метрику ROUGE-L-BERT F1-score.

Результаты выбора наиболее оптимальных параметров:

- Подход поиска релевантных чанков – BM25, без FAISS с эмбединговыми моделями.
- Максимальное количество чанков в средней по размеру старнице документной базы данных – 3.
- Коэффициент пересечения чанков – 0,3.
- Максимальное наполненность токенами контекста для дополненной генерации – 300.
- Генеративная дистиллированная большая языковая модель – gemma-2-2b с квантованием Q6.

- Параметр генеративной модели, отвечающий за случайность генерации «temperature» – 0,3.
- Параметр генеративной модели, отвечающий за вариативность генерации «top_p» – 0,3.
- Промт для генеративной модели:

```
""<s>[INST]<<SYS>>
```

Вы — полезный ассистент. Отвечайте строго на поставленный вопрос, опираясь на приведённый контекст. Отвечайте по-русски, максимально кратко (1-2 предложения), без добавления новых фактов. Если точный ответ не найден в контексте, попробуй сам логически его сформулировать.

```
<</SYS>>"" ""Контекст:\n"" ""\n\nВопрос:"",
""\n\n[/INST]\nОтвет:""
```

Тестирование показало, что разработанная и оптимизированная RAG-система существенно превосходит кастомизированный GPT от OpenAI [3]:

- по ROUGE-L Recall – в 4,68 раза
- по ROUGE-L F1-score – в 3 раза
- по ROUGE-L-BERT F1-score – в 2,25 раза.

Разработанная RAG-система адаптирована для использования в корпоративной среде, поддерживает локальное развёртывание и может применяться в автоматизации обработки обращений в государственных и бизнес-структурах. Применение RAG-подхода позволяет добиться высокой точности, масштабируемости и информационной безопасности без необходимости дорогостоящего переобучения моделей.

ЛИТЕРАТУРА

- [1] NVIDIA. Retrieval-Augmented Generation Explainer [Электронный ресурс] // NVIDIA Resources. – 2025. – URL: <https://resources.nvidia.com/en-us-generative-ai-chatbot-workflow/retrieval-augmented-generation-explainer> (дата обращения: 20.02.2025).
- [2] Notion. 1 [Электронный ресурс]. – URL: <https://www.gptunnel.ru/en/blog/how-do-llms-work-simple-overview-of-a-complex-technologyhttps://pgenesis.notion.site/1-c21afe3b51ec4baaa87eafd1e1936be1> (дата обращения: 22.02.2025).
- [3] OpenAI. Custom instructions for ChatGPT [Электронный ресурс]. – URL: <https://openai.com/index/custom-instructions-for-chatgpt/> (дата обращения: 22.02.2025).

Класс функций, определённый при помощи операции ограниченного префиксного суммирования

Харченко Константин Романович

Кафедра математической кибернетики

e-mail: folstagking@gmail.com

Научный руководитель — к.ф.-м.н. м.н.с. Савицкий Игорь Владимирович

В теории алгоритмов индуктивно определённые классы вычислимых функций представляют интерес благодаря тому, что позволяют задавать вычислительные процессы строго и компактно. Такие классы задаются как множества всех функций, получаемых из некоторых начальных функций путём последовательного применения фиксированного набора операций. Индуктивное построение может осуществляться как на множестве чисел, так и на множестве слов, заданных над конечным алфавитом. При этом вычислительная мощность результирующего класса существенно зависит от выбора исходных функций и операций.

В работе исследуются два индуктивно определённых класса функций на словах из алфавита $\{1, 2\}$, задаваемые с помощью операций суперпозиции и ограниченной рекурсии по структуре слова. Каждое слово является диадическим представлением натурального числа, включая ноль, что устанавливает биективное соответствие между множеством слов и множеством натуральных чисел. Благодаря этой связи все функции в рассматриваемом классе можно трактовать двояко: либо как словарные (манипулирующие строками), либо как арифметические (оперирующие числами в диадическом представлении).

Определение. Пусть $g(\bar{x}, z)$ — функция, заданная на $\{1, 2\}^*$. Будем говорить, что функция $f(\bar{x}, y)$ получена из функции $g(\bar{x}, z)$ с помощью операции ограниченной префиксной конкатенации, если

$$f(\bar{x}, y) = \text{Con}_{z \sqsubseteq y} g(\bar{x}, z),$$

где правая часть равенства является сокращением для выражения

$$g(\bar{x}, z_0) * g(\bar{x}, z_1) * \dots * g(\bar{x}, z_k),$$

а $z_0, \dots, z_k$ — все префиксы слова y (включая пустое слово $z_0 = \Lambda$ и слово $z_k = y$) в порядке возрастания длины.

Определение. Пусть $g(\bar{x}, z)$ — функция, заданная на $\{1, 2\}^*$. Будем говорить, что функция $f(\bar{x}, y)$ получена из функции $g(\bar{x}, z)$ с помощью операции ограниченного префиксного суммирования, если

$$f(\bar{x}, y) = \sum_{z \sqsubseteq y} g(\bar{x}, z),$$

где правая часть равенства есть сокращение для выражения

$$g(\bar{x}, z_0) + g(\bar{x}, z_1) + \dots + g(\bar{x}, z_k),$$

а $z_0, \dots, z_k$ — все префиксы слова y (включая пустое слово $z_0 = \Lambda$ и слово $z_k = y$) в порядке возрастания длины.

