

Министерство науки и высшего образования
Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тверской государственный университет»

**СТУДЕНЧЕСКАЯ
КОНФЕРЕНЦИЯ
ФАКУЛЬТЕТА ПМиК.
СБОРНИК ТРУДОВ**

Тверь
13–30 апреля 2026 г.

Под редакцией С. М. Дудакова и Б. Н. Карлова

Тверь 2026

УДК 004, 330, 519, 658
ББК 22.18, 65.05я43
С88

Тверской государственный университет
Факультет прикладной математики и кибернетики

С88 Студенческая конференция факультета ПМиК. Сборник трудов. Тверь, 13–30 апреля 2026 г. / под ред. С. М. Дудакова и Б. Н. Карлова. — Тверь : ТвГУ, 2026. — 255 с.

Сборник содержит научные работы, представленные на студенческой конференции, проводимой факультетом прикладной математики и кибернетики Тверского государственного университета. Сборник предназначен для научных работников, аспирантов и студентов старших курсов.

УДК 004, 330, 519, 658
ББК 22.18, 65.05я43

© Тверской государственный университет, 2026
© Авторы статей, 2026

Содержание

<i>Антонов М. Е.</i> Мультизадачный анализ музыкальных композиций с помощью предобученного аудио-трансформера	6
<i>Василец А. В.</i> Сравнительный анализ вероятностных методов подавления шума в аудио- и видеосигналах	12
<i>Вихров Е. О.</i> Моделирование поведения автономного транспортного средства на основе обучения с подкреплением	20
<i>Вьюнов А. В.</i> Построение и сравнительный анализ диверсифицированных инвестиционных портфелей на основе кластеризации криптовалютных активов	27
<i>Гаврилов А. Д.</i> Детекция и трекинг игроков футбольного матча с использованием методов компьютерного зрения	34
<i>Глинский М. С.</i> Рекомендательная система на основе музыкальных предпочтений пользователя	40
<i>Ермаков А. Э.</i> Анализ дивидендной торговой стратегии на фондовом рынке	48
<i>Ивашин М. Г.</i> Построение эконометрических моделей оценки стоимости подержанных автомобилей	58
<i>Ионжуа И. Т.</i> Анализ применения искусственного интеллекта в медицине: достижения, проблемы и перспективы	67
<i>Ищенко А. А.</i> Сравнительный анализ архитектур для одновременной детекции людей и лиц в видеопотоке	79

<i>Константинова Е. А.</i> Морфологический анализ слов русского языка с использованием конечных преобразователей	87
<i>Корепанов М. В.</i> Разработка программы биометрической аутентификации сотрудников с помощью искусственного интеллекта	93
<i>Кухар Н. В.</i> Статистические методы и алгоритмы автоматической верификации результатов термобарокамерных испытаний геофизического оборудования	102
<i>Литвиненко А. С.</i> Анализ структурной трансформации топливно-энергетического баланса России	112
<i>Льянов А. А.</i> Сложность логики АТЛ* и её фрагментов	119
<i>Майоршин Р. Н.</i> Автоматическая настройка данных ГИС в рамках петрофизической интерпретационной модели	126
<i>Мальков А. С.</i> Алгоритмы защиты эхо-сигналов от несинхронных импульсных помех	136
<i>Москвичёв Я. А.</i> Разработка рекомендательной системы для совершения покупок на основе анализа пользовательского поведения с использованием методов интеллектуального анализа данных	143
<i>Оздобихин Д. В.</i> Разработка инструментария для анализа и оптимизации структуры оборотных активов предприятия на основе теории временных предпочтений	150
<i>Осокин А. Д.</i> Управление хранением и унификацией справочной информации в системе 1С	162
<i>Родионов Д. А.</i> Системный анализ производственных процессов на основе искусственного интеллекта	171

<i>Свистунов А. Н.</i> Система рекомендаций для медиаконтента	180
<i>Терешкин М. В.</i> Сравнение функций потерь Triplet Loss и ArcFace для задачи One-Shot распознавания лиц	189
<i>Тимошенко Д. В.</i> Разработка программно-интеллектуальной системы прогнозирования урожайности сельскохозяйственных культур с применением алгоритма случайного леса	194
<i>Тоньмеу Е. Л.</i> Влияние индуктивных смещений архитектур ResNet, Vision Transformer и ConvNeXt на обобщающую способность моделей классификации изображений	201
<i>Федоров И. В.</i> Выразительные возможности классов категориальных грамматик	207
<i>Харинова С. А.</i> Метод распознавания числовых показаний из сканированных анкетных документов	215
<i>Шварц Н. А.</i> Прогнозирование влияния рекомендательных систем на экономические показатели интернет-коммерции	223
<i>Эквист В. Р.</i> Создание гибридного метода классификации текстов на основе меры TF-IDF и рекуррентных нейронных сетей	235
<i>Яковлев А. М.</i> Построение нейросетевых моделей для прогнозирования урожайности	243

УДК 004.934, 004.85

Мультизадачный анализ музыкальных композиций с помощью предобученного аудио-трансформера

Антонов М. Е.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий М. Ю. Кудряшовым

Аннотация. Реализована мультизадачная модель на основе предобученного Audio Spectrogram Transformer с общим стволом и четырьмя классификационными головами: сегментация структуры, классификация возбуждения и валентности, определение жанра. Обучение проведено на датасетах DEAM, Harmonix Set и GTZAN методом Round-Robin.

Ключевые слова: глубокое обучение, мультизадачное обучение, Audio Spectrogram Transformer, музыкальный информационный поиск, классификация аудио.

Введение

Музыкальный информационный поиск (MIR) объединяет ряд задач: определение жанра трека, оценку его эмоционального содержания, выделение структурных секций (вступление, куплет, припев и т. д.). На практике каждая из них решается отдельной моделью, что требует независимого хранения весов и игнорирует общую акустическую природу входных данных. Парадигма мультизадачного обучения [2] позволяет обойти это ограничение за счёт совместного использования общего ствола сети.

В области аудиоанализа стандартом сегодня являются модели на основе механизма самовнимания [3]; одной из наиболее успешных среди них является Audio Spectrogram Transformer (AST) [1] — адаптация Vision Transformer для спектрограммного представления звука, предобученная на датасете AudioSet (около 2 млн размеченных аудиоклипов). В настоящей работе AST используется как общий

ствол мультизадачной модели, поверх которого размещены четыре классификационные головы.

1. Архитектура модели

Входной аудиосигнал длительностью 10 с с частотой дискретизации 22,05 кГц преобразуется в лог-мел-спектрограмму размера $128 \times T$, где 128 — количество мел-фильтров, T — число временных кадров. Спектрограмма разбивается на патчи 16×16 , проецируемые линейным слоем в 768-мерное пространство; к полученной последовательности добавляются обучаемый служебный токен CLS и позиционные эмбединги.

Последовательность обрабатывается стеком из 12 блоков трансформера; каждый блок содержит многоголовое самовнимание (12 голов) и полносвязную сеть с расширением до 3072 нейронов. Веса первых 8 блоков инициализируются из AST, предобученного на AudioSet, и остаются замороженными. Последние 4 блока дообучаются на целевых данных, что помогает избежать катастрофического забывания признаков, выученных на AudioSet. Применяется дифференциальная скорость обучения: $5 \cdot 10^{-5}$ для бэкбона и 10^{-3} для голов.

С выхода последнего блока берётся эмбединг CLS-токена (\mathbb{R}^{768}), агрегирующий информацию о всём фрагменте, и подаётся в общий полносвязный слой $768 \rightarrow 256$; затем результат поступает на четыре независимые головы. Общее число параметров — 87 млн, из них 14 млн обучаемые (4 блока трансформера + общий слой + головы). Схема приведена на рис. 1.

2. Мультизадачное обучение

Каждая из четырёх голов — линейный слой $256 \rightarrow K_i$ с последующей функцией SOFTMAX; число классов K_i для каждой задачи приведено в табл. 1.

Частная функция потерь \mathcal{L}_i для каждой задачи — взвешенная кросс-энтропия со сглаживанием меток ($\varepsilon = 0,1$); веса классов обратно пропорциональны их частотам в обучающей выборке, что компенсирует дисбаланс (особенно выраженный для сегментации, где Verse

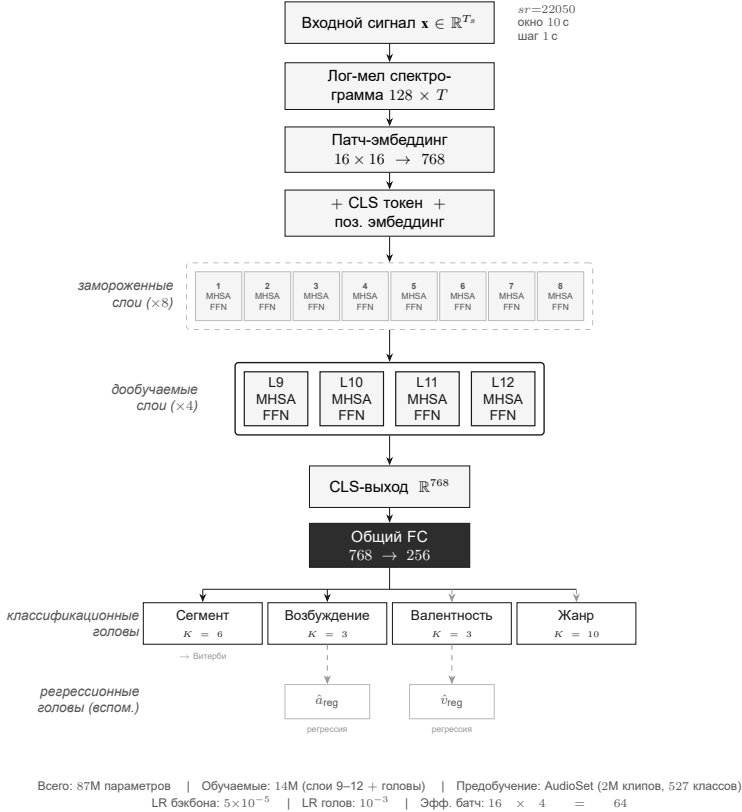


Рис. 1. Архитектура мультизадачной модели на основе AST.

и Chorus встречаются в 3–5 раз чаще Bridge и Outro). Совокупная функция потерь:

$$\mathcal{L} = \sum_{i=1}^4 \lambda_i \mathcal{L}_i,$$

где $\lambda_{seg} = 1,5$, $\lambda_i = 1,0$ для остальных задач (значения подобраны по валидационной выборке).

Используемые датасеты содержат метки только для одной из за-

Таблица 1. Классификационные головы модели.

Задача	K	Классы
Сегментация структуры	6	Intro, Verse, Bridge, Chorus, Instrumental, Outro
Возбуждение (arousal)	3	low, mid, high
Валентность (valence)	3	negative, neutral, positive
Жанр	10	blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock

доч: DEAM — эмоции, Harmonix Set — структура, GTZAN — жанр. Поэтому применяется стратегия Round-Robin: батч на каждом шаге обучения формируется только из одного датасета, потеря вычисляется только по активным головам. Общий ствол модели при этом поочерёдно получает градиенты от всех задач, реализуя жёсткое разделение параметров (*hard parameter sharing*).

3. Эксперименты и результаты

Обучающая выборка составлена из трёх открытых датасетов: DEAM (1802 трека с непрерывными аннотациями возбуждения и валентности), Harmonix Set (743 трека с разметкой структуры) и GTZAN (999 треков с разметкой по 10 жанрам). Совокупный объём данных — 3544 трека. Окна длительностью 10 секунд извлекаются с шагом 1 секунда, что даёт около 176 000 обучающих примеров.

Обучение проводилось в течение 8 эпох с использованием оптимизатора AdamW и планировщика COSINEANNEALINGWARM-RESTARTS. Размер эффективного батча — 64 (16 примеров с накоплением градиента за 4 шага). Применялась аугментация SpecAugment — маскирование случайных частотных и временных полос лог-мел-спектрограммы.

Результаты на тестовой выборке приведены в табл. 2.

На задаче определения жанра модель выходит на уровень, сопоставимый с однозадачными моделями для GTZAN (82,7% против 85–90% у SOTA). Возбуждение и валентность — 65,9% и 55,9%

Таблица 2. Метрики качества предложенной модели.

Задача	Accuracy	Macro-F1	Случайная базовая
Сегментация (6 классов)	31,4 %	31,5 %	16,7 %
Возбуждение (3 класса)	65,9 %	65,3 %	33,3 %
Валентность (3 класса)	55,9 %	56,0 %	33,3 %
Жанр (10 классов)	82,7 %	81,9 %	10,0 %

соответственно при базовой линии 33,3%; разница между ними отражает известную в литературе большую субъективность разметки валентности. Сегментация (31,4% при базовой 16,7%) остаётся наиболее сложной задачей: помимо дисбаланса классов, на качество влияет неоднозначность ручной разметки границ секций (расхождения между аннотаторами в Harmonix Set достигают ± 3 с).

Заключение

Реализована мультизадачная модель на основе предобученного AST с общим стволом и четырьмя классификационными головами. Round-Robin-обучение на трёх датасетах с непересекающимися наборами меток позволило получить рабочее качество по всем четырём задачам в рамках одной модели в 87 млн параметров: 82,7% для жанра, 65,9% и 55,9% для возбуждения и валентности, 31,4% F1 для сегментации.

Дальнейшие шаги: добавление регрессионных голов для непрерывных шкал возбуждения и валентности; постобработка предсказаний сегмента (декодирование Витерби с позиционными априорными вероятностями); расширение набора задач — определение темпа и тональности.

Список литературы

- [1] Gong, Y. AST: Audio Spectrogram Transformer / Y. Gong, Y.-A. Chung, J. Glass // Proceedings of Interspeech. — Red Hook, NY : Curran Associates, Inc., 2021. — P. 571–575. — DOI: <https://doi.org/10.21437/Interspeech.2021-698>.

- [2] Ruder, S. An Overview of Multi-Task Learning in Deep Neural Networks // arXiv. — 2017. — URL: <https://arxiv.org/abs/1706.05098>.
- [3] Vaswani, A. Attention Is All You Need / A. Vaswani, N. Shazeer, N. Parmar [et al.] // Advances in Neural Information Processing Systems (NeurIPS). — Red Hook, NY : Curran Associates, Inc., 2017. — Vol. 30. — P. 5998–6008. — URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Библиографическая ссылка

Антонов, М. Е. Мультизадачный анализ музыкальных композиций с помощью предобученного аудио-трансформера // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 6–11.

Сведения об авторах

АНТОНОВ МИХАИЛ ЕВГЕНЬЕВИЧ

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 621.391.8

Сравнительный анализ вероятностных методов подавления шума в аудио- и видеосигналах

Василец А. В.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа О. И. Сидоровой

Аннотация. В работе представлен кросс-доменный протокол тестирования алгоритмов шумоподавления. Проведён сравнительный анализ фильтра Винера, байесовского фильтра и метода на основе скрытых марковских моделей на аудио- и видеосигналах. Выявлены три фундаментальные закономерности: закон контекстной достаточности, феномен энергетико-структурной дивергенции и эффект статистического перекрытия. Результаты показывают зависимость эффективности методов от размерности пространства признаков и типа шума.

Ключевые слова: шумоподавление, фильтр Винера, байесовский фильтр, скрытые марковские модели, обработка аудио, обработка изображений, сравнительный анализ.

Введение

Современные системы обработки мультимедийных данных требуют эффективных алгоритмов шумоподавления для широкого спектра приложений: системы видеоконференцсвязи, медицинская визуализация, мобильная фотография, слуховые аппараты и системы автоматического распознавания речи. Несмотря на существование классических методов фильтрации (фильтр Винера, байесовские методы, скрытые марковские модели), в научной литературе отсутствует единая методология их сравнительного анализа для сигналов разной природы.

Исследования в области аудио- и видеообработки развиваются изолированно, что приводит к дублированию усилий и затрудняет

формирование универсальных практических рекомендаций. Инженеры вынуждены выбирать алгоритмы шумоподавления методом проб и ошибок, поскольку отсутствуют систематизированные данные об эффективности методов в различных условиях (тип шума, уровень искажений, характеристика сигнала).

Цель исследования — разработка кросс-доменного протокола тестирования алгоритмов шумоподавления и выявление фундаментальных закономерностей их работы на сигналах различной природы (аудио и изображения).

1. Материалы и методы

Объекты исследования. В качестве объектов исследования выбраны три классических алгоритма шумоподавления, представляющих различные математические подходы.

- **Фильтр Винера** — спектральный метод, работающий в частотной области на основе глобальной оценки мощности сигнала и шума.
- **Байесовский фильтр** — адаптивный метод, использующий локальную оценку дисперсии в пространственном окне.
- **Фильтр на основе НММ** — секвенциальный метод, моделирующий сигнал как последовательность скрытых состояний с известными правилами перехода.

Протокол тестирования. Для обеспечения сопоставимости результатов между доменами разработан единый протокол тестирования, включающий следующие компоненты.

1. *Единый параметр искажений.* Относительная сила шума α задаёт мощность шума как долю от мощности исходного сигнала:

$$\sigma_{\text{шума}}^2 = \alpha^2 \cdot \sigma_{\text{сигнала}}^2, \quad (1)$$

где $\alpha \in [0, 1; 1, 0]$. Это позволяет сравнивать эффективность фильтров на разных типах сигналов в одинаковых условиях.

2. *Кросс-доменные парные метрики.* Для оценки качества обработки используются согласованные метрики для аудио и изображений (табл. 1).

Таблица 1. Парные метрики оценки качества.

Тип	Аудио	Изображения
Энергетические	SNR (дБ)	PSNR (дБ)
Структурные	STOI [0; 1]	SSIM [0; 1]
Перцептивные	PESQ [-0,5; 4,5]	MS-SSIM [0; 1]

3. *Относительные метрики эффективности.* Для сравнения фильтров используются Δ -значения — изменение метрики относительно зашумлённого сигнала:

$$\Delta\text{SSIM} = \frac{\text{SSIM}_{\text{filtered}} - \text{SSIM}_{\text{noisy}}}{|\text{SSIM}_{\text{noisy}}|} \times 100\%. \quad (2)$$

4. *Систематический дизайн эксперимента.* Тестирование проводится по схеме: 3 типа шума (гауссов, цветной, импульсный) \times 10 уровней ($\alpha = 0,1 \dots 1,0$) \times 3 фильтра \times 7 сигналов в каждом домене.

Тестовые наборы данных.

Аудиосигналы (1D):

- женский и мужской речевые сигналы (высокая энтропия, нестационарные),
- музыкальный сигнал (средняя энтропия, гармоническая структура), детерминированный сигнал (низкая энтропия, периодический),
- тишина (контрольный сигнал).

Изображения (2D):

- портрет (высокая энтропия, выраженные границы),
- текстура дерева (средняя энтропия, стационарная),
- природные сцены (смешанная энтропия),
- градиентное небо (низкая энтропия, плавные переходы).

Таблица 2. Среднее улучшение метрик по типам шума ($\alpha = 0,5$). Для изображений: $\Delta SSIM$ (%), для аудио: ΔSNR (дБ).

Фильтр	Гауссов	Цветной	Импульсный
Винер (изобр.)	+227,9 %	+14,3 %	+391,4 %
Байес (изобр.)	+105,3 %	+12,63 %	+46,9 %
НММ (изобр.)	+18,1 %	+0,9 %	+12,5 %
Винер (аудио)	+0,13 дБ	+0,08 дБ	+0,04 дБ
Байес (аудио)	+4,67 дБ	+2,17 дБ	+8,9 дБ
НММ (аудио)	+2,24 дБ	+0,23 дБ	+1,01 дБ

2. Результаты

Эффективность фильтров в зависимости от типа шума. Результаты тестирования показали существенные различия в эффективности фильтров для разных типов шума (табл. 2).

Ключевые наблюдения.

- 1) Фильтр Винера демонстрирует наилучшие результаты для импульсного шума в 2D-домене (до +391,4% по SSIM), но показывает минимальное улучшение для аудио (+0,04 дБ по SNR).
- 2) Байесовский фильтр эффективен для гауссова шума в обоих доменах, но деградирует при цветном шуме.
- 3) НММ показывает умеренные результаты для речи, но неэффективен для изображений и детерминированных сигналов.

Закон контекстной достаточности. Анализ результатов выявил фундаментальную закономерность: эффективность линейных методов (Винера) при импульсном шуме экспоненциально растёт с увеличением размерности пространства признаков.

Это связано с тем, что в 1D-домене каждый отсчёт имеет только двух соседей, что недостаточно для надёжной оценки локальной статистики при импульсном шуме. В 2D-домене каждый пиксель окружён 24 соседями (окно 5×5), что обеспечивает достаточный контекст для точного выявления и замены выбросов.

Таблица 3. Влияние размерности на эффективность фильтра Винера при импульсном шуме.

Параметр	Аудио (1D)	Изображения (2D)
Число соседей в окне	2	24
$\Delta\text{SNR} / \Delta\text{SSIM} (\alpha = 0,5)$	+0,08 дБ	+211,18 %
$\Delta\text{SNR} / \Delta\text{SSIM} (\alpha = 1,0)$	+0,04 дБ	+476,85 %

Таблица 4. Метрики для синусоиды 1 кГц, гауссов шум, $\alpha = 0,5$.

Фильтр	ΔSNR (дБ)	ΔSTOI
Байесовский	-1,89	+0,0023
НММ	-0,54	+0,0009
Винер	+0,43	+0,002

Феномен энергетико-структурной дивергенции. При тестировании на детерминированных сигналах (синусоида 1 кГц, однородные области изображений) обнаружен эффект: адаптивные вероятностные методы (байесовский фильтр, НММ) демонстрируют противоречивое поведение метрик.

Они ошибочно интерпретируют детерминированный сигнал как шум (локальная дисперсия синусоиды близка к нулю), что приводит к его подавлению (отрицательный ΔSNR). Однако перцептивные метрики (STOI, PESQ) показывают рост, поскольку они оценивают разборчивость речи, а не энергетическую точность. Фильтр Винера избегает этой проблемы благодаря глобальной спектральной оценке.

Эффект статистического перекрытия. Для цветного шума (спектрально коррелированного) все фильтры демонстрируют снижение эффективности по сравнению с гауссовым шумом (табл. 5): спектр сигнала и шума перекрываются, что делает невозможным их надёжное разделение на основе частотных или статистических признаков. Это фундаментальное ограничение для всех вероятностных методов, не зависящее от размерности домена.

Таблица 5. Снижение эффективности для цветного шума относительно гауссова.

Фильтр	Аудио (Δ SNR)	Изображения (Δ SSIM)
Винер	$\times 0,02$	$\times 0,06$
Байесовский	$\times 0,67$	$\times 0,18$
НММ	$\times 0,33$	< 0

3. Обсуждение

Практические рекомендации. На основе полученных результатов сформулированы рекомендации по выбору фильтра для различных сценариев.

- **Импульсный шум (изображения):** фильтр Винера (преимущество до +476 % по SSIM).
- **Гауссов шум (аудио):** байесовский фильтр (+8,3 дБ по SNR для речи).
- **Цветной шум:** все методы показывают ограниченную эффективность; требуются специализированные подходы.
- **Детерминированные сигналы:** фильтр Винера (избегает энергетико-структурной дивергенции).
- **Речь в реальном времени:** НММ (сохраняет фонетическую структуру при умеренном шуме).

Ограничения протокола.

- 1) Упрощённая реализация PESQ (относительная Δ -метрика вместо абсолютных баллов MOS).
- 2) Глобальная оценка параметров шума (не адаптивна к нестационарным условиям).
- 3) Отсутствие глубокого обучения в сравнении (только классические методы).

Направления дальнейших исследований.

- Разработка адаптивной оценки параметров шума для нестационарных условий.
- Расширение протокола на видео (3D-домен: пространство + время).
- Интеграция классических и нейросетевых методов (гибридные подходы).
- Валидация на пользовательских исследованиях (субъективная оценка качества).

Заключение

Разработан кросс-доменный протокол тестирования алгоритмов шумоподавления и выявлены три фундаментальные закономерности: закон контекстной достаточности (экспоненциальный рост эффективности Винера в 2D при импульсном шуме), феномен энергетико-структурной дивергенции (противоречивое поведение метрик у адаптивных методов на детерминированных сигналах) и эффект статистического перекрытия (фундаментальное ограничение для цветного шума).

Практическая значимость результатов заключается в возможности формирования обоснованных рекомендаций по выбору фильтра для конкретных условий эксплуатации, что особенно важно для систем реального времени с ограниченными вычислительными ресурсами.

Список литературы

- [1] Тестирование эффективности гибридной методики шумоподавления в речевом сигнале для системы видеоконференций / С. В. Белов, С. С. Катунин, И. Ю. Кучин, М. Ф. Руденко // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. — 2025. — № 1. — С. 93–102. — DOI: <https://doi.org/10.24143/2072-9502-2025-1-93-102>.

- [2] Gonzalez, R. C. Digital Image Processing, Global Edition / R. C. Gonzalez, R. E. Woods. — 4th ed. — Harlow : Pearson Education, 2018. — 1024 p. — ISBN 978-1-292-22307-0. — URL: https://www.imageprocessingplace.com/downloads_V3/dip4e_downloads/dip4e_sample_book_material/dip4e_sample_chapters.pdf.
- [3] Vaseghi, S. V. Advanced Digital Signal Processing and Noise Reduction / S. V. Vaseghi. — 4th ed. — Chichester : John Wiley & Sons, 2008. — 514 p. — ISBN 978-0-470-75406-1. — DOI: <https://doi.org/10.1002/9780470740156>.

Библиографическая ссылка

Василец, А. В. Сравнительный анализ вероятностных методов подавления шума в аудио- и видеосигналах // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 12–19.

Сведения об авторах

ВАСИЛЕЦ АНАСТАСИЯ ВЯЧЕСЛАВОВНА

Студентка магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 004.89

Моделирование поведения автономного транспортного средства на основе обучения с подкреплением

Вихров Е. О.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий А. Б. Семёновым

Аннотация. В статье рассматривается разработка прототипа системы автономного управления грузовым транспортным средством. Основной задачей является построение программной архитектуры, объединяющей данные телеметрии, навигационный модуль, компьютерное зрение и агента обучения с подкреплением. В качестве модели принятия решений используется алгоритм Soft Actor-Critic, формирующий непрерывное управление транспортным средством: угол поворота рулевого колеса, степень нажатия на газ и тормоз. Для описания состояния среды используются признаки, получаемые из нескольких источников: положение транспорта на карте, расстояние и направление до следующей контрольной точки, параметры движения, информация о полосе, дорожной области и объектах впереди транспортного средства.

Ключевые слова: автономное транспортное средство, обучение с подкреплением, Soft Actor-Critic, компьютерное зрение.

Введение

Разработка автономных транспортных средств является одной из наиболее сложных задач современной прикладной информатики. Для построения подобных систем требуется не только распознавать дорожную сцену, но и принимать решения в условиях неполной информации, динамического изменения окружающей среды и необходимости непрерывного управления. В реальных условиях обучение автономного агента связано с высокой стоимостью экспериментов и риском повреждения оборудования. Поэтому на начальном этапе целесообразно использовать симуляционные среды, позволяющие

безопасно собирать данные, проверять алгоритмы и отлаживать архитектуру системы. В данной работе в качестве такой среды выбран Euro Truck Simulator 2. Эта среда моделирует движение грузового автомобиля по дорожной сети, содержит навигационные задания, трафик, штрафы, ограничения скорости и элементы дорожной инфраструктуры. Цель данной работы — разработать программную архитектуру автопилота грузового автомобиля на основе обучения с подкреплением. В качестве агента, решающего задачу доставки грузов, использовалась модель обучения с подкреплением. В процессе взаимодействия со средой агент получал численную оценку своих действий в виде вознаграждения. Положительное вознаграждение начислялось за продвижение по маршруту, удержание транспортного средства в пределах полосы движения, поддержание безопасной скорости и успешное завершение доставки. Отрицательное вознаграждение, напротив, назначалось за действия, повышающие риск столкновения, приводящие к отклонению от маршрута, нарушению правил дорожного движения или необоснованной остановке.

Работа была разделена на несколько самостоятельных блоков. Первый блок отвечает за получение телеметрии грузовика: скорости, положения, состояния заказа, штрафов и других параметров. Второй блок работает с картой и определяет направление движения к точке доставки. Третий блок реализует компьютерное зрение и извлекает информацию о полосе движения, дорожной области и объектах в кадре. Четвёртый блок объединяет все признаки в вектор состояния и передаёт его агенту обучения с подкреплением. Агент на основе архитектуры Soft Actor-Critic выдаёт распределение непрерывных действий, которые затем преобразуются в команды управления грузовиком: газ, тормоз и поворот руля.

1. Получение данных из окружающей среды

Блок телеметрии считывает параметры, доступные через SCS SDK. В состояние включаются скорость, текущая передача, обороты двигателя, остаток топлива, предупреждения систем грузовика, состояние фар и поворотников, стояночный тормоз, текущие значения руля, газа и тормоза, а также ограничение скорости. Отдельно фиксируются признаки доставки: наличие активного задания, город и компания назначения, факт завершения доставки, масса груза

и наличие прицепа. Кроме того, считаются координаты и углы поворота грузовика, необходимые для привязки к карте.

Эти данные используются как часть наблюдения и как источник сигналов для функции награды. Штрафы, простой без причины, превышение скорости и движение от цели уменьшают вознаграждение, а продвижение по маршруту и успешная доставка увеличивают его. Поэтому телеметрия связывает задачу управления не только с локальным удержанием полосы, но и с глобальной целью доставки.

2. Навигационный модуль

Для определения направления движения реализован граф карты ETS2. Вершинами графа являются дорожные узлы, а рёбрами — участки дорог и паромные соединения. Для дорог создаются два направленных перехода, которые являются возможными направлениями движения автотранспорта.

Навигатор получает текущие координаты грузовика и цель доставки. Цель определяется по идентификаторам города и компании из телеметрии. После построения маршрута положение грузовика сопоставляется с ближайшим допустимым сегментом. При этом учитываются не только расстояние до сегмента, но и направление движения, это важно для поиска корректного маршрута через сложные системы развязок на трассе.

Результатом работы модуля является пакет признаков: оставшееся расстояние по маршруту, расстояние до следующей контрольной точки, индекс текущего сегмента и нормированная поправка курса для достижения следующей контрольной точки по маршруту движения.

3. Модуль визуального восприятия

Визуальный блок извлекает признаки дорожной сцены с помощью виртуальной камеры, подобно реальным системам. Для этого используются две предобученные модели. Модель семейства YOLO выполняет детекцию объектов, влияющих на безопасность движения. Модель YOLOP применяется для сегментации проезжей области и разметки полосы.

На основе масок вычисляются геометрические признаки: смещение грузовика относительно центра полосы, ошибка направления, ширина коридора, уверенность распознавания полосы, признак потери разметки и индикаторы приближения к перекрёстку. После детекции объектов формируется описание трафика: количество объектов, занятость центральной, левой и правой зон перед грузовиком, минимальные расстояния до препятствий и оценка риска. Вместо полного изображения агент получает числовые признаки, непосредственно связанные с управлением.

4. Архитектура агента

В качестве алгоритма обучения с подкреплением был выбран Soft Actor-Critic [2]. Данный алгоритм относится к классу off-policy методов актор-критик и предназначен для задач с непрерывным пространством действий. Все действия агента возвращаются в окружающую среду в виде вещественных значений: угол поворота рулевого колеса, степень нажатия на газ и тормоз. Выбор SAC обусловлен тем, что алгоритм хорошо подходит для управления динамическими объектами в непрерывной среде и устойчиво работает с буфером опыта.

5. Формализация задачи

Задача управления формализуется как марковский процесс принятия решений [1]. Состояние среды в момент времени t описывается вектором

$$s_t \in \mathbb{R}^{44}.$$

Вектор содержит нормированные признаки скорости и ограничения скорости, текущие управляющие воздействия, флаги активного задания и загруженного прицепа, расстояние до цели, поправку курса, параметры полосы движения и показатели риска от объектов дорожной сцены.

Пространство действий непрерывно:

$$a_t = (u_t^{\text{steer}}, u_t^{\text{gas}}, u_t^{\text{brake}}),$$

где $u_t^{\text{steer}} \in [-1; 1]$, а $u_t^{\text{gas}}, u_t^{\text{brake}} \in [0; 1]$.

Один эпизод соответствует одной активной доставке. Эпизод начинается, когда есть задание по доставке, прицеп присоединён к транспорту, а маршрут до точки назначения построен с помощью соответствующего модуля. Завершение происходит при успешной доставке или при неудачном окончании задания.

6. Функция награды и обучение

Награда составлена из нескольких слагаемых. Основной положительный компонент — прогресс по маршруту: если оставшееся расстояние уменьшается, агент получает положительный сигнал. Дополнительное вознаграждение начисляется за приближение к следующей контрольной точке. Компонент скорости поощряет движение около целевой скорости, которая снижается при занятой передней зоне, высоком риске или перекрёстке. Превышение ограничения скорости штрафует.

Геометрический компонент связан с удержанием полосы. При высокой уверенности распознавания полосы агент штрафует за смещение относительно её центра и за ошибку направления. Если модуль восприятия сообщает о высоком риске впереди или о потере дорожного коридора, добавляется штраф безопасности. Отдельно учитываются штрафы ПДД из окружающей среды, отданные агенту за управление транспортного средства, а также простой без видимой причины.

Обобщённо награду можно записать в виде

$$r_t = r_t^p + r_t^c + r_t^v + r_t^l + r_t^{risk} + r_t^{fine} + r_t^{term},$$

где r_t^p отвечает за общий прогресс, r_t^c — за движение к контрольной точке, r_t^v — за скорость, r_t^l — за геометрию полосы, r_t^{risk} — за безопасность, r_t^{fine} — за штрафы ПДД, а r_t^{term} — за результат доставки.

Для обучения используется SAC с MLP-политикой. В конфигурации заданы буфер воспроизведения, число начальных шагов до обучения, коэффициент дисконтирования, скорость обновления целевых сетей и автоматическая настройка энтропийного коэффициента. Архитектура политики состоит из трёх полносвязных слоёв размерностей 256, 256 и 128. Переходы сохраняются в буфер опыта, а обновление модели выполняется во время активной доставки.

7. Возвращение действий агента в окружающую среду

После выбора действия значения газа, тормоза и руля передаются в модуль управления. Этот модуль использует контроллер SCS и записывает нормированные значения в поля движения вперёд, торможения и поворота руля. Перед отправкой в окружающую среду эти значения приводятся к допустимым диапазонам: газ и тормоз принимают значения от 0 до 1, а угол поворота руля — от -1 до 1 в соответствии с документацией контроллера.

Заключение

В работе разработан прототип системы моделирования поведения автономного грузового транспортного средства на основе обучения с подкреплением. Средой взаимодействия выбран Euro Truck Simulator 2, а агент построен на алгоритме SAC, подходящем для непрерывного управления. В единую архитектуру объединены телеметрия, графовая навигация, визуальное восприятие на основе YOLO и YOLOP, формирование 44-мерного наблюдения и контроллер действий.

Список литературы

- [1] Sutton, R. S. Reinforcement Learning: An Introduction / R. S. Sutton, A. G. Barto. — 2nd ed. — Cambridge, MA : MIT Press, 2018. — 552 p. — URL: <https://mitpress.mit.edu/9780262039246/reinforcement-learning/>.
- [2] Haarnoja, T. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor / T. Haarnoja, A. Zhou, P. Abbeel, S. Levine // Proceedings of the 35th International Conference on Machine Learning. — PMLR, 2018. — Vol. 80. — P. 1861–1870. — URL: <https://proceedings.mlr.press/v80/haarnoja18b.html>.

Библиографическая ссылка

Вихров, Е. О. Моделирование поведения автономного транспортного средства на основе обучения с подкреплением // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 20–26.

Сведения об авторах

Вихров Евгений Олегович

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 336.76, 519.237.8

Построение и сравнительный анализ диверсифицированных инвестиционных портфелей на основе кластеризации криптовалютных активов

Вьюнов А. В.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа О. И. Сидоровой

Аннотация. В работе предложен алгоритм фильтрации криптовалютных активов, который убирает из портфеля шумные и топологически нестабильные инструменты до построения ковариационной матрицы. Фильтр опирается на два независимых критерия: локальный PАС-индекс и идиосинкразическую волатильность к биткоину. На выборке из 234 криптоактивов за 2020–2025 гг. фильтр значительно снизил CVaR и максимальную просадку по сравнению с нефильтрованной MV-оптимизацией и портфелем $1/N$.

Ключевые слова: криптовалюты, портфельная оптимизация, консенсусная кластеризация, идиосинкразическая волатильность, хвостовые риски.