Класс функций ВРС (Bounded Prefix Concatenation) определяется как множество всех функций, получаемых из следующих начальных функций:

$$1, 2, x * y, x \overset{\text{sup}}{\ominus} y,$$

с помощью операций суперпозиции и ограниченной префиксной конкатенации.

Класс функций ВРС (Bounded Prefix Summation) определяется как множество всех функций, получаемых из следующих начальных функций:

$$1, 2, x * y, x \overset{\text{sup}}{\ominus} y, \text{smash}(x, y),$$

с помощью операций суперпозиции и ограниченного префиксного суммирования.

Одним из базовых примеров арифметических функций, реализуемых в классе ВРС, является операция сложения двух чисел в диадическом представлении. Её можно реализовать следующим образом:

$$x + y = \sum_{z \sqsubseteq 1} \left((x \cdot \overline{\text{sg}}(z)) * (y \cdot \text{sg}(z)) \right).$$

Реализация этой же операции в классе ВРС является более сложной задачей, вывод которой приведён в работе.

Функции в классах ВРС и ВРС обладают важным свойством: их вычисления могут быть реализованы с использованием ограниченной памяти, так как каждая итерация рекурсии зависит только от оставшейся части слова и значения на предыдущем шаге. Это позволяет использовать данные классы как модели вычислений с жёсткими ресурсными ограничениями. Тем не менее, такие ограничения делают невозможной реализацию некоторых функций, например, тех, которые требуют произвольного возврата к предыдущим префиксам или повторного анализа уже обработанных фрагментов.

Определение. Пусть $k \in \mathbb{N}_0$, $q \in \mathbb{N}_0$, $r \in \mathbb{N}$, $q < r$. Пусть заданы $g(x)$, $h(x, z)$, $w(x) = \lfloor \|x\|^{\frac{q}{r}} \rfloor$, $p(x) = \|x\|^k$ на множестве $\{1, 2\}^*$. Будем говорить, что функция $f(x, y)$ получается из функций $g(x)$ и $h(x, z)$ с помощью операции w - p -ограниченной рекурсии, если выполняются соотношения:

$$\begin{cases} f(x, \Lambda) = g(x), \\ f(x, y+1) = \begin{cases} h(x, f(x, y)), & \text{если } |h(x, f(x, y))| \leq w(x) \ \& \ y \leq p(x) \ \& \\ & \& \lfloor \|x\|^{1-\frac{q}{r}} \rfloor > 1, \\ \Lambda, & \text{в противном случае.} \end{cases} \end{cases}$$

Утверждение. w - p -ограниченная рекурсия с ограничениями $p(x) = \|x\|^k$ на число шагов, $w(x) = \lfloor \|x\|^{\frac{q}{r}} \rfloor$ на размер результата на каждом шаге, и дополнительным условием $\lfloor \|x\|^{1-\frac{q}{r}} \rfloor > 1$ выражима в классе ВРС.

При наличии ограничений сохраняется корректность вычислительного процесса и предотвращается неконтролируемое увеличение ресурсоёмкости.

Результатом, позволяющим рассматривать оба класса как единый фундамент для описания полиномиально ограниченных вычислений над конечными словами, является следующее утверждение, доказанное в работе:

Утверждение. $BPC = BPS$.

Для доказательства включения $BPS \subseteq BPC$ были использованы уже имеющиеся результаты.

В [1] было показано, что определение BPC является чисто словарным способом задать хорошо изученный класс TC^0 (см., например, [2]). Иными словами:

Теорема. *Классы BPC и TC^0 совпадают.*

В [3] был введён класс FOM , предназначенный для характеристики унифицированного класса TC^0 , и было доказано следующее:

Теорема. *Класс FOM замкнут относительно операции ограниченного суммирования вида*

$$\sum_{z=0}^{|y|} g(\bar{x}, z).$$

ЛИТЕРАТУРА

- [1] Савицкий И. В. О совпадении сложностных классов BPC и TC^0 // Вестник МГУ. Серия 15. Вычислительная математика и кибернетика. 2022. №4. С. 36–45.
- [2] Allender E. W., Loui M. C., Regan K. W. Complexity classes // Algorithms and Theory of Computation Handbook. General Concepts and Techniques / Ed. Mikhail J. Atallah, Marina Blanton. Boca Raton, FL : Taylor & Francis, 2009. P. 597–620.
- [3] Barrington D. A. M., Immerman N., Straubing H. On uniformity within NC^1 // Journal of Computer and System Sciences. 1990. Vol. 41, №3. P. 274–306.

Математическое моделирование процесса распространения лазерного импульса в Керровской среде с учетом дисперсии третьего порядка

Шаргаровский Михаил Денисович

Кафедра математической физики

e-mail: s02210594@gse.cs.msu.ru

Научный руководитель — к.ф.-м.н. доц. Романенко Татьяна Евгеньевна

Уравнения нелинейной динамики играют ключевую роль в описании распространения ультракоротких оптических импульсов в волоконных системах. Одной из наиболее фундаментальных моделей является нелинейное уравнение Шредингера, которое учитывает влияние групповой скорости и нелинейных эффектов. Однако при рассмотрении сверхкоротких импульсов (пикосекундного и фемтосекундного диапазона) становится необходимым дополнение классического нелинейного уравнения Шредингера слагаемыми более высоких порядков. В частности, существенное влияние начинает оказывать дисперсия третьего порядка, которая приводит к асимметрии формы импульсов и может существенно изменять их динамику.