Введение

Криптовалютный рынок плохо описывается классической портфельной теорией. Распределения дневных доходностей здесь имеют тяжёлые хвосты, волатильность кластеризуется во времени, а корреляции между активами заметно меняются от одной фазы рынка к другой. В таких условиях оптимизация Марковица и иерархический паритет рисков (Hierarchical Risk Parity, HRP) [1] работают на пределе своих возможностей, и плохо обусловленная ковариационная матрица оказывается главным источником ошибок.

Стандартный способ борьбы с этой проблемой — сжатие или регуляризация уже построенной ковариационной матрицы. В работе

выбран другой путь. Мы строим фильтр, который удаляет проблемные активы до того, как ковариационная матрица будет посчитана и передана в оптимизатор. Чем меньше шумных инструментов попадёт в матрицу, тем устойчивее окажется решение оптимизации.

Фильтр построен на двух критериях. Первый критерий — адаптация локального индекса доли неоднозначной кластеризации (Proportion of Ambiguous Clustering, PAC) из работы [3], измеряющего устойчивость кластерной принадлежности отдельного актива. Второй критерий — идиосинкразическая волатильность (idiosyncratic volatility, IVOL), вычисляемая как стандартное отклонение остатков скользящей регрессии доходности актива на доходность биткоина (BTC) в качестве рыночного прокси. Решение исключать актив принимается только при одновременном превышении обоих порогов. Это конъюнктивное правило сохраняет в портфеле так называемых истинных диверсификаторов — активы, которые могут блуждать между кластерами по экономическим причинам, но при этом имеют низкий шумовой риск.

1. Алгоритм двухфакторной фильтрации

Алгоритм работает в скользящем окне. На каждом обучающем окне выполняются три шага: оценка PAC, оценка IVOL, конъюнктивная фильтрация по двум полученным значениям.

Локальный индекс PAC. Сначала строится консенсусная матрица $C = (C_{ij})$. Для этого B раз генерируется бутстрэп-выборка методом стационарного блочного бутстрэпа [2], и на каждой выборке запускается алгоритм K -медоидов. В качестве меры близости берётся ранговое расстояние Спирмена $d_{ij} = \sqrt{0,5(1 - \rho_{ij})}$. Элемент C_{ij} — доля итераций, в которых активы i и j попадали в один кластер. Число кластеров не фиксируется заранее, а подбирается на каждом окне максимизацией индекса силуэта (Silhouette Score) на сетке $K \in [2; K_{\max}]$. Это нужно для адаптации к меняющейся структуре рынка — в спокойные периоды кластеров обычно больше, в стрессовые меньше.

Локальный индекс PAC для актива i задаётся формулой

$$\text{PAC}_i = \frac{1}{N-1} \sum_{j \neq i} \mathbb{I}(\tau_{\text{lower}} < C_{ij} < \tau_{\text{upper}}), \quad (1)$$

где $\mathbb{I}(\cdot)$ — индикаторная функция, N — число активов в окне, $\tau_{\text{lower}} = 0,1$ и $\tau_{\text{upper}} = 0,9$ — границы зоны топологической неоднозначности. Большое значение PAC_i говорит о том, что актив i систематически меняет кластерную принадлежность от одной бутстрэп-выборки к другой — такой актив трудно отнести к какой-либо устойчивой группе. В исходной работе [3] PAC применяется к разбиению в целом, для выбора оптимального K . В нашем случае метрика декомпозирована до уровня отдельного актива, что позволяет точно находить источники нестабильности.

Идиосинкразическая волатильность. Для каждого актива i строится серия скользящих однофакторных регрессий

$$r_{i,t} = \alpha_i^{(\tau)} + \beta_i^{(\tau)} r_{\text{BTC},t} + \varepsilon_{i,t}, \quad t \in [\tau - h; \tau], \quad (2)$$

где $h = 60$ дней — длина окна, $\varepsilon_{i,t}$ — ненаблюдаемая случайная ошибка регрессии. По МНК-оценкам коэффициентов $\hat{\alpha}_i^{(\tau)}$ и $\hat{\beta}_i^{(\tau)}$ вычисляются остатки регрессии $\hat{\varepsilon}_{i,t} = r_{i,t} - \hat{\alpha}_i^{(\tau)} - \hat{\beta}_i^{(\tau)} r_{\text{BTC},t}$. Метрика IVOL_i — это среднее стандартных отклонений остатков $\hat{\varepsilon}_{i,t}$ по всем скользящим окнам обучающего периода. Скользящая, а не глобальная регрессия здесь принципиальна: в крипторынке коэффициент β заметно меняется при смене режима, и фиксированный β дал бы смещённую оценку идиосинкразической компоненты.

Конъюнктивный критерий исключения. Актив исключается из универса тогда и только тогда, когда выполнены оба условия:

$$\text{PAC}_i > \tau_{\text{PAC}} \quad \text{и} \quad \text{IVOL}_i > \tau_{\text{IVOL}}. \quad (3)$$

Порог $\tau_{\text{PAC}} = 0,9$ выбран консервативно — так исключаются только активы, у которых почти все парные связи попадают в зону неоднозначности. Порог τ_{IVOL} подбирается на каждом окне поиском по сетке $\{0,02; 0,03; 0,05; 0,07; 0,10\}$. Окно делится в пропорции 70:30 на обучающую и валидационную часть, критерий выбора — минимум CVaR (95 %) на валидации.

2. Дизайн эмпирического исследования

Данные. Дневные цены закрытия 234 криптоактивов за период 2019–2025 гг. Состав универса обновляется ежегодно по топ-100

капитализации на 1 января; стейблкоины и обёрнутые токены исключены. Чтобы избежать ошибки выжившего, при делистинге актива фиксируется терминальный убыток $\log(10^{-6}) \approx -13,8$, что соответствует потере примерно 99,9999%. Для новых активов введён карантин 90 дней. Дневные доходности винзоризованы на уровне 1% и 99% квантилей, терминальные наблюдения маскируются.

Walk-Forward бэкстест. Тестовый период — 72 месяца с ежемесячной ребалансировкой. На каждой ребалансировке обучающее окно длиной 365 дней и тестовое окно одного месяца не пересекаются. Стационарный бутстрэп выполнялся с $B = 200$ итерациями и средней длиной блока $L = 10$ дней.

Стратегии и абляционный дизайн. Мы сравниваем восемь стратегий. Первые две — это минимально-дисперсионный портфель (MV-оптимизация, $w \propto \Sigma^{-1} \mathbf{1}$) и HRP с методом связи Уорда. Каждая из них берётся в трёх конфигурациях: без фильтрации (raw), только с PAC-фильтром (PAC-only) и с полным двухфакторным фильтром (PAC + IVOL). Седьмая стратегия — равновзвешенный портфель $1/N$, восьмая — buy & hold. Транзакционные издержки учтены на уровне 0,25% за одну сторону сделки.

Статистическая верификация. Попарное сравнение коэффициентов Шарпа выполнено тестом Ледуа–Вольфа. Доверительные интервалы разностей для CVaR (95%) и максимальной просадки (Maximum Drawdown, MDD) построены стационарным блочным бутстрэпом, $B = 1000$. Для контроля групповой вероятности ошибки первого рода (Family-Wise Error Rate, FWER) при множественных сравнениях применена процедура Хольма.

3. Результаты эмпирического исследования

В табл. 1 приведены ключевые метрики стратегий за весь тестовый период. Полная фильтрация (PAC + IVOL) дала MV-стратегии лучшую максимальную просадку ($-38,64\%$ против $-82,66\%$ у $1/N$) и наименьший CVaR (0,054). Коэффициент оборачиваемости портфеля у отфильтрованных стратегий в 3–4 раза меньше, чем у конфигурации PAC-only (17–24% против 50–66%). Это первый явный признак того, что IVOL-фильтр играет роль стабилизатора состава портфеля.

Таблица 1. Ключевые метрики стратегий за 2020–2025 гг.

Стратегия	Шарп	MDD, %	CVaR (95 %)	Оборачиваемость
MVO + PAC + IVOL	1,28	−38,6	0,054	0,18
MVO + PAC	1,35	−48,6	0,078	0,66
MVO (raw)	1,34	−48,3	0,055	0,16
HRP + PAC + IVOL	1,01	−53,3	0,078	0,24
HRP (raw)	0,73	−81,9	0,084	0,16
1/N	1,07	−82,7	0,087	0,04
Buy & hold	0,87	−79,5	0,089	0,01

Дифференциальные роли двух факторов. Абляционный анализ показал, что PAC и IVOL работают не одинаково. Изолированное применение PAC увеличивает риск-скорректированную доходность: для MV-стратегии коэффициент Шарпа растёт с 1,34 (raw) до 1,35 (PAC-only); для HRP — с 0,73 до 1,15. Но у PAC-only конфигураций высокий коэффициент оборачиваемости портфеля (50–66 %) и слабая хвостовая защита (CVaR \approx 0,078). Добавление IVOL снижает оборачиваемость в 3–4 раза и заметно улучшает хвостовые метрики, хотя часть коэффициента Шарпа при этом теряется. Получается, что каждый критерий работает на свою метрику: PAC увеличивает эффективность инвестиций, а IVOL отвечает за хвостовую защиту и стабильность состава. Именно это даёт эмпирическое обоснование конъюнктивному критерию (3): каждый фактор по отдельности решает только половину задачи.

Стресс-тестирование. В табл. 2 приведены просадки для двух периодов экстремальных шоков 2022 года — коллапса экосистемы Тетра-LUNA и банкротства криптобиржи FTX. Эффект фильтра здесь принципиально разный. На LUNA-кризисе максимальная просадка стратегии MV + PAC + IVOL составила −23,0 % против −53,8 % у 1/N и −35,7 % у нефильтованной MVO — разрыв 30 и 13 процентных пунктов соответственно. На FTX расхождение с MVO + raw уже меньше одного процентного пункта. Это можно объяснить природой шоков: коллапс LUNA развивался идиосинкразически (фильтр успевал отсеять проблемные токены), а контрагентский кризис FTX распространялся системно, и здесь от идиосинкразиче-

Таблица 2. Максимальная просадка в стрессовых периодах, %.

Стратегия	LUNA-2022	FTX-2022
MVO + PAC + IVOL	-23,0	-18,2
MVO (raw)	-35,7	-18,3
HRP + PAC + IVOL	-32,6	-27,4
HRP (raw)	-53,1	-31,6
1/N	-53,8	-34,1

ского фильтра много ждать не приходится.

Статистическая значимость. Из 18 попарных тестов (две отфильтрованные стратегии \times три бенчмарка \times три метрики) после поправки Хольма с $\text{FWER} = 0,05$ отвергнуто 13 гипотез. В частности, MVO + PAC + IVOL значимо лучше MVO + raw по CVaR ($p_{\text{Holm}} = 0,007$) и MDD ($p_{\text{Holm}} = 0,018$). Все девять сравнений MVO + PAC + IVOL с 1/N и нефильТРованным HRP отвергают нулевую гипотезу на уровне $p_{\text{Holm}} < 0,001$.

Анализ устойчивости. Однофакторный анализ чувствительности к параметрам τ_{lower} , τ_{upper} , τ_{IVOL} , K и длине бутстрэп-блока на референтном валидационном окне показал, что коэффициент Шарпа меняется в узком коридоре 1,88–2,04. Это говорит о том, что результаты устойчивы к выбору гиперпараметров и сохраняются при разумных вариациях.

Заключение

В работе предложен и эмпирически проверен двухфакторный фильтр криптовалютных активов на основе локального PAC-индекса и идиосинкразической волатильности к биткоину. Главный результат тестирования — статистически значимое снижение CVaR и максимальной просадки по сравнению с нефильТРованной MV-оптимизацией и наивным бенчмарком 1/N. Абляционный анализ показал, что PAC и IVOL работают на разные стороны задачи: PAC отвечает за повышение риск-скорректированной доходности, IVOL — за хвостовую защиту и стабильность состава портфеля. Это объясняет, почему именно конъюнктивное правило исключения даёт лучший компромисс между доходностью и риском. Среди возможных направлений

дальнейшей работы — адаптивная калибровка порога τ_{RAC} , интеграция с методами сжатия ковариационной матрицы и расширение алгоритма на смешанные портфели криптовалют и традиционных активов.

Список литературы

- [1] Lopez de Prado, M. Building Diversified Portfolios that Outperform Out of Sample / M.Lopez de Prado // Journal of Portfolio Management. — 2016. — Vol. 42, №4. — P. 59–69. — DOI: <https://doi.org/10.3905/jpm.2016.42.4.059>.
- [2] Politis, D.N. The Stationary Bootstrap / D.N.Politis, J.P.Romano // Journal of the American Statistical Association. — 1994. — Vol. 89, №428. — P.1303–1313. — DOI: <https://doi.org/10.1080/01621459.1994.10476870>.
- [3] Şenbabaoğlu, Y. Critical limitations of consensus clustering in class discovery / Y.Şenbabaoğlu, G.Michailidis, J.Z.Li // Scientific Reports. — 2014. — Vol. 4. — Article 6207. — DOI: <https://doi.org/10.1038/srep06207>.

Библиографическая ссылка

Вьюнов, А. В. Построение и сравнительный анализ диверсифицированных инвестиционных портфелей на основе кластеризации криптовалютных активов // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 27–33.

Сведения об авторах

Вьюнов Антон Вадимович
Студент магистратуры
Направление «Прикладная информатика»

УДК 004.932.2, 004.85

Детекция и трекинг игроков футбольного матча с использованием методов компьютерного зрения

Гаврилов А. Д.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий М. Ю. Кудряшовым

Аннотация. Рассмотрена задача автоматической детекции и многообъектного трекинга игроков футбольного матча на видеозаписях. Разработана модульная система на основе свёрточной нейронной сети YOLOv8m, дообученной на специализированном наборе данных, и алгоритма трекинга ByteTrack. На валидационной выборке достигнута значения mAP50 = 0,831 и точности 0,960.

Ключевые слова: детекция объектов, YOLOv8, многообъектный трекинг, компьютерное зрение, футбол.

Введение

Профессиональный футбол генерирует значительные объёмы видеоданных: каждый матч фиксируется множеством камер. Тренерские штабы, аналитические компании и вещательные организации нуждаются в инструментах автоматического анализа происходящего на поле. Ручная обработка видеозаписей требует значительных временных затрат и подвержена субъективности.

Задача автоматического анализа футбольного видео включает несколько подзадач: детекцию (обнаружение) объектов — игроков, вратарей, судей и мяча; трекинг — отслеживание каждого объекта на протяжении видеопоследовательности с сохранением уникального идентификатора; и классификацию — определение принадлежности игрока к команде.

Развитие глубокого обучения, в частности свёрточных нейронных сетей, привело к появлению эффективных одностадийных детекторов семейства YOLO, способных работать в реальном времени. В со-

четании с современными алгоритмами многообъектного трекинга, такими как ByteTrack [1], это открывает возможности для построения практических систем анализа спортивного видео.

Целью данной работы является разработка системы автоматической детекции и трекинга игроков футбольного матча на видеозаписях на основе методов глубокого обучения и компьютерного зрения.

1. Методы детекции и трекинга

Для детекции объектов выбрана архитектура YOLOv8 [3] — восьмая версия семейства YOLO. В отличие от двухстадийных детекторов, таких как Faster R-CNN, YOLOv8 является одностадийной: за один проход через нейронную сеть формируются все предсказания координат ограничивающих рамок и классов объектов. Это обеспечивает высокую скорость работы при сохранении точности.

Архитектура YOLOv8 состоит из трёх основных компонентов: магистральной сети (backbone) на основе CSPDarknet для извлечения визуальных признаков из изображения; промежуточного блока (neck) FPN + PAN для объединения признаков на разных масштабах; разделённой выходной части (head), в которой задачи классификации и регрессии координат решаются независимо. Ключевой особенностью YOLOv8 является отсутствие предопределённых якорных рамок, что упрощает настройку модели.

Используется модель YOLOv8m, обеспечивающая баланс между точностью и вычислительной эффективностью. Детекция ведётся по четырём классам: игрок, вратарь, судья, мяч.

Для многообъектного трекинга применяется алгоритм ByteTrack [1]. В его основе лежит фильтр Калмана, предсказывающий положение объекта на следующем кадре по текущему состоянию. Ключевая особенность ByteTrack — двухуровневое сопоставление детекций с треками. На первом этапе уверенные детекции (с высоким значением параметра уверенности) сопоставляются с предсказанными положениями треков на основе метрики пересечения по объединению. На втором этапе повторно обрабатываются оставшиеся, менее уверенные детекции. Такой подход позволяет сохранять идентичность объектов при частичном или полном перекрытии одного объекта другим.

2. Датасет и обучение модели

Для дообучения модели используется набор данных `football-players-detection` с платформы Roboflow [2], содержащий 255 размеченных изображений футбольных матчей. Разметка включает четыре класса объектов. Набор данных разделён на обучающую (204 изображения), валидационную (38) и тестовую (13) выборки.

Обучение проводится методом переноса обучения: предобученная на наборе данных COCO (330 000 изображений, 80 классов) модель YOLOv8m дообучается на футбольном наборе данных. Данный подход позволяет эффективно использовать выученные низкоуровневые визуальные признаки и достигать высокого качества при ограниченном объёме обучающих данных.

Обучение проведено в среде Google Colab на графическом процессоре NVIDIA Tesla T4 (16 ГБ видеопамяти) в течение 200 эпох при входном разрешении 640×640 пикселей. Для повышения обобщающей способности модели применяются аугментации: объединение четырёх изображений, наложение двух изображений, случайные аффинные преобразования и модификации в цветовом пространстве HSV.

3. Результаты экспериментов

Качество детекции оценивается стандартными метриками. Метрика пересечения по объединению (IoU) определяется как отношение площади пересечения предсказанной и эталонной рамок к площади их объединения:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|},$$

где A — предсказанная ограничивающая рамка, B — эталонная разметка.

На основе пересечения по объединению вычисляются точность (доля верных детекций среди всех предсказанных), полнота (доля найденных объектов среди всех реальных) и средняя усреднённая точность mAP — площадь под кривой точность-полнота, усреднённая по всем классам.

Результаты на валидационной выборке представлены в табл. 1.

Таблица 1. Метрики качества детекции YOLOv8m.