В данной работе рассматривается одно из уравнений, позволяющих математически описать поведение комплексной амплитуды $u(z, t)$, а именно уравнение типа Шредингера в безразмерных координатах, описывающее распространение лазерного импульса с учетом второго и третьего приближения дисперсии (SOD, TOD) в керровской среде относительно медленно меняющейся комплексной амплитуды $u(z, t)$:

$$\frac{\partial u}{\partial z} + iD_2 \frac{\partial^2 u}{\partial t^2} + D_3 \frac{\partial^3 u}{\partial t^3} + i\alpha |u|^2 u = 0, z > 0, 0 < t < L_t \quad (1)$$

Начальные условия:

$$u(0, t) = e^{-(t - \frac{L_t}{2})^2 - i\Omega(t - \frac{L_t}{2})^2} \quad (2)$$

Граничные условия:

$$u_{t=0} = \frac{\partial u}{\partial t} \Big|_{t=0} = u_{t=L_t} = 0 \quad (3)$$

Непрерывные инварианты задачи Для получения инвариантов задачи в дальнейшем будет рассматриваться не только уравнение (1), но и сопряженное к нему:

$$\frac{\partial u^*}{\partial z} - iD_2 \frac{\partial^2 u^*}{\partial t^2} + D_3 \frac{\partial^3 u^*}{\partial t^3} - i\alpha |u^*|^2 u^* = 0, z > 0, 0 < t < L_t \quad (4)$$

Рассмотрим уравнение (1) $\cdot u^* + (4) \cdot u$, проинтегрируем по всей области по t , воспользуемся формулой интегрирования по частям по переменной t , тогда с учетом условий (2), (3) получим выражение для первого инварианта, называемого так же интегралом энергии:

$$\int_0^{L_t} |u|^2 dt = I_1(z) \quad (5)$$

Проделявая аналогичные выкладки для выражения $(1) \cdot \frac{\partial u^*}{\partial t} - (4) \cdot \frac{\partial u}{\partial t}$ и пользуясь дополнительно введенным условием $\left. \frac{\partial u}{\partial t} \right|_{t=L_t} = 0$, которое согласуется с тем, что решение ищется финитным, получаем второй непрерывный инвариант задачи:

$$\int_0^{L_t} u \frac{\partial u^*}{\partial t} dt = I_2(z) \quad (6)$$

С новым условием на обнуление частной производной на правом конце отрезка, аналогично получаем из выражения $(1) \cdot \frac{\partial u^*}{\partial z} - (4) \cdot \frac{\partial u}{\partial z}$ третий непрерывный инвариант, или гамильтониан:

$$iD_2 \int_0^{L_t} u^* \frac{\partial^2 u}{\partial t^2} dt + D_3 \int_0^{L_t} \frac{\partial u}{\partial t} \frac{\partial^2 u^*}{\partial t^2} dt + \frac{i\alpha}{2} \int_0^{L_t} |u|^4 dt = I_3(z) \quad (7)$$

Разностная схема и итерационный процесс Для того, чтобы получить численное решение уравнения с учетом нелинейных слагаемых, нужно прибегнуть к методу простых итераций. Для нахождения $(s+1)$ -ой итерации решения A^{l+1} на $(l+1)$ -ом слое, обозначаемого $\overset{s+1}{A}$, нелинейные слагаемые берутся с предыдущей, (s) -ой итерации, и решается СЛАУ с пятидиагональной матрицей относительно $\overset{s+1}{A}$, где слагаемые в правой части зависят от предыдущей итерации $\overset{s}{A}$ и от значений с предыдущего слоя A^l . Для решения СЛАУ используется метод пятидиагональной прогонки. Итерационный процесс строится по следующей разностной схеме:

$$\begin{cases} \frac{\overset{s+1}{\hat{A}}_k - A_k}{h} + (iD_2 \Lambda_{\bar{t}t} + D_3 \Lambda_{ttt}) \overset{s+1}{A}_k + i\alpha |\overset{s}{A}_k|^2 \overset{s}{A}_k = 0, k = 2, \dots, N_t - 1 \\ \overset{s+1}{\hat{A}}_0 = \frac{\overset{s+1}{\hat{A}}_1 - \overset{s+1}{\hat{A}}_0}{\tau} = 0 \\ \overset{s+1}{\hat{A}}_{N_t} = 0 \end{cases} \quad (8)$$

где использованы следующие обозначения:

$$\omega_z = \{z_j = jh, j = 0, 1, \dots, N_z, hN_z = L_z\},$$

$$\omega_t = \{t_k = k\tau, k = 0, 1, \dots, N_t, \tau N_t = L_t\}$$

$$A = A_{j,k} = A(z_j, t_k), \hat{A} = A(z_{j+1}, t_k), \overset{0.5}{A} = 0.5(\hat{A} + A), |\overset{0.5}{A}|^2 = 0.5(|\hat{A}|^2 + |A|^2)$$

$$\Lambda_{ttt} = \begin{cases} \Lambda_{\bar{t}t} : (A_{k+2} - 3A_{k+1} + 3A_k - A_{k-1})/\tau^3, & k = 1 \\ \Lambda_{\bar{t}t}^\circ : (A_{k+2} - 2A_{k+1} + 2A_{k-1} - A_{k-2})/(2\tau^3), & 2 \leq k \leq N_t - 2 \\ \Lambda_{\bar{t}t} : (A_{k+1} - 3A_k + 3A_{k-1} - A_{k-2})/\tau^3, & k = N_t - 1 \end{cases}$$