Класс	Точность	Полнота	mAP50	mAP50-95
игрок	0,974	0,981	0,982	0,766
вратарь	0,965	0,776	0,969	0,713
судья	0,952	0,942	0,925	0,644
мяч	0,948	0,400	0,448	0,173
Среднее	0,960	0,775	0,831	0,574

Наилучшие результаты достигнуты для класса «игрок»: $mAP50 = 0,982$ на 754 экземплярах. Класс «мяч» показывает наихудший результат ($mAP50-95 = 0,173$), что объясняется малым размером объекта (порядка 10×10 пикселей при входном разрешении) и размытием при быстром движении. Скорость работы модели на Tesla T4 составляет примерно 19 кадров в секунду (8,4 мс предварительная обработка, 39,5 мс детекция, 4,7 мс заключительная обработка).

Тестирование полного программного конвейера (детекция, трекинг и визуализация) проведено на 30-секундном видеофрагменте матча Бельгия – Россия (UEFA EURO 2020), содержащем 750 кадров при частоте 25 кадров в секунду. Система стабильно обнаруживает 23 объекта на кадре: 22 полевых игрока и мяч. Алгоритм ByteTrack обеспечивает устойчивое сохранение уникальных идентификаторов объектов между кадрами. Потеря идентичности наблюдается при длительных перекрытиях объектов (более 1 секунды).

Дополнительно исследована возможность классификации игроков по командам на основе цвета формы. Из ограничивающей рамки каждого игрока выделяется область торса, выполняется фильтрация зелёных пикселей (маска в пространстве HSV), вычисляется медианный цвет оставшихся пикселей, после чего применяется алгоритм кластеризации k -средних с $k = 2$ для разделения на две группы. Эксперименты показали, что данный подход неустойчив при низком разрешении видео: область торса содержит слишком мало пикселей, тени и блики искажают цвет, кластеризация нестабильна. Модуль не включён в окончательную версию системы и рассматривается как направление дальнейшего развития.

Заключение

В работе разработана модульная система автоматической детекции и трекинга игроков футбольного матча на видеозаписях. Модель YOLOv8m дообучена на специализированном наборе данных и достигла значения $mAP50 = 0,831$ при значении точности 0,960. Реализован полный программный конвейер обработки видео с использованием алгоритма ByteTrack для многообъектного трекинга. Экспериментальная оценка на реальном видео матча подтвердила работоспособность системы.

В качестве направлений дальнейшего развития можно выделить: использование нейронных сетей повторной идентификации для классификации команд; обработку видео более высокого разрешения для улучшения детекции мяча; построение тепловых карт перемещений игроков на основе данных трекинга.

Список литературы

- [1] ByteTrack: Multi-Object Tracking by Associating Every Detection Box / Y. Zhang, P. Sun, Y. Jiang [et al.] // Computer Vision — ECCV 2022. LNCS, vol. 13682. — Cham : Springer, 2022. — P. 1–21. — DOI: https://doi.org/10.1007/978-3-031-20047-2_1.
- [2] Football-Players-Detection : открытый набор данных / Roboflow Universe. — URL: <https://universe.roboflow.com/roboflow-jvuqo/football-players-detection-3zvbc> (дата обращения: 01.12.2025). — Загл. с титул. экрана.
- [3] Jocher, G. Ultralytics YOLOv8 : сайт / G. Jocher, A. Chaurasia, J. Qiu. — 2023. — URL: <https://github.com/ultralytics/ultralytics> (дата обращения: 01.12.2025). — Загл. с титул. экрана.

Библиографическая ссылка

Гаврилов, А. Д. Детекция и трекинг игроков футбольного матча с использованием методов компьютерного зрения // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 34–38.

Сведения об авторах

ГАВРИЛОВ АНТОН ДМИТРИЕВИЧ

Студент магистратуры

Направление «Прикладная математика и информатика»

УДК 004.89

Рекомендательная система на основе музыкальных предпочтений пользователя

Глинский М. С.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий Е. Н. Гришиной

Аннотация. В статье описывается подход к музыкальным рекомендациям на основе датасета Yambda, объединяющий коллаборативную и контентную фильтрацию, выделение доминирующих кластеров и темпоральное затухание интереса. Замена поиска похожих пользователей семантическим подбором треков снизила вычислительные затраты, а экспоненциальное взвешивание событий и персонализированное ограничение доли незнакомого контента повысили точность: Recall@10 достиг 0,0576 (против 0,0373), NDCG@100 — 0,1605 (против 0,0934). Предложенная гибридная стратегия обеспечивает баланс качества, разнообразия и быстродействия.

Ключевые слова: рекомендательные системы, коллаборативная фильтрация, кластеризация, темпоральная динамика предпочтений.

Введение

В стриминговых сервисах с миллионами треков персонализированные рекомендации критичны для удержания аудитории и навигации. Специфика музыкального контента — преобладание неявной обратной связи (прослушивания) над явной (лайки/дизлайки) и временная динамика предпочтений. Это требует алгоритмов, одновременно извлекающих информацию из разреженных явных оценок, опирающихся на неявные сигналы и адаптирующихся к смещению вкусов. В работе предлагается гибридный подход, поэтапно разбиваемый на реальных данных Яндекс.Музыки (датасет Yambda, около 48 млн записей с векторными представлениями треков). Базовый двухэтапный алгоритм для альбомов (агрегация активности +

поиск похожих пользователей) адаптирован к уровню треков: коллаборативный отбор кандидатов заменён семантической близостью эмбедингов, введены временное затухание значимости событий и персонализированное управление долей незнакомого контента. Результат — кратный рост точности при приемлемых вычислительных затратах.

1. Описание данных

Исследование выполнено на минимальном разбиении датасета Yambda (Яндекс.Музыка), содержащем 47 790 449 записей о реальных действиях пользователей. Задействованы четыре таблицы: событий, векторных представлений (7 721 749), альбом–трек (8 653 783) и исполнитель–трек (9 270 506). Таблица событий фиксирует для каждого действия идентификаторы пользователя и трека, временную метку, признак органичности, тип события, долю прослушанной длины и полную длительность трека. Векторные представления доступны лишь для 46 219 700 событий; отсутствующие заполнены нулевыми векторами, что позволило сохранить всю выборку. Распределение типов событий характерно для музыкального сервиса: 46 467 212 прослушиваний, 881 456 лайков, 107 776 дизлайков, 312 972 отмен лайка и 21 033 отмены дизлайка. Таким образом, неявная обратная связь резко доминирует над явной. Органические и неорганические события представлены почти поровну (24,9 и 22,9 млн). Временной охват датасета — около 300 суток. Активность пользователей и треков отличается сильной правосторонней асимметрией: основная масса пользователей генерирует мало событий, а немногочисленное ядро обладает длинными историями. Более 90 % треков имеют менее 100 прослушиваний за весь период, в то время как узкая группа сверхпопулярных композиций аккумулирует непропорционально большую долю воспроизведений.

2. Рекомендация альбомов

Для построения множества альбомов-кандидатов применён двухэтапный подход: агрегирование оценок прослушанных альбомов и коллаборативный отбор по похожим пользователям. На первом этапе для каждой пары «пользователь – альбом» вычисляются показатели:

суммарные органические прослушивания ($is_organic$), общий накопленный процент прослушивания ($played_ratio_pct$), число прослушиваний ($number$), доля прослушанных треков альбома (pct_album), средний процент прослушивания на событие ($avg_played_ratio_pct$), процент уникальных прослушанных треков (num_unique_tracks) и оценка $coff_like$:

$$buf = likes - 0,25 \text{ unlikes} - dislikes + \min(0,05 \text{ is_organic}, 0,3),$$

$$coff_like = \min\left(\max\left(\frac{buf}{tracks}, -1\right), 1\right).$$

Сырые показатели нормализуются экспоненциальным сглаживанием:

$$is_organic_norm = 1 - \exp(-is_organic),$$

$$avg_norm = 1 - \exp\left(-\frac{avg_played_ratio_pct}{45}\right),$$

$$pct_norm = 1 - \exp\left(-\frac{pct_album}{0,05}\right),$$

$$unique_norm = \frac{num_unique_tracks}{100},$$

$$coff_norm = \frac{coff_like + 1}{2}.$$

Итоговая оценка $weight$ вычисляется как взвешенная сумма:

$$weight = 0,25 \cdot is_organic_norm + 0,15 \cdot pct_norm + \\ + 0,3 \cdot avg_norm + 0,05 \cdot unique_norm + 0,25 \cdot coff_norm.$$

Весовые коэффициенты подобраны эмпирически, исходя из относительной важности органического интереса, глубины ознакомления с альбомом и его содержательного соответствия профилю.

По значению $weight$ альбомы делятся на четыре категории: «наиболее понравившиеся» ($> 0,6$), «любимые» ($0,2-0,6$), «доизучения» ($-0,3-0,2$) и «игнорирования» (остальные).

На втором этапе для пользователя ищутся похожие. Близость пары определяется по совместно прослушанным альбомам. Для pct_album и $avg_played_ratio_pct$ вычисляется косинусное сходство.

Дополнительно учитываются: доля альбомов, прослушанных не по рекомендации (`org_count`), сходство по проценту уникальности вкуса:

$$\text{sim_unique} = \exp(-0,001 \cdot \Delta\text{num_unique_tracks}).$$

Третий фактор — линейная корреляция Пирсона между векторами коэффициентов оценок (`coff_like`) пользователей по совместно прослушанным альбомам; обозначим её как `correlation`.

Итоговый вес сходства `weight_sim` вычисляется как взвешенная сумма перечисленных компонент:

$$\text{weight_sim} = 0,25 \cdot \text{pct_album} + 0,2 \cdot \text{avg_played_ratio_pct} + \\ + 0,15 \cdot \text{sim_unique} + 0,3 \cdot \text{correlation} + 0,1 \cdot \text{org_count}.$$

Коэффициенты в `weight_sim` подобраны так, что наибольший вклад дают согласованность глубины прослушивания (первые два слагаемых) и корреляция симпатий, а показатели органического потребления и нишевости играют вспомогательную роль.

Отобранные альбомы преобразуются в упорядоченный дневной плейлист с учётом почасового распределения пользовательской активности. Треки чередуются по категориям альбомов для поддержания разнообразия и внимания. При завершении и возобновлении сессии генерируется новый плейлист, из которого исключаются уже прослушанные за день треки, что предотвращает повторы.

Полный пересчёт матрицы сходства «пользователь–пользователь» и оценок для всех пар «пользователь–альбом» на десятках миллионов событий требует значительных временных затрат, даже при распараллеливании.

3. Рекомендация треков

Перенос подхода с уровня альбомов на уровень отдельных треков не потребовал принципиального изменения алгоритмической схемы и был выполнен путём замены объекта агрегации. Для каждого пользователя на основе таблицы событий формировались агрегированные показатели по каждому прослушанному треку: количество органических взаимодействий (`is_organic`), суммарный процент прослушанного материала (`played_ratio_pct`), число сеансов

прослушивания (`number`), последняя оценка трека пользователем (`end_event_type`), доля трека в общем объёме прослушанного пользователем контента (`pct_track`), а также средний процент прослушивания `avg_played_ratio_pct`.

Оценка трека считается так:

$$\begin{aligned} \text{is_organic_norm} &= 1 - \exp(-\text{is_organic}), \\ \text{avg_norm} &= 1 - \exp\left(-\frac{\text{avg_played_ratio_pct}}{45}\right), \\ \text{pct_norm} &= 1 - \exp\left(-\frac{\text{pct_album}}{0.05}\right), \\ \text{coff_norm} &= \frac{\text{coff_like} + 1}{2}. \end{aligned}$$

Итоговая оценка `weight` вычисляется как взвешенная сумма:

$$\begin{aligned} \text{weight} &= 0,25 \cdot \text{is_organic_norm} + 0,18 \cdot \text{pct_norm} + \\ &\quad + 0,32 \cdot \text{avg_norm} + 0,25 \cdot \text{coff_norm}. \end{aligned}$$

Формула для поиска похожих пользователей также немного изменилась, была убрана нормировка от 0 до 1 за ненадобностью:

$$\begin{aligned} \text{weight_sim} &= 0,25 \cdot \text{pct_album} + 0,2 \cdot \text{avg_played_ratio_pct} + \\ &\quad + 0,15 \cdot \text{sim_unique} + 0,3 \cdot \text{correlation} + 0,1 \cdot \text{org_count}. \end{aligned}$$

Данный алгоритм в среднем показал результаты хуже, чем некоторые классические алгоритмы, такие как `itemKNN`, `SASRec`, поэтому необходимо было улучшить алгоритм как с точки зрения оптимизации, так и качества.

4. Алгоритм рекомендации треков на основе векторных представлений и доминирующих кластеров

Для снижения вычислительной сложности поиск похожих пользователей был заменён отбором по эмбедингам: для каждого трека заранее отобраны 100 ближайших по косинусному сходству. Кандидатами для пользователя становятся треки из объединения таких

множеств для всех прослушанных им позиций, исключая уже известные. Оценка кандидата вычисляется как сумма по прослушанным трекам, для которых он попал в сотню ближайших, произведений weight трека на косинусное сходство эмбедингов. Такой контентно-ориентированный отбор радикально сократил время построения рекомендаций при сохранении качества и повысил разнообразие и охват каталога.

Следующим шагом, нацеленным на дополнительное повышение точности, треки были кластеризованы в том же пространстве эмбедингов (полученные кластеры не отождествляются с музыкальными жанрами). Для каждого пользователя строилось распределение прослушиваний по кластерам и выделялись доминирующие кластеры, покрывающие 60 % всей его активности. При формировании итогового списка предпочтение отдавалось трекам именно из этих кластеров. Модификация дала незначительный прирост точности, но одновременно снизила разнообразие, сместив фокус в сторону уже знакомого пользователю музыкального материала.

5. Алгоритм рекомендаций треков на основании локальных трендов

Дальнейший анализ выявил две проблемы: статичность оценки треков и несбалансированность знакомого и незнакомого контента. Для учёта временной динамики введено экспоненциальное затухание значимости событий с полураспадом 3,5 дня:

$$e^{-\lambda \Delta t}, \quad \text{где } \lambda = \frac{\ln 2}{3,5}, \quad \Delta t = \text{время, прошедшее с момента события.}$$

Это адаптирует веса к смещению интересов.

Баланс скорректирован ограничением доли незнакомых треков: для давних пользователей $< 5\%$, для новых — выше, стимулируя исследование каталога. Совокупность модификаций привела к кратному росту качества без увеличения времени вычислений и при незначительном снижении разнообразия. Итоговые метрики (в скобках — базовые решения создателей датасета): $\text{Recall}@10 = 0,0576 (0,0373)$, $\text{Recall}@100 = 0,2196 (0,1297)$, $\text{NDCG}@10 = 0,1201 (0,0781)$, $\text{NDCG}@100 = 0,1605 (0,0934)$. Таким образом, темпоральное затухание и персонализированное управление новизной обеспечили

существенное превосходство над базовыми методами по точности ранжирования при сохранении эксплуатационной производительности.

Заключение

Предложен и усовершенствован гибридный подход к музыкальным рекомендациям, объединяющий коллаборативную фильтрацию, контентный анализ эмбедингов и учёт временной динамики. Замена поиска похожих пользователей семантическим отбором по векторным представлениям значительно снизила вычислительную сложность без потери точности. Введение временного затухания значимости событий и персонализированного ограничения доли незнакомого контента обеспечило кратный прирост ключевых метрик при сохранении разнообразия и быстродействия. Результаты подтверждают эффективность стратегии, адаптирующейся к долгосрочным вкусам и текущим интересам, и открывают путь к улучшениям за счёт более тонкого моделирования сессий и многомодальных признаков.

Список литературы

- [1] Менькин, А. В. Разработка музыкальной рекомендательной системы на основе обработки метаданных контента // Вестник НГУ. Серия: Информационные технологии. — 2019. — Т. 17, № 3. — С. 43–60. — DOI: <https://doi.org/10.25205/1818-7900-2019-17-3-43-60>.
- [2] Segaran, T. Programming Collective Intelligence. — Sebastopol, CA : O'Reilly Media, 2007. — 334 p. — URL: https://github.com/newhandLiu/collective_intelligence/blob/master/programming_collective_intelligence.aug.2007.pdf.

Библиографическая ссылка

Глинский, М. С. Рекомендательная система на основе музыкальных предпочтений пользователя // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 40–46.

Сведения об авторах

Глинский Максим Станиславович

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 330.4

Анализ дивидендной торговой стратегии на фондовом рынке

Ермаков А. Э.

Тверской государственный университет

Рекомендовано к публикации профессором кафедры
математической статистики и системного анализа Г. М. Соломахой

Аннотация. В данной статье описана новая дивидендная торговая стратегия, а также модель оптимизации получения максимальной прибыли и доходности с учётом и без учёта выплаты дивидендов. С использованием технического анализа выбираются оптимальные моменты и даты для входа и выхода (покупка и продажа акций), а также делается расчет предполагаемой прибыли, которую инвестор получит после завершения операций. Представленная модель отбирает наилучшие варианты комбинаций компаний, в акции которых целесообразно вкладывать деньги и получать с этих комбинаций перемещений денежных средств максимальную прибыль. Приведен пример использования модели.

Ключевые слова: дивиденды, стратегия, оптимизация, максимальная прибыль, максимальная доходность.

Введение

В настоящее время оптимизация получения прибыли на фондовом рынке остаётся важным вопросом для многих инвесторов, так как в условиях постоянных изменений на рынке, в частности, из-за понижения ключевых ставок, что, как правило, влечёт за собой понижение и процентных ставок по банковским вкладам. При этом дивиденды остаются до сих пор одним из основных источников стабильного пассивного дохода [1].

Разработка новых торговых стратегий позволяет оптимизировать получение прибыли инвесторам. В данной статье представлена модель, которая рассчитывает прибыль для комбинаций компаний, а затем выбирает наилучшие варианты для вложений, то есть такие, где была зафиксирована максимальная прибыль.

1. Технический анализ фондового рынка

Технический анализ предназначается для изучения биржевых курсов ценных бумаг эмитента (организации, которая регистрирует и выпускает ценные бумаги или другие финансовые активы для привлечения капитала с целью реализации своих основных задач) и нужен чаще всего для инвестиций на короткие временные интервалы. Методы технического анализа делятся на графические и количественные. Последние представляют из себя математические индикаторы, формирующие торговые сигналы, которые могут как появляться, так и исчезать в процессе торгов.

Очень часто на практике применяется индикатор под названием скользящее среднее значение. Основная задача такого инструмента — отображать среднее значение цен закрытия в течение определённого количества интервалов времени (дней, часов, недель и т. д.). Это позволяет определить время начала новой тенденции, а также предупредить о ее завершении или повороте. При подборе моментов для покупки или продажи акций используются экспоненциальные скользящие средние, которые вычисляются по формуле

$$EMA_t = EMA_{t-1} + k \times (P_t - EMA_{t-1}), \quad (1)$$

где:

- EMA_t — текущее значение экспоненциальной средней;
- EMA_{t-1} — предыдущее значение экспоненциальной средней;
- P_t — текущее значение цены;
- k — коэффициент сглаживания.

Коэффициент сглаживания вычисляется по формуле

$$k = \frac{2}{n + 1}, \quad (2)$$

где n — порядок средней.

Скользящие средние, особенно экспоненциальные, позволяют инвестору определять течения тенденций, чтобы проводить биржевую игру в нужном направлении.

Стохастический осциллятор достаточно точно оценивает краткосрочную тенденцию в условиях бокового тренда, существующего в более крупном масштабе. Если торговля идёт в широком коридоре цен, то появляется сигнал о переходе в фазу движения от одного его края к другому. В частности, используется показатель K , который вычисляется по следующей формуле:

$$K = \frac{P_t - L_{min}}{H_{max} - L_{min}} \times 100, \quad (3)$$

где:

- P_t — текущее значение цены закрытия;
- H_{max} — значение максимальной цены;
- L_{min} — значение минимальной цены.

Стохастический осциллятор мгновенно реагирует на любое изменение цены, поэтому сигналы являются опережающими, и они могут помочь инвестору раньше принять решение.

Индекс относительной силы измеряет силу рынка через изменения его цен закрытия. Он может подавать сигналы с опережением цен. Рассчитывается такой индекс по следующей формуле:

$$RSI = 100 - \frac{100}{1 + RS}, \quad (4)$$

где RS — относительная сила, численно равная отношению среднего значения повышения цен закрытия к среднему значению понижения цен закрытия.

Индекс RSI меняется от 0 до 100. Подъём индекса над верхней пограничной линией показывает силу быков и вступление рынка в состояние перекупленности (более 70 %). Если индекс падает на нижнюю пограничную линию, это отражение силы медведей и вступления рынка в состояние перепроданности (менее 30 %). Самые надёжные сигналы индекс подаёт при расхождении с ценами. Самые сильные из них возникают при достижении цен новых максимумов или минимумов, а RSI при этом не пересекает пограничные линии.

Инструментов для проведения технического анализа на самом деле довольно много [2], [3], причём не надо ограничиваться одним

из них, так как это может привести к неточным и недостоверным результатам, что скажется на размере прибыли. Методы технического анализа надо комбинировать таким образом, чтобы они могли подтверждать сигналы друг друга и позволять инвестору принимать правильные решения.

2. Описание модели оптимизации получения максимальной прибыли

При разработке новой дивидендной торговой стратегии была создана модель, цель которой — оптимизация получения прибыли инвесторам. После хронологической сортировки дат собраний компаний, выбранных из списка голубых фишек фондового рынка, происходит перебор всех возможных вариантов скачков. Модель осуществляет расчёт прибыли без учёта и с учётом выплаты дивидендов в случае одной компании (без скачков) и задаётся следующими формулами:

$$\max_{1 \leq i \leq N} (P_i[0, t_i] + P_i[t_i, T]), \quad (5)$$

$$\max_{1 \leq i \leq N} (P_i[0, t_{i_d}] + P_i[t_{i_d}, T] + D_i). \quad (6)$$

где:

- N — количество компаний из списка;
- i — номер компании в списке;
- P_i — прибыль, полученная с i -й компании;
- 0 — начало года;
- t_i — время закрытия реестра i -й компании с учётом входа вне дивидендного окна (без учёта дивидендов);
- t_{i_d} — время закрытия реестра i -й компании с учётом входа в дивидендное окно (с учётом дивидендов);
- T — конец года;
- D_i — сумма дивидендов, полученная с i -й компании.

Модель, которая предлагает получение прибыли без учёта и с учётом дивидендов на примере скачков по временам выплаты дивидендов компаниями, задаётся следующими формулами:

$$\max_{1 \leq i \leq N} (P_1[0, t_1] + P_2[t_1, t_2] + \dots + P_i[t_{i-1}, T]), \quad (7)$$

$$\max_{1 \leq i \leq N} (P_1[0, t_{1_d}] + P_2[t_{1_d}, t_{2_d}] + \dots + P_i[t_{(i-1)_d}, T]) + \sum_{i=1}^N D_i. \quad (8)$$

Прибыль вычисляется как для каждой компании по отдельности, так и для всех вариантов скачков. Далее выбирается такая компания или последовательность компаний, для которых была зафиксирована максимальная прибыль.

Для случая получения прибыли с учётом дивидендов на примере нескольких компаний реализована схема, которая приведена на рис. 1. После сортировки компаний по датам собраний инвестор входит в первую компанию, покупая акции в начальный период времени. Позиция закрывается с продажей акций, время которой рассчитывается при помощи технического анализа, но инвестор делает это после дивидендного окна, так как если это осуществить до начала формирования списков держателей акций и закрытия реестра, то дивиденды после продажи акции он получить не сможет. После фиксации даты и момента выхода из первой компании инвестор проверяет, можно ли ему осуществить скачок в следующую компанию или нет. Если момент выхода попал в диапазон дивидендного окна следующей компании, он переходит в неё, и подбираются новые оптимальные моменты для входа и выхода. Если момент выхода выпал на период, когда собрание акционеров ещё не началось, инвестор ожидает в текущей компании. Если же момент выхода выпал на период после дивидендного окна, это означает, что осуществить скачок в следующую по хронологии компанию не получится и надо переместиться в следующую компанию из списка. Аналогично осуществляется переход из второй компании в третью, из третьей в четвёртую и так далее. В итоге будет получена прибыль для всех возможных вариантов скачков, учитывая ситуации, когда инвестор может не успеть войти из-за опозданий в период дивидендных окон.

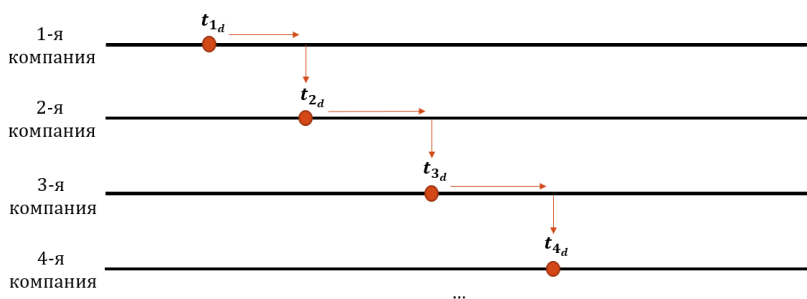


Рис. 1. Схема модели скачков по временам выплаты дивидендов.

3. Пример

Рассмотрим пример нахождения максимальной прибыли и лучших комбинаций вложений 1 млн рублей в пять разных компаний из списка голубых фишек фондового рынка Московской Биржи с учётом и без учёта выплаты дивидендов, чтобы сравнить полученные результаты и выявить эффективность предложенной дивидендной стратегии.

В качестве входных данных возьмём сведения о выплатах дивидендов компаниями, представленные в табл. 1. Отбор выполнен посредством учёта стабильности и регулярности выплат за прошлые годы, включая кризисные, например, период пандемии Covid-19. Компании упорядочены по датам собраний. После формирования портфеля отберём компании из разных отраслей для анализа.

Таблица 1. Сортировка компаний по датам собраний за 2025 год.

Ранг	Компания	Дата собрания	Дата отсечки	Дата выплат дивидендов
1	Полюс	11.03.2025	25.04.2025	28.05.2026
2	Мосбиржа	26.05.2025	10.07.2025	08.09.2025
3	Сбербанк	03.06.2025	18.07.2025	27.07.2026
4	Роснефть № 1	05.06.2025	20.07.2025	22.01.2026
5	Ростелеком	29.06.2025	13.08.2025	12.10.2025

Далее подбираются при помощи методов технического анализа оптимальные моменты для входа и выхода, чтобы определить, какую прибыль инвестор может получить. Сначала сформируем табл. 2.

Таблица 2. Лучшие моменты входа/выхода без учёта дивидендов.

Дата входа	Дата выхода	Цена покупки	Цена продажи
25.04.2025	27.04.2025	1832,6	1854,6
10.07.2025	15.08.2025	168,93	187,66
18.07.2025	15.08.2025	300,45	319,4
18.07.2025	23.07.2025	406	424,7
14.08.2025	15.08.2025	67,48	71,3

Применяя формулы (5) и (7), рассчитываются значения прибыли без учёта дивидендов для всех комбинаций компаний. Затем отбираются среди всех вариантов наилучшие комбинации, то есть такие, где была зафиксирована максимальная прибыль, и эти значения отображаются на графике, изображённом на рис. 2.

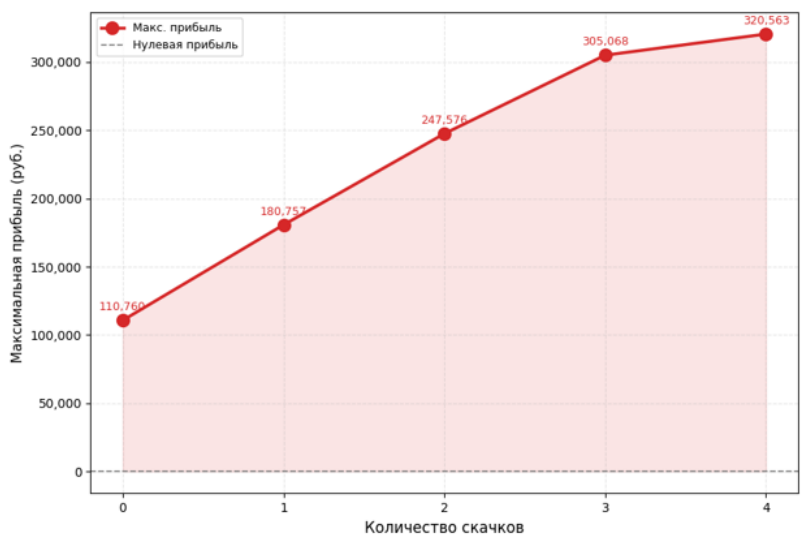


Рис. 2. График значений прибыли без учёта дивидендов.

Теперь подберём при помощи методов технического анализа оптимальные моменты для покупки и продажи акций, представленные в табл. 3, попутно руководствуясь схемой на рис. 1.

Таблица 3. Лучшие моменты входа/выхода с учётом дивидендов.

Дата входа	Дата выхода	Цена покупки	Цена продажи	Размер дивиденда
07.04.2025	23.04.2025	1666,8	1948	70,85
26.05.2025	09.07.2025	181,45	194,35	19,57
14.07.2025	17.07.2025	304,47	328,46	37,64
14.07.2025	23.07.2025	395,05	426,4	11,56
30.06.2025	12.08.2025	56,53	72,44	2,71

Применяя формулы (6) и (8), рассчитываются значения прибыли с учётом дивидендов для всех комбинаций компаний. Затем отбираются наилучшие комбинации и соответствующие значения максимальной прибыли отображаются на графике, изображённом на рис. 3.

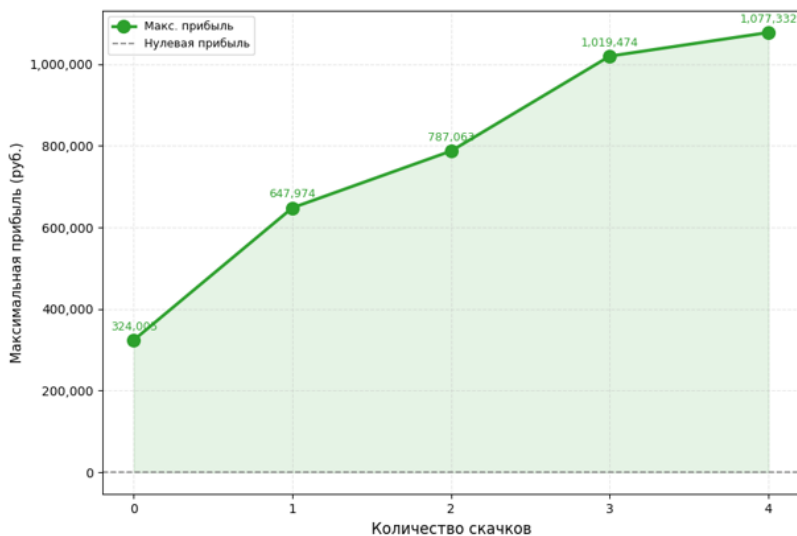


Рис. 3. График значений прибыли с учётом дивидендов.

Результаты анализа эффективности новой дивидендной стратегии приведены в табл. 4 и табл. 5.

Таблица 4. Результаты анализа стратегии без учёта дивидендов.

Лучшая комбинация скачков	Максимальная доходность
Мосбиржа	+11,08 %
Сбербанк – Мосбиржа	+18,08 %
Мосбиржа – Ростелеком – Сбербанк	+24,76 %
Роснефть № 1 – Мосбиржа – Ростелеком – Сбербанк	+30,51 %
Сбербанк – Роснефть № 1 – Ростелеком – Полус – Мосбиржа	+32,06 %

Таблица 5. Результаты анализа стратегии с учётом дивидендов.

Лучшая комбинация скачков	Максимальная доходность
Ростелеком	+32,4 %
Полус – Ростелеком	+64,8 %
Полус – Сбербанк – Ростелеком	+78,71 %
Полус – Мосбиржа – Сбербанк – Ростелеком	+101,95 %
Полус – Мосбиржа – Сбербанк – Роснефть № 1 – Ростелеком	+107,73 %

Сравнивая показатели доходности в обеих таблицах, можно сделать вывод о том, что максимальная доходность с учётом дивидендов превосходит максимальную доходность без учёта дивидендов примерно в 3–3,5 раза по каждой разновидности комбинаций компаний. Это иллюстрирует эффективность предложенной дивидендной стратегии.

Заключение

В данной работе была разработана модель максимизации прибыли с учётом выплаты дивидендов компаниями, в которые вкладываются денежные средства. Приведённый пример показывает, что применение данной модели позволяет получить наилучшие комбинации последовательного перемещения капитала в несколько компаний

в случаях со скачками с учётом и без учёта дивидендов, а также повысить максимальную доходность инвестору.

Список литературы

- [1] Кери, И. Т. Особенности дивидендных выплат в российских компаниях / И. Т. Кери, Е. И. Громова, Н. В. Синельщикова // Современные тенденции в экономике и управлении: новый взгляд. — 2013. — № 20. — С. 173–179. — URL: <https://www.elibrary.ru/item.asp?id=20882250>.
- [2] Элдер, А. Как играть и выигрывать на бирже: Психология. Технический анализ. Контроль над капиталом / А. Элдер. — 4-е изд., перераб. и доп. — М. : Альпина Бизнес Букс, 2007. — 472 с. — URL: https://winoptionru.online/userfiles/books/kak_igrat_i_vyigryvat_na_birzhe.pdf.
- [3] Яковлева, Е. К. Сравнение различных методов технического анализа при принятии решений об инвестициях / Е. К. Яковлева // Финансовая аналитика: проблемы и решения. — 2015. — № 11 (245). — С. 55–63. — URL: <https://www.fin-izdat.ru/journal/fa/detail.php?ID=65137>.

Библиографическая ссылка

Ермаков, А. Э. Анализ дивидендной торговой стратегии на фондовом рынке // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 48–57.

Сведения об авторах

ЕРМАКОВ АЛЕКСАНДР ЭДУАРДОВИЧ
Студент магистратуры
Направление «Прикладная информатика»

УДК 330.43, 004.8

Построение эконометрических моделей оценки стоимости подержанных автомобилей

Ивашин М. Г.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа С. В. Архиповым

АННОТАЦИЯ. В статье представлены результаты эконометрического анализа факторов ценообразования на вторичном автомобильном рынке России. На основе выборки свыше 1,8 млн объявлений с платформы Drom.ru построены сегментированные регрессионные модели для шести типов кузова с применением двух методологических подходов: пошаговой МНК-регрессии и регуляризации Elastic Net. Установлено, что ключевыми детерминантами стоимости выступают возраст и пробег с нелинейным характером влияния, мощность двигателя, брендовое позиционирование, тип трансмиссии, а также факторы риска сделки. Сравнительный анализ подходов показал, что регуляризация обеспечивает более устойчивые оценки и улучшает прогностическую точность, особенно в малых выборках.

КЛЮЧЕВЫЕ СЛОВА: гедонистическое ценообразование, вторичный рынок автомобилей, множественная линейная регрессия, Elastic Net, регуляризация, пошаговый отбор переменных.

Введение

Рынок подержанных автомобилей представляет собой сложную экономическую систему, в которой итоговая цена формируется под воздействием множества разнородных факторов: объективных технических характеристик и степени износа, субъективных предпочтений покупателей, региональной специфики и макроэкономической конъюнктуры. Точная оценка рыночной стоимости транспортного средства является критически важной задачей для всех участников

рынка — от частных покупателей и продавцов до профессиональных игроков: банков, страховых компаний и дилерских центров.

Эконометрическое моделирование предоставляет научно обоснованный инструментарий для решения данной задачи. Применение теории гедонистического ценообразования, рассматривающей товар как совокупность полезных атрибутов, позволяет формализовать зависимость цены от характеристик объекта. Вместе с тем высокая размерность признакового пространства и наличие тесной корреляции между характеристиками автомобилей обуславливают необходимость применения как классических регрессионных методов, так и современных алгоритмов регуляризации.

Настоящее исследование направлено на построение и сравнительную оценку эконометрических моделей стоимости подержанных автомобилей с использованием двух подходов — пошаговой МНК-регрессии и регуляризации Elastic Net — применительно к сегментированному рынку по типу кузова транспортного средства.

Целью исследования является проведение эконометрического анализа и построение моделей оценки стоимости подержанных автомобилей с использованием методов регрессионного анализа и регуляризации, а также сравнительная оценка их прогностических свойств в разрезе рыночных сегментов.

Для достижения поставленной цели решаются следующие задачи: формирование и предобработка репрезентативной выборки рыночных объявлений; спецификация лог-линейных регрессионных моделей с учётом нелинейных эффектов и взаимодействий; построение моделей методом пошагового отбора переменных и методом Elastic Net; сравнительный анализ полученных спецификаций по критериям точности, устойчивости и интерпретируемости.