и введено следующее условие останова итерационного процесса на каждом слое, учитывающее как относительную, так и абсолютную погрешность:

$$\|\overset{s+1}{\hat{A}} - A\| \leq \varepsilon_1 \|\overset{s+1}{\hat{A}}\| + \varepsilon_2, \text{ где } \varepsilon_1 = 10^{-7}, \varepsilon_2 = 10^{-9}$$

В работе показано, что все операторы частных разностных производных схемы (8) имеют второй порядок аппроксимации по каждой переменной, хотя в силу модификации оператора для третьей частной разностной производной в точках $1, N_t - 1$ порядок аппроксимации по переменной t снижается до первого, что в совокупности с граничными условиями дает первый порядок аппроксимации всей схемы по обеим переменным.

Консервативность разностной схемы Для проверки консервативности разностной схемы (8) были получены разностные аналоги интеграла энергии и гамильтониана, путем проведения разностных аналогов операций, использованных для получения непрерывных инвариантов (5), (6), (7). Обозначив за (9) разностное уравнение:

$$\frac{\hat{A}_k - A_k}{h} + (iD_2\Lambda_{\bar{t}t} + D_3\Lambda_{ttt})A_k^{0.5} + i\alpha|A_k|^{0.5}A_k^{0.5} = 0 \quad (9)$$

и за (10) сопряженное к нему уравнение:

$$\frac{\hat{A}_k^* - A_k^*}{h} + (-iD_2\Lambda_{\bar{t}t} + D_3\Lambda_{ttt})A_k^{*0.5} - i\alpha|A_k|^{0.5}A_k^{*0.5} = 0 \quad (10)$$

получим из (9) $\cdot A_k^{*0.5} + (10) \cdot A_k^{0.5}$, просуммированного от $k = 2$ до $N_t - 1$ и домноженного на τ и h , выражение для разностного аналога интеграла энергии:

$$\hat{I}_1 - I_1 = -\frac{D_3h}{2\tau^2}(-6|A_{N_t-1}^{0.5}|^2 - |A_2^{0.5}|^2 + 2\text{Re}(A_{N_t-1}^{*0.5}A_{N_t-2}^{0.5})),$$

где

$$I_1 = \tau \sum_{k=2}^{N_t-1} |A_k|^2 \quad (11)$$

Аналогично, из (9) $\cdot A_{z,k}^* - (10) \cdot A_{z,k}$ получим выражение для разностного аналога Гамильтониана:

$$\begin{aligned} \hat{I}_3 - I_3 = -\frac{D_3}{4\tau^2} & (-2|A_{N_t-2}|^2 + 2|\hat{A}_{N_t-2}|^2 + 2\text{Re}(A_{N_t-2}\hat{A}_{N_t-2}^*) + A_{N_t-2}^*A_{N_t-3} - \\ & A_{N_t-2}^*\hat{A}_{N_t-3} + \hat{A}_{N_t-2}^*A_{N_t-3} - \hat{A}_{N_t-2}^*\hat{A}_{N_t-3}), \end{aligned}$$

где

$$I_3 = \tau \sum_{k=2}^{N_t-1} \left(\frac{i\alpha}{2}|A_k|^4 - iD_2A_tA_t^* + D_3i\text{Im}(A_tA_{tt}^*) \right) \quad (12)$$

Выражения (11) и (12) являются разностными аналогами выражений (5) и (7) соответственно. В работе показано сохранение разностных инвариантов с точностью, соответствующей первому порядку по каждому из шагов сетки. Так же разностная схема была протестирована в линейном случае ($\alpha = 0, \Omega = 0$) и было показано совпадение аналитического решения и численного.

Результаты численного моделирования Для выявления эффектов, вызываемых теми или иными параметрами уравнения, была проведена серия экспериментов, где в каждом наборе решений исходные данные, кроме одного значения, одинаковые. На графиках, приведенных в тексте работы, видно, что коэффициент D_3 влияет на асимметрию решения (чем больше значения коэффициента, тем сильнее эффект), при на порядки меньших значениях D_3 относительно D_2 асимметрия практически не наблюдается (Рис. 1, $D_3 = -1, D_3 = -3, D_3 = -0.01$ соответственно); коэффициент D_2 влияет на размывание пучка во времени и наклон направления распространения импульса (чем больше значение коэффициента, тем сильнее эффект); коэффициент Ω влияет на чирпирование (аналогичный дисперсии третьего порядка эффект), коэффициент α влияет на степень самофокусировки/самодефокусировки, проявляющихся в изменении максимального $|u(z, t)|^2$ (чем больше значения коэффициентов, тем сильнее соответствующие им эффекты).