1. Постановка задачи

Задача прогнозирования стоимости автомобиля формализуется как задача регрессионного анализа [3]: требуется найти функциональную зависимость между ценой и набором объясняющих переменных. Зависимая переменная y — рыночная цена автомобиля — отражает результат взаимодействия всех рыночных факторов и является

случайной величиной. Объясняющие переменные

$$x = (x_1, x_2, \dots, x_m)$$

описывают технические, качественные и рыночные характеристики объекта.

В качестве базовой принята модель множественной линейной регрессии [2]:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon,$$

где β_0 — константа, β_1, \dots, β_k — коэффициенты модели, k — число факторов, ε — случайная ошибка. В матричной форме:

$$y = X\beta + \varepsilon.$$

Оценка параметров осуществляется методом наименьших квадратов (МНК) путём минимизации суммы квадратов остатков, что при выполнении предпосылок классической линейной регрессионной модели (КЛРМ) обеспечивает наилучшие линейные несмещённые оценки — теорема Гаусса–Маркова [1]:

$$\hat{\beta} = (X^T X)^{-1} X^T y.$$

Поскольку распределение цен на вторичном рынке характеризуется выраженной правосторонней асимметрией, в качестве зависимой переменной используется натуральный логарифм цены $\ln(P_i)$. Это позволяет интерпретировать коэффициенты модели как приближённые относительные изменения стоимости при единичном изменении соответствующего фактора:

$$\ln(P_i) = \beta_0 + \sum \beta_j x_{ij} + \varepsilon_i.$$

При наличии мультиколлинеарности и высокой размерности признакового пространства МНК-оценки теряют устойчивость. Для её повышения применяется регуляризация Elastic Net [4, 5], которая модифицирует функцию потерь, добавляя штрафные члены L_1 и L_2 :

$$\min_{\beta} \left\{ \frac{1}{2n} \|y - X\beta\|_2^2 + \lambda \rho \|\beta\|_1 + \frac{\lambda(1 - \rho)}{2} \|\beta\|_2^2 \right\},$$

где λ — параметр регуляризации, $\rho \in [0; 1]$ — соотношение между L_1 - и L_2 -штрафами. L_1 -компонента обеспечивает разреженность решения и отбор признаков, L_2 -компонента стабилизирует оценки при коллинеарности.

2. Данные и переменные

Эмпирическая база исследования сформирована на основе данных объявлений о продаже подержанных автомобилей с платформы Drom.ru, агрегированных на ресурсе Kaggle. После процедур предобработки совокупный объём выборки превысил 1,8 млн наблюдений. В рамках предобработки из анализа исключены автомобили с пробегом менее 10 000 км, объёмом двигателя менее 1 литра, с электрическими и газовыми двигателями, а также объявления, размещённые после 2024 года. Ценовой диапазон ограничен интервалом от 150 тыс. до 50 млн рублей.

Анализ описательной статистики выявил выраженную правостороннюю асимметрию ценового распределения: среднее значение составило 1 038 235 руб. при медиане 700 000 руб. Средний возраст автомобилей составляет 17,1 года, средний пробег — 196 725 км, средняя мощность двигателя — 137,9 л. с. По типу кузова доминируют седаны (свыше 500 000 наблюдений); внедорожники, хэтчбеки и универсалы также существенно представлены, тогда как купе и минивэны занимают нишевое положение. Ценовые характеристики сегментов существенно дифференцированы: от 710 тыс. руб. в среднем для хэтчбеков до 1 851 тыс. руб. для внедорожников.

Спецификация модели включает три группы объясняющих переменных.

Количественные факторы износа и технических характеристик: натуральный логарифм пробега $\ln(\text{Пробег})$; возраст автомобиля и его квадрат Возраст^2 для учёта убывающего темпа обесценения; мощность двигателя в л. с.; интервальные дамми мощности (100–150, 150–200, 200–250 и свыше 250 л. с.); удельная мощность (л. с. на литр объёма); кусочно-линейные сплайны пробега с узлами в точках 50, 150 и 250 тыс. км; взаимодействие возраста и логарифма пробега ($\text{Возраст} \times \ln(\text{Пробег})$).

Категориальные характеристики (фиктивные переменные): тип трансмиссии — АКПП, РКПП, вариатор (база: МКПП); тип двигате-

ля — дизель (база: бензин); наличие полного привода; цвет кузова — популярный (белый, чёрный, серый, серебристый) и редкий (розовый, жёлтый, оранжевый); характеристики продавца — тип (фирма) и стаж; факторы риска сделки — необходимость ремонта и проблемы с документами; брендовая принадлежность — премиальный сегмент, азиатский масс-маркет, китайские бренды; расположение руля; региональные дамми — Москва/ЦФО, Санкт-Петербург/СЗФО, Юг, Сибирь, Дальний Восток.

Сконструированные взаимодействия: «Премиум-бренд × Возраст» — для учёта убывающей брендовой надбавки; «Премиум × Москва»; «Дизель × Полный привод»; «Правый руль × Дальний Восток».

Выборка разбита на обучающую (80 %) и тестовую (20 %) части. Программная реализация выполнена на языке Java с использованием библиотек Apache Commons Math и SMILE.

3. Методы построения моделей

На первом этапе для каждого из шести сегментов построены МНК-модели с алгоритмом прямого пошагового включения переменных [2]. На каждой итерации в модель добавляется переменная, обеспечивающая наибольший прирост R^2 на тестовой выборке; процедура завершается при исчерпании значимого прироста. Предварительно из рассмотрения исключаются переменные без вариативности в обучающей выборке.

Диагностика мультиколлинеарности осуществляется через показатель VIF — фактор инфляции дисперсии [1]:

$$VIF_j = \frac{1}{1 - R_j^2},$$

где R_j^2 — коэффициент детерминации регрессии j -й переменной на остальные. Значения $VIF > 10$ интерпретируются как признак существенной мультиколлинеарности.

На втором этапе для тех же сегментов построены модели с регуляризацией Elastic Net (параметры: L1-ratio = 0,05, $\alpha = 0,1$) на полном наборе признаков без предварительного отбора. Качество моделей оценивалось по R^2 , скорректированному $Adj. R^2$, RMSE и

MAE. Сравнение подходов проводилось как по метрикам качества, так и по структуре и устойчивости оценённых коэффициентов.

4. Результаты

В табл. 1 приведены сравнительные метрики качества моделей по сегментам.

Таблица 1. Метрики качества моделей.

Сегмент	R^2 пош.	R^2 EN	ΔR^2	RMSE пош.	RMSE EN	Δ RMSE
COUPE	0,804	0,827	+0,023	0,402	0,384	-0,019
SEDAN	0,786	0,786	≈ 0	0,364	0,362	-0,001
SUV	0,815	0,819	+0,004	0,342	0,341	-0,001
WAGON	0,784	0,786	+0,003	0,334	0,331	-0,003
HATCHBACK	0,762	0,765	+0,003	0,336	0,334	-0,002
MINIVAN	0,757	0,756	≈ 0	0,357	0,355	-0,002

Коэффициент детерминации варьируется от 0,757 до 0,819. Наибольшую объясняющую способность демонстрируют модели внедорожников и купе. Регуляризация Elastic Net обеспечивает улучшение RMSE во всех сегментах; наиболее значимый эффект наблюдается для купе ($\Delta R^2 = +0,023$), что объясняется малым объёмом выборки данного сегмента и склонностью пошагового отбора к переобучению в подобных условиях.

Разберем ключевые детерминанты стоимости. Возраст оказывает нелинейное убывающее влияние на цену: каждый дополнительный год снижает стоимость примерно на 8,5 % для седанов и 10,1 % для внедорожников, однако положительный квадратичный член фиксирует замедление данного эффекта по мере старения автомобиля. Пробег включён в логарифмической форме: увеличение пробега на 1 % снижает цену на 0,16 % для седанов и 0,14 % для внедорожников.

Принадлежность к премиальному сегменту формирует значительную ценовую надбавку: 75,5 % для седанов и 84,3 % для внедорожников. Отрицательный коэффициент взаимодействия «Премиум \times Возраст» (-0,013 и -0,027 соответственно) свидетельствует о постепенном сближении цен с массовым рынком по мере старения автомобиля. Азиатские марки формируют надбавку порядка 34–39 %, тогда как китайские бренды несут дисконт 33–40 % вследствие неустоявшейся рыночной репутации.

Мощность двигателя оказывает пороговый нелинейный эффект, особенно выраженный для внедорожников: переход в диапазон 200–250 л. с. увеличивает цену примерно на 101 %, а свыше 250 л. с. — на 105 %. В сегменте седанов соответствующие надбавки скромнее (54 % и 52 %), что отражает меньшую роль мощности в ценообразовании массового рынка.

Факторы риска сделки демонстрируют наиболее высокую устойчивость между методами. Необходимость ремонта снижает цену на 48–60 % в зависимости от сегмента, проблемы с документами — на 33–44 %. Любая автоматизированная трансмиссия даёт положительную надбавку относительно механической, наибольшую — РКПП (~ 39 % для седанов). Исключение составляет сегмент купе, где АКПП имеет отрицательный коэффициент (–0,117), что отражает предпочтение покупателей спортивных автомобилей в пользу механической трансмиссии.

5. Сравнительный анализ подходов

Применение Elastic Net выявило острую структурную мультиколлинеарность переменных износа в полной спецификации: VIF для взаимодействия «Возраст × Пробег» достигает 454 для седанов и 1 184 для внедорожников. Природа данной мультиколлинеарности является структурной — переменные «Возраст», «Возраст²», «Возраст × ln(Пробег)» и сплайны пробега совместно параметризуют единую нелинейную поверхность износа. Пошаговый алгоритм устранял данную проблему механистическим исключением части переменных ценой нарушения принципа иерархичности спецификации; L_2 -компонента Elastic Net распределяет объясняющую нагрузку между ними без исключения.

Брендовые надбавки демонстрируют высокую межмодельную устойчивость: для седанов коэффициент «Премиум-бренд» составляет 0,755 в пошаговой модели и 0,751 в Elastic Net. В сегменте купе, напротив, брендовый коэффициент возрастает с 0,431 до 0,577, поскольку полная спецификация позволяет разграничить эффекты мощности и бренда, конкурировавшие за объяснение вариации цены в пошаговой модели. Региональные эффекты более полно отражаются в Elastic Net: переменные, отброшенные пошаговым алгоритмом, получают небольшие ненулевые коэффициенты, реализуя

непрерывную шкалу значимости вместо бинарного включения или исключения.

Практическая верификация на реальных объявлениях подтверждает преимущество регуляризации. Для Nissan Almera (825 тыс. руб.) Elastic Net даёт отклонение -4% (790 тыс. руб.), тогда как пошаговая модель завышает оценку на 15% (951 тыс. руб.), а сторонний сервис занижает её до 690 тыс. руб. Для Audi A6 (4 650 тыс. руб.) Elastic Net практически воспроизводит рыночную цену (4 664 тыс. руб.), тогда как сторонний сервис занижает оценку до 4 330 тыс. руб.

6. Заключение

В ходе исследования построены и сравнительно оценены эконометрические модели стоимости подержанных автомобилей для шести рыночных сегментов. Коэффициенты детерминации в диапазоне $0,757-0,819$ свидетельствуют о высокой объясняющей способности всех спецификаций. Установлено, что ключевыми детерминантами цены выступают нелинейные эффекты возраста и пробега, мощность двигателя с пороговыми эффектами, брендовое позиционирование и факторы риска сделки.

Сравнительный анализ показал, что пошаговая регрессия обеспечивает компактные и легко интерпретируемые спецификации, тогда как Elastic Net предпочтительна по критериям устойчивости, сохранения иерархии модели и прогностической точности — особенно в малых выборках. Выявленные закономерности воспроизводятся независимо от метода оценивания, что свидетельствует об их содержательной, а не артефактной природе.

Полученные результаты подтверждают применимость гедонистического подхода к моделированию вторичного автомобильного рынка и могут служить основой для разработки прикладных систем оценки рыночной стоимости транспортных средств.

Список литературы

- [1] Гмурман, В. Е. Теория вероятностей и математическая статистика. — 12-е изд. — М. : Издательство Юрайт, 2015. — 479 с. — URL: <https://djvu.online/file/6psbmnrHRtEc4>.

- [2] Дрейпер, Н. Прикладной регрессионный анализ: В 2-х кн. Кн. 1 / Н. Дрейпер, Г. Смит. — 2-е изд., перераб. и доп. — М. : Финансы и статистика, 1986. — 366 с. — URL: <https://djvu.online/file/P1XrffZGifEyp>.
- [3] Магнус, Я. Р., Катышев П. К., Пересецкий А. А. Эконометрика. Начальный курс : учеб. / Я. Р. Магнус, П. К. Катышев, А. А. Пересецкий. — 6-е изд., перераб. и доп. — М. : Дело, 2004. — 576 с. — URL: <https://djvu.online/file/ChilMzJmQaqDf>.
- [4] Tibshirani, R. Regression Shrinkage and Selection via the Lasso // Journal of the Royal Statistical Society. Series B. — 1996. — Vol. 58, № 1. — P. 267–288. — URL: <https://www.jstor.org/stable/2346178>.
- [5] Zou, H. Regularization and Variable Selection via the Elastic Net / H. Zou, T. Hastie // Journal of the Royal Statistical Society. Series B. — 2005. — Vol. 67, № 2. — P. 301–320. — DOI: <https://doi.org/10.1111/j.1467-9868.2005.00503.x>.

Библиографическая ссылка

Ивашин, М. Г. Построение эконометрических моделей оценки стоимости подержанных автомобилей // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 58–66.

Сведения об авторах

ИВАШИН МАКСИМ ГЕННАДЬЕВИЧ
Студент магистратуры
Направление «Прикладная информатика»

УДК 004.8, 616-07

Анализ применения искусственного интеллекта в медицине: достижения, проблемы и перспективы

Ионкуа И. Т.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры
информационных технологий Р. Н. Гордеевым

Аннотация. Цель исследования — провести сравнительный анализ алгоритмов машинного обучения (ML) для классификации медицинских данных и оценить перспективы применения искусственного интеллекта (ИИ) в российской медицинской диагностике. Методы: выполнено экспериментальное сравнение пяти ML-алгоритмов (логистическая регрессия, решающее дерево, случайный лес, градиентный бустинг, многослойный перцептрон) на двух синтетических датасетах, моделирующих задачи диагностики сердечно-сосудистых заболеваний и сахарного диабета. Оценка проводилась с использованием 5-кратной кросс-валидации по метрикам Ассигасу, F1-score и AUC-ROC. Результаты: ансамблевые методы (градиентный бустинг, случайный лес) продемонстрировали более высокую производительность по сравнению с линейными моделями, однако достижение 100 % ассигасу на синтетических данных свидетельствует о тривиальной разделимости выборок и является артефактом генерации, а не показателем качества моделей. Выводы: ансамблевые методы обладают потенциалом для задач медицинской классификации, однако полученные результаты требуют обязательной валидации на реальных клинических данных, а перенесение выводов на практическое применение преждевременно.

Ключевые слова: искусственный интеллект, машинное обучение, медицинская диагностика, классификация, синтетические данные, ансамблевые методы, сравнительный анализ.

Введение

Искусственный интеллект (ИИ) является одним из наиболее динамично развивающихся направлений современной информатики, а его применение в медицине привлекает внимание исследователей и практиков во всём мире. По данным Всемирной организации здравоохранения (ВОЗ), к 2025 году технологии ИИ планируется внедрить в 64 странах Европейского региона [3]. Управление по санитарному надзору за качеством пищевых продуктов и медикаментами США (FDA) утвердило сотни медицинских изделий на основе ИИ для клинического применения [2]. В Российской Федерации также наблюдается рост интереса к цифровизации здравоохранения: разрабатываются национальные программы внедрения ИИ в диагностический процесс, формируется нормативно-правовая база [1]. Вместе с тем масштабное внедрение ИИ в медицинскую практику сталкивается с рядом этических, правовых и технологических барьеров, требующих системного анализа.

Несмотря на активное развитие ИИ в медицине, остаются нерешёнными вопросы о сравнительной эффективности различных алгоритмов машинного обучения (ML) для конкретных задач диагностики, а также о реалистичных перспективах внедрения ИИ-систем в российское здравоохранение с учётом регуляторных, этических и технологических ограничений. Дефицит сравнительных исследований, опирающихся на воспроизводимую методологию, затрудняет обоснованный выбор алгоритмов для практического применения.

Целью исследования является сравнительный анализ алгоритмов ML для классификации медицинских данных и оценка перспектив применения ИИ в российской медицинской диагностике. Для достижения цели поставлены следующие задачи: 1) проанализировать текущее состояние применения ИИ в медицинской диагностике; 2) выявить ключевые проблемы внедрения ИИ в здравоохранение; 3) провести экспериментальное сравнение пяти ML-алгоритмов на задачах диагностики; 4) оценить ограничения и перспективы применения методов в клинической практике.

Таблица 1. Типы машинного обучения.

Тип	Методы	Примеры применения в медицине
С учителем (supervised)	Логистическая регрессия, решающее дерево, случайный лес, градиентный бустинг, метод опорных векторов	Классификация заболеваний, прогнозирование исходов лечения
Без учителя (unsupervised)	<i>K</i> -средних, иерархическая кластеризация, метод главных компонент	Выявление групп пациентов, снижение размерности данных
С подкреплением (reinforcement)	Q-learning, policy gradient	Оптимизация стратегий лечения, управление дозировкой препаратов

1. Теоретические основы применения ИИ в медицине

Искусственный интеллект (ИИ) — область информатики, направленная на создание систем, способных выполнять задачи, требующие интеллектуальных способностей человека: распознавание образов, обучение на основе опыта, принятие решений в условиях неопределённости [4]. Термин «искусственный интеллект» был предложен Дж. Маккарти в 1956 году на Дартмутской конференции, однако ключевые идеи были заложены ранее в работах А. Тьюринга по тесту на интеллектуальность машин. Современный этап развития ИИ характеризуется доминированием методов машинного обучения (ML) — подхода, при котором система выявляет закономерности в данных без явного программирования правил [4]. В отличие от ранних экспертных систем, основанных на вручную заданных правилах, ML-методы автоматически адаптируются к структуре данных, что обеспечивает их гибкость при решении разнородных задач.

Основные типы машинного обучения представлены в табл. 1.

В рамках обучения с учителем ансамблевые методы, в частности градиентный бустинг, реализованный в библиотеках XGBoost (eXtreme Gradient Boosting) и LightGBM (Light Gradient Boosting Machine), демонстрируют высокую эффективность в задачах классификации табличных медицинских данных, обеспечивая конкурентную точность при умеренных вычислительных затратах [4].