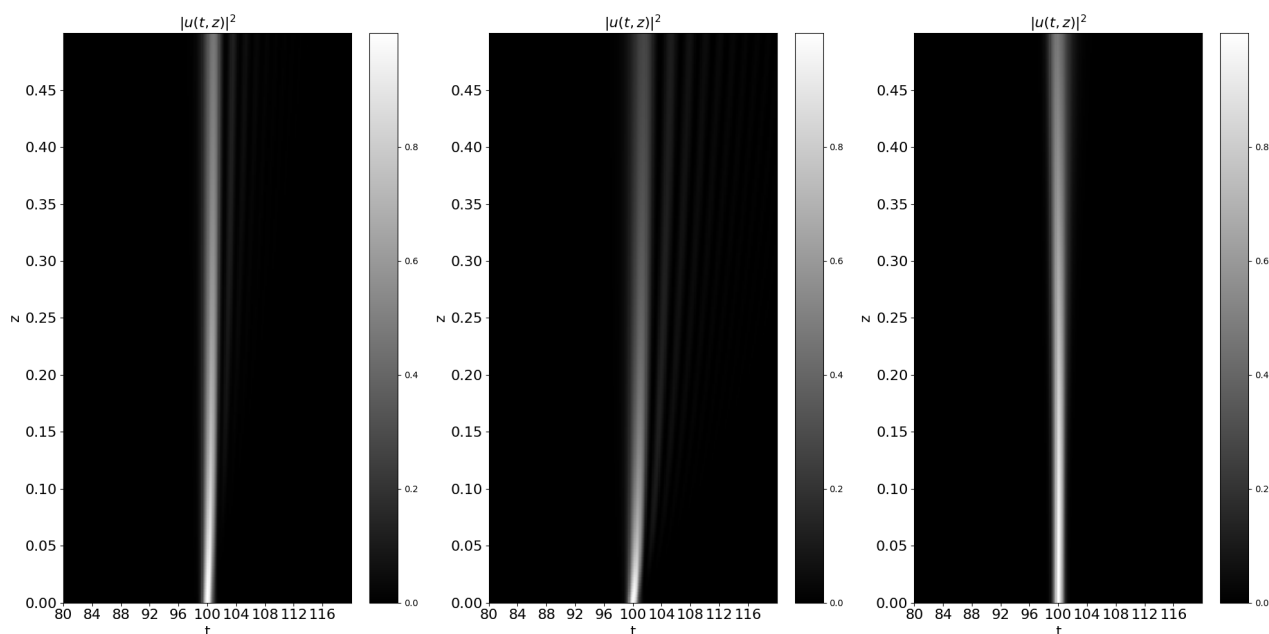


Рис. 1: Сравнение результатов моделирования для различных значений D_3 в линейном случае.

ЛИТЕРАТУРА

- [1] Agrawal G. P. Nonlinear Fiber Optics. Fifth Edition. 2013. P. 71–72.
- [2] Effect of third-order dispersion on passive mode locking / H. A. Haus, J. D. Moores, L. E. Nelson // Optics Letters. 1993. Vol. 18, N 1. P. 51–53.
- [3] Николаев Е. С. Методы решения сеточных уравнений. 2018. С. 44–45.

Аппроксимация функций распределения фи-статистик

Работа удостоена диплома 1 степени

Шеряев Антон Владимирович

Кафедра математической статистики

e-mail: sheriaev.anton@gmail.com

Научный руководитель — д.ф.-м.н. проф. Ульянов Владимир Васильевич

Введение. Рассмотрим классическую задачу математической статистики: пусть вектор $\mathbf{Y} = (Y_1, Y_2, Y_3)^\top$ имеет мультиномиальное распределение $M_3(n, \boldsymbol{\pi})$, то есть

$$\mathbb{P}(Y_1 = n_1, Y_2 = n_2, Y_3 = n_3) = \begin{cases} n! \prod_{j=1}^3 (\pi_j^{n_j} / n_j!), & n_j = \overline{0, n}, \sum_{j=1}^3 n_j = n \\ 0, & \text{иначе,} \end{cases}$$

где $\boldsymbol{\pi} = (\pi_1, \pi_2, \pi_3)^\top$, $\pi_j > 0$, $\sum_{j=1}^3 \pi_j = 1$, и считается выполненной нулевая гипотеза $H_0 : \boldsymbol{\pi} = \mathbf{p}$. Основным объектом исследования в работе является построенное по вышеуказанному распределению семейство *фи-статистик*

$$T_\phi = \frac{2n}{\phi''(1)} \cdot \sum_{j=1}^3 p_j \phi\left(\frac{Y_j}{np_j}\right),$$

где ϕ — выпуклая функция на $\mathbb{R}_+$ такая, что

- $\phi(u) \geq 0 \quad \forall u > 0$;
- $\phi(1) = \phi'(1) = 0$;
- $\phi''(1) > 0$.

Известно, что при $n \rightarrow \infty$ распределение данной статистики сходится к распределению хи-квадрат с двумя степенями свободы, однако при проверке гипотез важно понимать, насколько велика ошибка аппроксимации реального распределения предельным. В данной работе задача состоит в исследовании точности приближения хи-квадрат распределением для $\mathbb{P}(T_\phi(\mathbf{Y}) < c)$, где $c > 0$. Поскольку компоненты вектора $\mathbf{Y}$ удовлетворяют равенству $Y_1 + Y_2 + Y_3 = n$, введём случайные величины

$$X_j = (Y_j - np_j) / \sqrt{n},$$

и будем исследовать свойства вектора $\mathbf{X}$. Компоненты $\mathbf{X}$ сконцентрированы на решётке

$$L = \{\mathbf{x} = (x_1, x_2) : x_j = (n_j - np_j) / \sqrt{n}, j = 1, 2\}, \quad n_j \in \mathbb{Z}_+.$$

Тогда распределение статистики T_ϕ можно записать в виде

$$\mathbb{P}(T_\phi(\mathbf{Y}) < c) = \mathbb{P}(\mathbf{X} \in B^\phi(c)),$$

где $B^\phi = \{(x, y) : T_\phi(x, y) < c\}$ и

$$T_\phi(x, y) = \frac{2n}{\phi''(1)} \left(p_1 \phi\left(1 + \frac{x}{\sqrt{np_1}}\right) + p_2 \phi\left(1 + \frac{y}{\sqrt{np_2}}\right) + p_3 \phi\left(1 - \frac{x+y}{\sqrt{np_3}}\right) \right).$$

Можно показать, что множество $B^\phi(c)$ принадлежит классу так называемых обобщённых выпуклых множеств.

Определение. Множество $B \subset \mathbb{R}^2$ называется обобщённым выпуклым множеством, если его можно представить в следующем виде:

$B = \{(x, y) : \lambda_1(y) < x < \theta_1(y), y \in B_1\} = \{(x, y) : \lambda_2(x) < y < \theta_2(x), x \in B_2\}$, где $B_1 \subset \mathbb{R}$, $B_2 \subset \mathbb{R}$ и $\lambda_1, \theta_1, \lambda_2, \theta_2$ — непрерывные функции, определённые на B_1 и B_2 соответственно.

Ярнольд в своей работе [1] получил асимптотическое разложение для произвольного обобщённого выпуклого множества B :

$$\mathbb{P}(\mathbf{X} \in B) = J_1 + J_2 + O(n^{-1}),$$

где

$$J_1 = \iint_B \varphi(\mathbf{x}) \left(1 + \frac{1}{\sqrt{n}} h_1(\mathbf{x}) + \frac{1}{n} h_2(\mathbf{x}) \right) d\mathbf{x},$$

$$h_1(\mathbf{x}) = -\frac{1}{2} \sum_{j=1}^3 \frac{x_j}{p_j} + \frac{1}{6} \sum_{j=1}^3 x_j \left(\frac{x_j}{p_j} \right)^2,$$

$$h_2(\mathbf{x}) = \frac{1}{2} h_1^2(\mathbf{x}) + \frac{1}{12} \left(1 - \sum_{j=1}^3 \frac{1}{p_j} \right) + \frac{1}{4} \sum_{j=1}^3 \left(\frac{x_j}{p_j} \right)^2 - \frac{1}{12} \sum_{j=1}^3 x_j \left(\frac{x_j}{p_j} \right)^3,$$

и $x_3 = -x_1 - x_2$,

$$J_2 = -\frac{1}{n} \sum_{y \in L_2} \mathbb{I}_{B_1}(y) [S_1(\sqrt{n}x + p_1 n) \varphi(x, y)]_{\lambda_1(y)}^{\theta_1(y)} - \frac{1}{\sqrt{n}} \int_{-\infty}^{+\infty} \mathbb{I}_{B_2}(x) [S_1(\sqrt{n}y + p_2 n) \varphi(x, y)]_{\lambda_2(x)}^{\theta_2(x)} dx,$$

где

$$L_2 = \{y : y = (m - np_2)/\sqrt{n}, m \in \mathbb{Z}\},$$

$$S_1(x) = x - [x] - 1/2, \quad [h(x)]_{\lambda(y)}^{\theta(y)} = h(\theta(y)) - h(\lambda(y)),$$

$$\varphi(x, y) = \frac{1}{2\pi|\Omega|^{1/2}} \exp\left(-\frac{1}{2}(x, y)\Omega^{-1}(x, y)^\top\right),$$

и Ω — ковариационная матрица вектора $\mathbf{X}$.

В работе [2] получено представление для J_1 и J_2 для фи-статистик при дополнительном предположении, что ϕ четыре раза дифференцируема и $\phi^{(4)}(t)$ непрерывна в точке $t = 1$:

$$J_1^\phi = \mathbb{P}(\chi_2^2 < c) + O(n^{-1}), \quad J_2^\phi = (N^\phi - nV^\phi) \frac{e^{-c/2}}{2\pi n} \sqrt{p_1 p_2 p_3} + o(1), \quad (1)$$

где $N^\phi = |L \cap B^\phi(c)|$ — число точек решётки L в множестве $B^\phi(c)$, V^ϕ — площадь множества $B^\phi(c)$. Таким образом, задача свелась к оценке члена J_2^ϕ .

Для члена J_2^ϕ справедлива оценка Ярнольда: $J_2^\phi = O(n^{-1/2})$, однако её можно улучшить.

Рассмотрим важное подсемейство фи-статистик: степенные статистики согласия, получаемые при любом $\lambda \in \mathbb{R}$ выбором следующей функции

$$\phi_\lambda(u) = \frac{u^{\lambda+1} - (\lambda+1)(u-1) - 1}{\lambda(\lambda+1)}.$$

В работе [3] Асылбеков, Зубов и Ульянов показали, что для $\forall \lambda \in \mathbb{R}$ и для любого фиксированного $c > 0$ выполнено

$$\mathbb{P}(T_{\phi_\lambda} < c) = \mathbb{P}(\chi_2^2 < c) + O\left(n^{-50/73}(\log n)^{315/146}\right).$$

В настоящей работе доказан аналогичный результат для более широкого класса фи-статистик.