Глубокое обучение — подраздел ML, основанный на многослойных нейронных сетях — включает архитектуры различной специализации: многослойный перцептрон (MLP) для задач классификации структурированных данных, свёрточная нейронная сеть (CNN) для анализа медицинских изображений (рентгенограммы, КТ, МРТ) и рекуррентная нейронная сеть (RNN) для обработки последовательностей, таких как временные ряды показателей пациентов из электронных медицинских карт [4]. Выбор конкретного метода определяется характером данных и поставленной задачей: табличные данные, изображения и временные ряды требуют различных архитектурных решений.

2. Анализ применения ИИ в медицине

В области медицинской диагностики ИИ-системы наиболее активно применяются в узких, хорошо формализованных задачах, прежде всего при анализе медицинских изображений и поддержке принятия решений. По данным FDA, значительная часть зарегистрированных медицинских изделий на основе ИИ относится именно к диагностическим и радиологическим приложениям [2]. Указанные достижения не означают готовности ИИ к автономной диагностике в широкой клинической практике: результаты алгоритмов требуют клинической валидации, контроля качества данных и участия медицинского специалиста.

По данным FDA, к 2024 году утверждены сотни медицинских изделий на основе ИИ для клинического применения; подавляющее большинство предназначено для задач медицинской визуализации [2]. В Российской Федерации развиваются проекты внедрения ИИ в диагностику: системы анализа медицинских изображений применяются в пилотном режиме в ряде медицинских учреждений [1]. При этом масштабной оценки клинической эффективности отечественных ИИ-систем в реальной практике пока не проводилось.

Масштабное внедрение технологий ИИ в здравоохранение требует системного решения ряда проблем. Смещение (bias) в обучающих данных может приводить к неравной точности диагностики для разных групп пациентов, что представляет особую опасность в медицине, где цена ошибки высока [3]. Проблема интерпретируемости

(«чёрный ящик») ограничивает доверие врачей к рекомендациям ИИ-систем и затрудняет выявление причин ошибочных предсказаний [3]. Качество и представительность данных для обучения остаются ключевыми факторами, определяющими клиническую применимость моделей: неполные, зашумлённые или нерепрезентативные данные приводят к ненадёжным результатам [4]. Перечисленные ограничения обуславливают необходимость осторожного подхода к внедрению ИИ в клинический процесс.

3. Этические и правовые аспекты

Применение ИИ в медицине порождает комплекс этических проблем: алгоритмическое смещение, способное усугубить неравенство в доступе к качественной диагностике; недостаточная прозрачность принятия решений ИИ-системами, затрудняющая верификацию рекомендаций; неопределённость ответственности за ошибочные диагнозы, поставленные при участии алгоритмов, — между врачом, разработчиком системы и медицинским учреждением [3].

В Российской Федерации регулирование и организационное внедрение ИИ в медицине развиваются вместе с практикой цифровизации здравоохранения: появляются пилотные проекты, регламентируются подходы к оценке эффективности и безопасности, формируются требования к применению программных решений в клиническом процессе [1]. В международной практике медицинские ИИ-системы рассматриваются как технологии, требующие контроля качества, прозрачности, управления рисками и оценки безопасности до внедрения в практику [3]. Регуляторный подход FDA также подчёркивает необходимость проверки медицинских изделий на основе ИИ перед клиническим использованием [2].

4. Материалы и методы

Данные. Для проведения эксперимента использованы два синтетических датасета, сгенерированных с помощью функции `make_classification` библиотеки `scikit-learn` с параметрами, моделирующими структуру реальных медицинских `datasets`. Датасет Heart Disease содержит 1 000 записей и 13 признаков с бинарной целевой переменной (наличие/отсутствие сердечно-сосудистого заболевания), что

по структуре соответствует реальному датасету UCI Heart Disease (Cleveland), включающему такие клинические показатели, как возраст, артериальное давление, уровень холестерина и результаты электрокардиографии. Датасет Diabetes содержит 1 000 записей и 8 признаков с бинарной целевой переменной (наличие/отсутствие сахарного диабета), что аналогично структуре датасета Pima Indians Diabetes, содержащего результаты глюкозотолерантного теста, индекс массы тела и другие метаболические показатели. Конкретные параметры генерации (`n_informative`, `n_redundant`, `class_sep`, `flip_y`) в исходной работе не документированы, что ограничивает воспроизводимость эксперимента. Классы в обоих датасетах сбалансированы (соотношение 50/50). Необходимо учитывать, что синтетические данные не отражают реальные клинические распределения признаков и могут содержать артефакты, обуславливающие более высокую разделимость классов по сравнению с реальными выборками.

Алгоритмы и гиперпараметры. В эксперименте использованы пять алгоритмов машинного обучения, представляющих основные методологические подходы к классификации: логистическая регрессия (Logistic Regression) — линейный метод, с параметрами по умолчанию scikit-learn; решающее дерево (Decision Tree) — с параметрами по умолчанию scikit-learn; случайный лес (Random Forest) — ансамблевый метод на основе bagging (`n_estimators=100`, остальные параметры по умолчанию); градиентный бустинг (Gradient Boosting) — ансамблевый метод последовательного обучения (`learning_rate=0,1`, `n_estimators=100`, остальные параметры по умолчанию); многослойный перцептрон (MLP) — нейросетевой метод (`hidden_layer_sizes` по умолчанию scikit-learn). Выбор данных алгоритмов обусловлен необходимостью представления различных классов моделей: линейных, древовидных, ансамблевых и нейросетевых, что позволяет провести корректное сравнение методологических подходов.

Процедура эксперимента. Каждый датасет разделён на обучающую и тестовую выборки в соотношении 80/20 с использованием стратифицированного разбиения (stratified split) для сохранения пропорций классов. Для оценки обобщающей способности моделей применена 5-кратная стратифицированная кросс-валидация (StratifiedKFold) на обучающей выборке; средний показатель кросс-

Таблица 2. Результаты классификации на датасете Heart Disease.

Алгоритм	Accuracy	F1-score	AUC-ROC
Logistic Regression	0,87	0,87	0,93
Decision Tree	0,92	0,92	0,92
Random Forest	0,96	0,96	0,99
Gradient Boosting	0,97	0,97	0,99
MLP	0,90	0,90	0,96
DummyClassifier	0,50	0,00	0,50

валидации составил CV Mean = 0,994. Эксперимент реализован на языке Python с использованием библиотеки scikit-learn. В качестве базового уровня (baseline) использован `DummyClassifier(strategy="most_frequent")`, предсказывающий наиболее частый класс; на сбалансированных данных (50/50) данный классификатор демонстрирует точность около 50%, что служит нижней границей для оценки значимости полученных результатов.

Метрики оценки. Качество классификации оценивалось по трём метрикам, стандартным для задач бинарной классификации в медицине: Accuracy (точность — доля правильных предсказаний от общего числа наблюдений); F1-score (F1-мера — гармоническое среднее precision и recall, учитывающее баланс между полнотой и точностью положительного класса); AUC-ROC (площадь под ROC-кривой — мера качества бинарной классификации, отражающая способность модели различать классы при различных порогах принятия решения). Выбор данного набора метрик обусловлен необходимостью комплексной оценки: Accuracy отражает общую долю верных предсказаний, F1-мера учитывает дисбаланс ошибок I и II рода, а AUC-ROC оценивает качество ранжирования независимо от порога классификации.

5. Результаты

Результаты классификации на датасете Heart Disease представлены в табл. 2.

На датасете Heart Disease ансамблевые методы (Gradient Boosting, Random Forest) показали более высокие результаты по всем трём

Таблица 3. Результаты классификации на датасете Diabetes.

Алгоритм	Accuracy	F1-score	AUC-ROC
Logistic Regression	0,93	0,93	0,98
Decision Tree	0,97	0,97	0,97
Random Forest	1,00	1,00	1,00
Gradient Boosting	1,00	1,00	1,00
MLP	0,95	0,95	0,99
DummyClassifier	0,50	0,00	0,50

метрикам по сравнению с линейным методом (Logistic Regression) и нейросетевым подходом (MLP). Решающее дерево продемонстрировало промежуточные результаты.

Результаты классификации на датасете Diabetes представлены в табл. 3.

На датасете Diabetes Gradient Boosting и Random Forest показали полноту классификации (accuracy = 100 %, F1-score = 1,0, AUC-ROC = 1,0). DummyClassifier на обоих датасетах продемонстрировал точность около 50 %, что соответствует ожидаемому значению для сбалансированных данных и подтверждает, что классификация не является тривиальной на уровне постоянного предсказания мажоритарного класса. Различия в результатах между двумя датасетами (Heart Disease: максимальная точность 97 %; Diabetes: максимальная точность 100 %) указывают на разные уровни сложности задач классификации, что может быть связано с различным числом признаков (13 против 8) и свойствами сгенерированных данных.

6. Обсуждение

Полученные результаты согласуются с известными закономерностями: ансамблевые методы (Gradient Boosting, Random Forest) превосходят линейные (Logistic Regression) на обоих датасетах, что объясняется способностью ансамблей моделировать нелинейные взаимодействия признаков. Нейросетевой метод (MLP) показал промежуточные результаты, что может быть связано с чувствительностью многослойного перцептрона к подбору гиперпараметров и объёму обучающей выборки. Различия между датасетами (максимальная

ассурасу 97 % на Heart Disease против 100 % на Diabetes) могут объясняться разной структурой задач: датасет Diabetes содержит меньшее число признаков (8 против 13), и генерация синтетических данных с меньшей размерностью могла привести к более высокой разделимости классов.

Сопоставление с исследованиями на реальных клинических данных показывает, что результаты на синтетических выборках обычно оказываются более оптимистичными, чем результаты на реальных медицинских данных. Поэтому полученные показатели следует интерпретировать как характеристику сгенерированных выборок и экспериментальной постановки, а не как доказательство превосходства использованных моделей в клинической практике.

Достижение ассурасу = 100 % на датасете Diabetes наиболее вероятно обусловлено совокупностью факторов: (1) тривиально разделимые данные, сгенерированные функцией `make_classification` с параметрами по умолчанию (`class_sep = 1,0`), которые могут формировать линейно разделимые классы; (2) потенциальная утечка информации (data leakage) между обучающей и тестовой выборками, связанная с общими параметрами генерации распределений; (3) практическое отсутствие шума в метках (`flip_y = 0,01` по умолчанию). Значение кросс-валидации CV Mean = 0,994 подтверждает не высокую обобщающую способность моделей, а свойства самих данных. DummyClassifier показал точность около 50 %, что исключает тривиальность классификации на уровне предсказания мажоритарного класса, однако не снимает вопроса о высокой информативности признаков в сгенерированной выборке.

Кратко резюмируя, полученные результаты отражают прежде всего свойства сгенерированных данных, а не реальную эффективность моделей. Для обоснованного вывода о применимости ансамблевых методов в медицинской диагностике необходимы эксперименты на реальных клинических данных с корректной процедурой валидации. Подробный анализ ограничений эксперимента представлен в следующем разделе.

7. Ограничения исследования

Настоящее исследование имеет ряд существенных ограничений, которые необходимо учитывать при интерпретации результатов. Во-

первых, использованы синтетические данные, сгенерированные программным способом, которые не отражают реальные клинические распределения признаков и закономерности; перенос результатов на реальные медицинские данные не гарантирован. Во-вторых, объём каждого датасета составляет 1 000 записей, что недостаточно для надёжной оценки производительности ML-моделей в медицинских задачах, где обычно требуются выборки существенно большего объёма. В-третьих, результаты не прошли внешнюю валидацию — не проверены на независимой выборке или в другом медицинском учреждении. В-четвёртых, система не тестировалась в реальных клинических условиях, не имеет регистрации в качестве медицинского изделия и не может рассматриваться как клинический инструмент диагностики. В-пятых, генератор синтетических данных может вносить систематические искажения (bias), влияющие на производительность моделей и не характерные для реальных данных. В-шестых, не проводился анализ объяснимости моделей, что является существенным ограничением для медицинского применения, где интерпретируемость решений обязательна [3]. В-седьмых, существует высокая вероятность утечки информации (data leakage) между обучающей и тестовой выборками вследствие общих параметров генерации распределений, что может объяснять аномально высокие показатели accuracy, вплоть до 100 %.

Указанные ограничения необходимо учитывать при интерпретации результатов. Основной вывод исследования — демонстрация методологии сравнительного анализа ML-алгоритмов, а не оценка их клинической применимости.

Заключение

Проведённый сравнительный анализ пяти алгоритмов машинного обучения на двух синтетических датасетах, моделирующих задачи диагностики сердечно-сосудистых заболеваний и сахарного диабета, показал, что ансамблевые методы (градиентный бустинг, случайный лес) превосходят линейные (логистическая регрессия) по всем рассмотренным метрикам (Accuracy, F1-score, AUC-ROC). Данный результат согласуется с известными закономерностями ML-литературы и объясняется способностью ансамблей моделировать нелинейные взаимодействия признаков [4].

Вместе с тем достижение ассигасы = 100% на синтетическом датасете Diabetes является артефактом генерации данных, а не показателем качества моделей. Сопоставление с исследованиями на реальных клинических данных подтверждает, что синтетические данные могут существенно завышать оценки производительности. Полученные результаты отражают прежде всего свойства сгенерированных данных, а не реальную эффективность алгоритмов.

В Российской Федерации развиваются проекты внедрения ИИ в медицинскую диагностику [1]; при этом международные рекомендации и регуляторные подходы подчёркивают необходимость валидации ИИ-систем на реальных клинических данных, оценки рисков и контроля безопасности перед применением в клинической практике [2,3]. В представленном виде система является исследовательским прототипом.

Перспективными направлениями дальнейших исследований являются: проведение экспериментов на реальных клинических данных с корректной процедурой валидации; развитие методов объяснимого ИИ (ХАИ) для повышения доверия врачей к рекомендациям алгоритмов; совершенствование нормативно-правовых рамок, учитывающих специфику обучаемых алгоритмов в медицине [3].

Список литературы

- [1] Итоги внедрения искусственного интеллекта в здравоохранении России / В. Ваньков, О. Р. Артемова, О. Э. Карпов [и др.] // Врач и информационные технологии. — 2024. — № 3. — С. 32–43. — DOI: https://doi.org/10.25881/18110193_2024_3_32.
- [2] Artificial Intelligence-Enabled Medical Devices // U. S. Food and Drug Administration : [сайт]. — URL: <https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-enabled-medical-devices> (дата обращения: 17.05.2026).
- [3] Ethics and governance of artificial intelligence for health. — Geneva : World Health Organization, 2021. — 150 p. — URL: <https://www.who.int/publications/i/item/9789240029200>.

- [4] Goodfellow, I. Deep Learning / I. Goodfellow, Y. Bengio, A. Courvill. — Cambridge : The MIT Press, 2016. — 800 p. — URL: <https://www.deeplearningbook.org/>.

Библиографическая ссылка

Ионкуа, И. Т. Анализ применения искусственного интеллекта в медицине: достижения, проблемы и перспективы // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 67–78.

Сведения об авторах

ИОНКУА ИДЖОНГАНГ ТИЕРРИ

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 004.932.2

Сравнительный анализ архитектур для одновременной детекции людей и лиц в видеопотоке

Ищенко А. А.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий И. С. Солдатенко

Аннотация. В статье проводится сравнительный анализ архитектур для одновременного обнаружения людей и выделения лиц в видеопотоке в реальном времени. В рамках данной работы проведены эксперименты с последовательными пайплайнами на основе YOLO, MTCNN и YOLO-face, включая вырезание областей интереса (ROI), которые показали, что такие конфигурации приводят к падению производительности до 2–4 FPS. Оптимальным решением оказывается параллельный запуск детектора людей YOLO (модели yolo261-pose.pt) и детектора лиц YOLO-face на полном кадре без вырезания ROI, достигающий 17 FPS. Для сопоставления лиц с людьми используется геометрический метод по ключевым точкам головы.

КЛЮЧЕВЫЕ СЛОВА: компьютерное зрение, детекция лиц, YOLO.

Введение

За последнее десятилетие количество камер видеонаблюдения выросло настолько, что ручная обработка видеоданных стала невозможной, поэтому существует потребность в автоматической обработке видеопотоков. Одна из базовых задач — одновременно находить в кадре людей и выделять их лица. Это необходимо для систем контроля доступа, подсчёта посетителей, анализа поведения покупателей и многих других приложений.

На первый взгляд задача кажется тривиальной: достаточно взять готовый детектор людей и готовый детектор лиц и запустить их один за другим. Однако на практике возникает ряд трудностей.

Семейство нейросетей YOLO (You Only Look Once) известно своей высокой скоростью обнаружения объектов — оно способно обрабатывать десятки кадров в секунду даже на скромном оборудовании [2]. Но стандартные модели YOLO обучены на общих классах объектов (человек, машина, собака и т. п.) и не специализируются на лицах. Специализированные детекторы лиц, например MTCNN, показывают высокую точность, но работают значительно медленнее [1]. В результате простой последовательный запуск приводит к падению частоты кадров до величин, неприемлемых для реального времени.

Цель данной работы — экспериментально сравнить несколько архитектурных конфигураций и найти ту, которая обеспечит наилучший баланс между скоростью обработки и полнотой информации (боксы людей, боксы лиц и их взаимная привязка).

1. Экспериментальная установка

Все эксперименты проводились на одном и том же оборудовании: персональный компьютер с процессором Intel Core i7-12700H, видеокартой NVIDIA GeForce RTX 3080 Ti (12 ГБ видеопамяти) и 32 ГБ оперативной памяти. Операционная система — Windows 11, язык программирования — Python 3.10, библиотеки — Ultralytics YOLO, MTCNN.

В качестве видеоданных использовались последовательности из общедоступных бенчмарков MOT17 и MOT20. Исходные наборы поставляются в виде отдельных кадров (PNG и JPG), поэтому предварительно они были объединены в непрерывные видеофайлы формата MP4 с сохранением оригинальной частоты кадров (25–30 FPS) и разрешения (от 640×480 до 1920×1080). Всего было подготовлено 18 видеопоследовательностей: 14 из MOT17 и 4 из MOT20. Этот материал охватывает как сцены с умеренной плотностью пешеходов, так и сцены с высокой скученностью (до 40 человек в кадре), что позволяет оценить поведение алгоритмов в широком диапазоне условий.