Результаты. Обозначим $\mathbb{R}_{++} := \mathbb{R}_+ \setminus \{0\}$. В данной работе доказана следующая

Теорема. Пусть $\phi \in C^4(\mathbb{R}_{++})$. Тогда $\forall c > 0$ справедлива оценка

$$J_2^\phi = O_{c,\phi,\mathbf{p}}\left(n^{-50/73}(\log n)^{315/146}\right).$$

Как следствие, получена аппроксимация $\mathbb{P}(T_\phi < c)$:

Теорема. Пусть $\phi \in C^4(\mathbb{R}_{++})$. Тогда при фиксированном $c > 0$ справедливо

$$\mathbb{P}(T_\phi < c) = \mathbb{P}(\chi_2^2 < c) + O_{c,\phi,\mathbf{p}}\left(n^{-50/73}(\log n)^{315/146}\right).$$

Замечание. Запись $O_\alpha(\cdot)$ означает, что константа, участвующая в определении O -большого, и натуральный номер N , начиная с которого неравенство начинает быть истинным, зависят от параметра α .

Замечание. В статье [4] приведена нижняя оценка на порядок $J_2^{\phi_1}$ (где $\phi_1(u) = (u-1)^2/2$): $O(n^{-3/4} \log \log n)$. Результат же в настоящей работе можно записать следующим образом:

$$\mathbb{P}(T_\phi < c) = \mathbb{P}(\chi_2^2 < c) + O_{c,\phi,\mathbf{p}}\left(n^{-3/4+\beta}\right),$$

где $\beta \approx 0,065$.

Структура доказательства. Доказательство основного результата можно поделить на три логических шага:

1. Доказательство свойств множеств $B^\phi(c)$;
2. Уточнение оценки остатка в J_2^ϕ ;
3. Применение результатов из области теории чисел для оценки главной части J_2^ϕ .

В первом шаге изучены свойства множеств $B^\phi(c)$, включая их равномерную по n ограниченность, равномерную отделимость границы от нуля, строгую выпуклость, а также существование четырежды непрерывно-дифференцируемой параметризации границы в полярных координатах. Доказанные свойства позволили заменить остаток $o(1)$, фигурирующий в (1), на $O(n^{-3/4})$, а также показать применимость теоремы Хаксли для оценки разности $N^\phi - nV^\phi$:

Теорема (Хаксли). Пусть B - выпуклая евклидова плоская область площади A , ограниченная простой замкнутой кривой C , состоящей из конечного числа частей C_i , каждая из которых три раза непрерывно-дифференцируема в следующем смысле: радиус кривизны ρ непрерывен относительно угла смежности (тангенциального угла) ψ и не равен нулю на каждой части C_i , а также непрерывно-дифференцируем относительно угла смежности. Пусть число M достаточно велико, и пусть MB обозначает множество, образованное увеличением множества B линейно в M раз. Тогда для любого изометрического вложения множества MB в евклидову плоскость число целых точек (m, n) в MB есть

$$AM^2 + O\left(IM^{46/73}(\log M)^{315/146}\right),$$

где I - число, зависящее от кривой C , но не от M и не от вложения множества MB .

Если помимо вышеуказанного части C_i четырежды непрерывно дифференцируемы в том смысле, что ρ дважды непрерывно дифференцируем по отношению к углу смежности ψ , то тогда

$$\begin{aligned} I = \sum_i \min_{C_i} \left(1 + \frac{1}{\rho^2} \left(\frac{d\rho}{d\psi} \right)^2 \right)^{-69/146} \rho^{46/73} + \\ + \sum_i \int_{C_i} \left(1 + \frac{|\rho \cdot d^2\rho/d\psi^2|}{\rho^2 + (d^2\rho/d\psi^2)^2} \right) \times \\ \times \left(1 + \frac{1}{\rho^2} \left(\frac{d\rho}{d\psi} \right)^2 \right)^{-69/146} \left| \frac{d\rho}{d\psi} \right| \rho^{-33/73} d\psi, \end{aligned}$$

при условии, что M достаточно велико для выполнения неравенств

$$M \geq \frac{1}{\rho} \quad \text{и} \quad \frac{1}{\rho^{64}} \left| \frac{d\rho}{d\psi} \right|^{53} \leq M^{11} (\log M)^{387/8}.$$

ЛИТЕРАТУРА

- [1] Yarnold J. K. Asymptotic approximations for the probability that a sum of lattice random vectors lies in a convex set // The Annals of Mathematical Statistics. 1972. Vol. 43. N 5. P. 1566—1580.

- [2] Taneichi N., Sekiya Y., Suzukawa A. An asymptotic approximation for the distribution of φ -divergence multinomial goodness-of-fit statistic under local alternatives // Journal of the Japan Statistical Society. 2001. Vol. 31. N 2. P. 207–224.
- [3] Asylbekov Zh. A., Zubov V. N., Ulyanov V. V. On approximating some statistics of goodness-of-fit tests in the case of three-dimensional discrete data // Siberian mathematical journal. 2011. Vol. 52. N 4. P. 571–584.
- [4] Hardy Gr. H. On Dirichlet's divisor problem // Proceedings of the London Mathematical Society. 1917. Vol. 2. N 1. P. 1–25.
- [5] Huxley M. N. Exponential sums and lattice points II // Proceedings of the London Mathematical Society. 1993. Vol. 3. N 2. P. 279–301.

Разработка подсистемы контроля процедурами реконфигурации в программно-конфигурируемых сетях с использованием распределенного реестра

Шпигун Игорь Кириллович

Кафедра автоматизации систем вычислительных комплексов

e-mail: shpigun.i.k@lvk.cs.msu.ru

Научный руководитель — к.ф.-м.н. в.н.с. Писковский Виктор Олегович

Разработка подсистемы контроля процедурами реконфигурации в ПКС с использованием распределенного реестра представлена как инновационный метод решения критических проблем современных ПКС, включая уязвимость к физическим сбоям оборудования, длительное время восстановления и централизованные точки отказа. Анализ существующих решений выявил фундаментальные ограничения:

- Централизованные системы хранения состояний (NetRevert [1]) создают единые точки отказа и ограничивают масштабируемость
- Большинство подходов (ReSDN [2], LegoSDN [3]) не учитывают физические сбои оборудования
- Традиционные механизмы контрольных точек требуют полной остановки обновлений на время создания контрольной точки (NetCP [4])
- Существующие решения, покрывающие обработку физического сбоя (AFRO [5]) демонстрируют рост времени восстановления пропорционально размеру сети

В основе метода лежит применение блокчейн-технологии, обеспечивающей децентрализованное хранение состояний сети и самоорганизацию компонентов, что позволяет системе адаптироваться к динамичным изменениям инфраструктуры, включая каскадные отказы узлов, при гарантии целостности данных и минимальном времени простоя.

Формальная модель сети описывает состояние как

$$S(t) = \langle D(t), E(t) \rangle,$$

где $D(t)$ представляет устройства (коммутаторы, хосты), а $E(t)$ — каналы связи с метриками пропускной способности, задержки, потерь и джиттера. При этом под корректным состоянием сети будем подразумевать, что сеть удовлетворяет всем требованиям SLA (Service Level Agreement). Этот контракт включает параметры QoS (Quality of Service), которые определяют уровень качества обслуживания, гарантируемый поставщиком услуг. Основными параметрами QoS являются доступность, процент доставленных пакетов, задержка и пропускная способность. Эти параметры позволяют клиентам легко оценить качество обслуживания.

Для мониторинга SLA необходимо сопоставление параметров QoS с метриками производительности сети (NPM - Network Performance Metrics). NPM

включает в себя такие метрики, как доступность, потери пакетов, задержка и эффективное использование канала. Эти метрики позволяют измерять производительность сети и гарантировать соответствие установленным параметрам QoS.

Блокчейн-алгоритм восстановления использует распределенный реестр на базе Cosmos SDK для хранения контрольных точек сети в виде цепочки блоков, каждый из которых содержит хеш предыдущего состояния, транзакции изменений конфигурации, временные метки. Консенсусный алгоритм обеспечивает согласованность данных между узлами, позволяя 2/3 узлов подтверждать валидность последнего корректного состояния при сбое.

Процедура отката инициируется контроллером при обнаружении сбоя и включает запрос к блокчейну за последним корректным состоянием сети, при его обнаружении происходит загрузка необходимых правил на коммутаторы. Экспериментальная проверка проводилась на стенде из 16 виртуальных машин с использованием Mininet v2.2.2 для эмуляции топологий, Golang 1.21 и Cosmos SDK v0.50 для блокчейна, а также REST API для интеграции с контроллером RunOS. Ключевыми метриками стали время восстановления и потери пакетов.

В работе представлены результаты исследования метода, доказана эффективность в создании отказоустойчивых ПКС за счет децентрализации хранения данных в распределенном реестре, устраняющей единые точки отказа, автоматизации самоорганизации компонентов, сокращающей RTO даже при каскадных сбоях, и безопасности, обеспечиваемой механизмами хеширования и консенсуса, гарантирующими целостность состояний. Перспективы развития включают оптимизацию консенсус-алгоритмов для сетей свыше 1000 узлов, интеграцию с гетерогенными средами и реализацию смарт-контрактов для автоматизации SLA-мониторинга, внося вклад в развитие децентрализованных систем и предлагая универсальный инструмент для критически важных сетевых инфраструктур.

ЛИТЕРАТУРА

- [1] Zhang Y., Beheshti N., Manghirmalani R. NetRevert: Rollback recovery in SDN //Proceedings of the third workshop on Hot topics in software defined networking. – 2014. – С. 231-232.
- [2] Chen G. R., Li C. Y., Wang K. ReSDN: A lightweight solution for data-plane state recovery in software-defined networks //2017 IEEE International Conference on Communications (ICC). – IEEE, 2017. – С. 1-7.
- [3] Chandrasekaran B., Tschaen B., Benson T. Isolating and tolerating SDN application failures with LegoSDN //Proceedings of the Symposium on SDN Research. – 2016. – С. 1-12.
- [4] Yu Y. et al. NetCP: Consistent, non-interruptive and efficient checkpointing and rollback of SDN //2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). – IEEE, 2018. – С. 1-10.

-
- [5] Kuźniar M. et al. Automatic failure recovery for software-defined networks
//Proceedings of the second ACM SIGCOMM workshop on Hot topics in
software defined networking. – 2013. – С. 159-160.