Для оценки скорости обработки использовался модуль `time`, измерялось время выполнения непосредственно вызова модели (`results = model(frame)`). При этом следует учитывать, что операционная система и среда выполнения Python могут вносить дополнительные

задержки (например, переключение контекста, сборка мусора, кэширование данных). Для всего видеопотока время работы модели на каждом кадре суммировалось, после чего данная метрика переводилась в кадры в секунду (FPS) — сколько кадров в секунду смогла бы обработать модель без учёта накладных расходов на работу с диском или камерой. Для снижения влияния различных факторов на результаты каждый эксперимент повторялся не менее трёх раз на каждой видеопоследовательности, после чего вычислялось среднее арифметическое.

2. Первые эксперименты: детекторы по отдельности

Сначала измерялась производительность базовых моделей по отдельности. Модель YOLO (для детекции людей) показала среднюю скорость 21 кадр в секунду. Это означает, что она успевает обрабатывать видео в реальном времени, но при этом выдаёт только ограничивающие рамки людей и их идентификаторы (благодаря встроенному трекеру). Никакой информации о лицах она не даёт.

Детектор лиц MTCNN на тех же видео работал значительно медленнее — всего 3 кадра в секунду. Качество детекции при этом было хорошим: MTCNN успешно находил даже небольшие лица и дополнительно выдавал пять ключевых точек (глаза, нос, уголки рта). Однако скорость 3 FPS неприемлема для систем реального времени, где требуется хотя бы 15–25 кадров в секунду для плавного отображения.

3. Последовательный пайплайн с вырезанием ROI

Следующим шагом стала комбинация детекторов в последовательный пайплайн: сначала быстрый YOLO находит всех людей на кадре, затем для каждого человека вырезается его ограничивающая рамка (эта область называется область интереса — Region of Interest — ROI), и уже на этом маленьком фрагменте запускается MTCNN для поиска лица. Ожидалось, что за счёт уменьшения входного изображения детектор лиц будет работать быстрее, а общая производительность должна возрасти.

Такой пайплайн был реализован. Вырезание ROI выполнялось с помощью стандартных срезов NumPy (`roi = frame[y1:y2, x1:x2]`), что является достаточно эффективной операцией. Однако результат оказался противоположным ожидаемому. Средняя производительность упала до 2 кадров в секунду — даже ниже, чем при использовании одного MTCNN на полном кадре.

Анализ показал причину. Проблема не в медленности MTCNN как такового, а в том, что он вызывается многократно — столько раз, сколько человек обнаружено на кадре. В сценах со средней плотностью это 5–10 человек. Каждый отдельный вызов MTCNN на маленьком ROI выполняется быстрее, чем на полном изображении, но суммарное время 5–10 вызовов плюс накладные расходы на вырезание превышают время одного вызова на всём кадре. Кроме того, MTCNN не оптимизирован для обработки множества мелких изображений последовательно — преимущества GPU используются неэффективно.

4. Замена MTCNN на быстрый детектор лиц

Следующим шагом стала попытка заменить MTCNN на более современный и быстрый детектор лиц, основанный на архитектуре YOLO — YOLO-face (модель `yolov12l-face.pt`) [3]. В автономном режиме (на полном кадре) этот детектор показал впечатляющую скорость 23 кадра в секунду, что почти не отличается от скорости детекции людей с помощью YOLO.

Мы встроили YOLO-face в тот же последовательный пайплайн с вырезанием ROI: YOLO находит людей, вырезаются ROI, на каждом ROI запускается YOLO-face. Гипотеза была такой: раз YOLO-face сам по себе очень быстрый, то на маленьких ROI он будет работать ещё быстрее, и множественные вызовы не сильно навредят.

Однако, фактические результаты разошлись с выдвинутой гипотезой. Средняя производительность поднялась до 4 кадров в секунду — лучше, чем 2 FPS с MTCNN, но всё равно далеко до реального времени. Причина осталась прежней: детектор лиц вызывается много раз (по числу людей в кадре). Даже очень быстрая модель при 5–10 запусках на кадр теряет свои преимущества, потому что накладные расходы на последовательные вызовы и передачу данных суммируются.

5. Параллельный запуск детекторов на полном кадре

После этих экспериментов можно прийти к важному выводу: проблема не в выборе конкретного детектора лиц, а в самом принципе «один человек = один вызов детектора». Пока детектор лиц запускается столько же раз, сколько людей в кадре, производительность будет падать пропорционально плотности сцены. Даже если каждый вызов выполняется за 20 миллисекунд, десять вызовов дадут 200 миллисекунд, то есть 5 FPS — но это в идеальном случае без учёта накладных расходов. На практике накладные расходы (вырезание, переключения контекста) добавляют ещё больше задержки.

Возникла идея вообще отказаться от вырезания ROI и запускать детектор лиц один раз на весь кадр, параллельно с детектором людей. Тогда времени на детекцию лиц тратится ровно столько, сколько нужно одному вызову модели, независимо от количества людей. Единственная проблема — как потом сопоставить найденные лица с конкретными людьми, ведь детекторы работают независимо.

Предлагается следующая архитектура: на каждом кадре независимо (в смысле логики, а не многопоточности) запускаются две модели. Первая — YOLO (yolo261-pose.pt — детектор людей с дополнительным выводом ключевых точек тела). Эта модель не только находит ограничивающие рамки людей и присваивает им уникальные идентификаторы через встроенный трекер, но и выдаёт положение ключевых точек головы: нос, левый и правый глаз, левое и правое ухо. Вторая модель — YOLO-face (yolov121-face.pt), которая находит все лица на кадре. Обе работают с полным изображением, без вырезания ROI. Вызовы выполняются последовательно (в текущей реализации программы, GPU не может обрабатывать две модели одновременно, но это не принципиально).

Результат этой конфигурации — 17 кадров в секунду. Это лишь на 6 FPS меньше, чем у одного YOLO-face (23 FPS), и на 4 FPS меньше, чем у одного YOLO для людей (21 FPS). При этом система выдаёт полную информацию: боксы людей с идентификаторами, боксы лиц, а также ключевые точки головы. Скорость 17 FPS достаточно для практических систем реального времени (обычно порогом считается 15 FPS).

6. Сопоставление лиц с людьми по ключевым точкам

Теперь нужно решить задачу сопоставления — понять, какое лицо принадлежит какому человеку. Поскольку YOLO даёт нам ключевые точки головы для каждого человека, на основе этого предлагается простой геометрический алгоритм.

Для каждого человека из его набора точек выделяем пять точек головы (нос, левый глаз, правый глаз, левое ухо, правое ухо). Для каждого обнаруженного лица (рамка от YOLO-face) подсчитываем, сколько из этих пяти точек попадает внутрь рамки лица. Лицо привязывается к тому человеку, для которого количество попавших точек максимально. Дополнительным условием установлено, что это количество должно быть не менее двух — чтобы избежать случайных совпадений.

Этот метод оказался очень надёжным. Он не требует дополнительных вычислений (просто проверка принадлежности точки прямоугольнику), работает за время $O(\text{количество людей}^2)$ — что пренебрежимо мало по сравнению с нейросетевым инференсом — и устойчив к небольшим перекрытиям. По визуальной оценке на случайной выборке кадров алгоритм правильно сопоставлял лица в подавляющем большинстве случаев. Единственные ошибки возникали при экстремальных ракурсах (камера сверху) или когда лицо было почти полностью закрыто.

7. Итоговая система и её характеристики

Собрав все компоненты, получается работающая система, которая:

- принимает на вход видеофайл или поток с веб-камеры;
- на каждом кадре параллельно выполняет детекцию людей (с трекингом) и детекцию лиц;
- сопоставляет лица с людьми по ключевым точкам головы;
- визуализирует результат: рисует боксы людей и лиц, подписывает идентификаторы людей, отображает ключевые точки головы (опционально).

Производительность 17 FPS достигается на тестовых видео с разрешением 1920×1080. Задержка обработки одного кадра составляет около 59 миллисекунд, что приемлемо для большинства приложений видеонаблюдения.

Заключение

В статье описан сравнительный анализ архитектур для одновременного обнаружения людей и выделения лиц в видеопотоке. Последовательно протестировано несколько конфигураций: от использования детекторов по отдельности до последовательных пайплайнов с вырезанием ROI. Оказалось, что подход с вырезанием областей интереса не только не ускоряет обработку, но и приводит к её замедлению из-за многократных вызовов детектора лиц, а замена медленного MTCNN на быстрый YOLO-face не решает проблему.

Наилучший результат получился при отказе от вырезания ROI и при переходе к параллельному запуску двух независимых детекторов на полном кадре: одного для людей (с выводом ключевых точек головы), другого для лиц. Такая конфигурация обеспечила стабильные 17 кадров в секунду. Для сопоставления лиц с людьми успешно применён метод на основе ключевых точек головы, получаемых от детектора позы.

Полученная система может быть использована в системах видеонаблюдения реального времени, контроля доступа, подсчёта посетителей и других прикладных задачах.

Список литературы

- [1] Акобян, Т. С. Сравнение современных технологий распознавания образов // Успехи кибернетики. — 2025. — Т. 6, № 1. — С. 175–179. — URL: <https://www.mathnet.ru/links/f64cb9ad78e5468152dec9db7da18f56/uk209.pdf>.
- [2] Гриценко, Г. Г. Аналитический обзор архитектур, моделей, методов и алгоритмов для локализации и трекинга неригидных объектов / Г. Г. Гриценко, В. П. Фраленко // Программные системы: теория и приложения. — 2024. — Т. 15, № 4 (63). — С. 111–151. — DOI: <https://doi.org/10.25209/2079-3316-2024-15-4-111-151>.

- [3] YOLO-face: a real-time face detector / W. Chen, H. Huang, S. Peng [et al.] // The Visual Computer. — 2020. — Vol. 37, № 4. — P. 805–813. — URL: <https://doi.org/10.1007/s00371-020-01831-7>.

Библиографическая ссылка

Ищенко, А. А. Сравнительный анализ архитектур для одновременной детекции людей и лиц в видеопотоке // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 79–86.

Сведения об авторах

ИЩЕНКО АНДРЕЙ АНДРЕЕВИЧ

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 004.912

Морфологический анализ слов русского языка с использованием конечных преобразователей

Константинова Е. А.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры информатики

Б. Н. Карловым

Аннотация. В статье рассмотрены три метода морфологического анализа русских слов на основе конечных преобразователей и вероятностных моделей: детерминированный FST, вероятностный суффиксный преобразователь и Наивный Байес. Проведено экспериментальное сравнение методов на словаре из 4 млн словоформ. Суффиксный ML-подход показал наивысшую точность 91,6 % при малом времени обучения, что подтверждает его эффективность для обработки неизвестных слов.

Ключевые слова: морфологический анализ, конечные преобразователи, детерминированный конечный автомат, вероятностный суффиксный преобразователь, русский язык, обработка естественного языка.

Введение

Морфологический анализ — это определение по слову его леммы и грамматических признаков (часть речи, падеж, число и т. д.). Он необходим во многих компьютерных системах, работающих с русским языком: переводчиках, поисковых программах, голосовых помощниках. Русский язык отличается сложной морфологией: одно и то же слово может менять окончания многими способами. Поэтому задача морфологического разбора для русского языка особенно трудна и актуальна. Один из самых удобных и быстрых инструментов для этой задачи — конечные преобразователи (finite-state transducers, FST). Они работают как автомат, который считывает буквы слова и выдаёт его возможные разборы. Основы применения конечных преобразователей для морфологического анализа заложены в работе [1]

В работе сравнивается несколько методов получения морфологического анализатора. Цель состоит в том, чтобы выяснить, какой из подходов лучше подходит для создания качественного морфологического анализатора для русского языка без долгой ручной работы.

1. Описание данных

Корпус словоформ представлен в виде текстового файла словаря, подготовленного М. Хагеном на основе «Полной акцентуированной парадигмы по Зализняку». Объём словаря — около 4,06 млн словоформ для 138,5 тыс. лемм. Формат файла: словарные статьи разделены пустыми строками; каждая строка словоформы содержит три поля, разделённых символом табуляции: словоформа, набор морфологических характеристик, уникальный цифровой идентификатор. Для экспериментального сравнения методов исходный словарь был разделён на две непересекающиеся выборки: обучающую (80 % лемм, 3,62 млн словоформ) и тестовую. Разбиение проводилось на уровне лемм, чтобы гарантировать, что ни одна словоформа из тестовой выборки не встречалась в обучении — это необходимо для объективной оценки способности моделей обобщать на неизвестные слова. Из-за большого объёма и полноты парадигм, датасет оптимально подходит для поставленных задач.

2. Прямой детерминированный конечный автомат

Для построения детерминированного конечного автомата была реализована структура данных, представляющая собой префиксное дерево (trie). Каждый узел дерева содержит словарь переходов по символам и поле для хранения тегов — наборов морфологических признаков. Например, словоформе «абажур» соответствует тег (сущ, неод, ед, муж, им), а словоформе «гнездиться» соответствует (гл, несов, непер, воз, инф). При этом при наличии нескольких тегов для одной словоформы они объединяются через разделитель «|». Процесс построения выполнялся в два этапа. На первом этапе осуществлена вставка всех пар (словоформа, тег) из обучающего корпуса: для каждой словоформы последовательно создавались узлы по символам, а

в терминальном узле фиксировался набор соответствующих тегов. На втором этапе реализована процедура минимизации полученного автомата алгоритмом Хопкрофта, что позволило уменьшить число состояний за счёт объединения эквивалентных узлов. Для анализа словоформ реализована функция, которая по символам слова проходит по автомату и возвращает список тегов из достигнутого узла либо пустой список при отсутствии пути. Точность на тренировочной выборке составила 100 %, что ожидаемо, поскольку автомат запоминает все увиденные словоформы. Однако на тестовой выборке, содержащей не входившие в обучение слова, точность оказалась менее 1 % (фактически, 0 % для неизвестных словоформ). Размер модели составил 1 030 862 перехода, что соответствует компактному представлению после минимизации. Время построения заняло 66,41 сек. Таким образом, прямой FST обеспечивает идеальную точность на закрытом словаре, но полностью неспособен к обобщению на новые слова.

3. Вероятностный суффиксный преобразователь

Этот метод основан на статистическом анализе окончаний словоформ. Модель обучается на парах (словоформа, тег) без явного задания лингвистических правил. Как отмечает Коробов [2], для русского языка эффективно использовать суффиксные правила для угадывания морфологических характеристик неизвестных слов. В ходе обучения для каждого суффикса длины от 1 до заданного максимального значения (в работе выбрано $L = 6$) накапливается частотный словарь тегов. Кроме того, целые словоформы также сохраняются в этом словаре для обеспечения точного распознавания знакомых слов. При предсказании для нового слова алгоритм перебирает суффиксы от максимальной длины к минимальной до тех пор, пока не встретит известный суффикс. Для найденного суффикса вычисляются сглаженные вероятности тегов, и возвращается наиболее вероятный из них. Если суффикс не обнаружен, используется априорное распределение тегов по всему корпусу. Благодаря такому подходу модель способна обобщать и корректно анализировать словоформы, отсутствовавшие в обучении. Точность на тестовой выборке составила 91 %, что подтверждает высокую эффективность

суффиксного метода для морфологического анализа русского языка. Размер модели — 2 469 051 суффикс (включая целые слова). Время построения (обучения) — 29,77 секунд, что значительно быстрее, чем минимизация прямого FST, и сопоставимо с временем его простого построения без минимизации. Таким образом, суффиксный преобразователь сочетает высокую точность на новых данных, быстрое обучение и компактное представление.

4. Наивный Байес

Третий метод представляет собой байесовский классификатор. Вероятность тега при данном слове оценивается по формуле Байеса:

$$P(tag | word) = P(tag)P(word | tag).$$

Априорные вероятности тегов $P(tag)$ вычисляются как относительные частоты тегов в обучающем корпусе. Эмиссионные вероятности для знакомых слов $P(word | tag)$ определяются по частотам пар (слово, тег) с применением сглаживания Лапласа, что позволяет избежать нулевых вероятностей для редких сочетаний. Сглаживание Лапласа, используемое для оценки вероятностей тегов по суффиксам, является стандартным методом для работы с разреженными данными в языковых моделях [3]

Для обработки незнакомых слов, отсутствовавших в обучении, в модель встроен суффиксный механизм, аналогичный второму методу: для каждого суффикса длины от 1 до 6 накапливается статистика тегов, а при предсказании вероятность $P(word | tag)$ оценивается как взвешенная сумма по всем длинам суффиксов с экспоненциальным затуханием, где более длинные суффиксы имеют больший вес. Если суффикс не найден, эмиссионная вероятность принимается равномерной по всем тегам. Точность на тестовой выборке (только неизвестные слова) составила 87 %, что ниже, чем у суффиксного метода (91 %). Это объясняется тем, что априорные вероятности тегов вносят дополнительный «шум»: редкие теги, имеющие характерные окончания, иногда проигрывают более частым, но неподходящим тегам. Размер модели составил 414 860 суффиксов плюс 1 288 уникальных тегов. Время построения (обучения) — 36,86 секунд, что сопоставимо с суффиксным методом. Таким образом,

Наивный Байес показывает хорошую точность, но уступает более простому суффиксному подходу при анализе изолированных слов.

Заключение

В результате были реализованы три метода морфологического анализа русских слов на основе конечных преобразователей и вероятностных моделей. Первый метод — детерминированный конечный преобразователь, построенный прямым префиксным деревом с последующей минимизацией. Второй метод — вероятностный суффиксный преобразователь, обучающийся на статистике окончаний словоформ. Третий метод — наивный Байес, объединяющий априорные вероятности тегов и суффиксную эмиссию для незнакомых слов. Результаты показали, что прямой детерминированный FST не способен обрабатывать неизвестные слова, но даёт 100 % на знакомых. Суффиксный преобразователь достиг наивысшей точности — 91,6 % при небольшом времени обучения. Байес показал точность 87,8 %, что ниже, чем у суффиксного метода, при значительно большем времени построения. Таким образом, для задачи морфологического анализа изолированных слов русского языка простой вероятностный суффиксный метод оказался эффективнее, чем Наивный Байес, что объясняется высокой предсказательной силой окончаний и отсутствием необходимости в априорных частотах тегов.

Список литературы

- [1] Beesley, K.R. Finite State Morphology / K.R. Beesley, L. Karttunen. — Stanford, CA : CSLI Publications, 2003. — 538 p. — URL: https://www.researchgate.net/publication/37705086_Finite-State_Morphology.
- [2] Korobov, M. Morphological Analyzer and Generator for Russian and Ukrainian Languages / M. Korobov // Analysis of Images, Social Networks and Texts / ed. by M. Yu. Khachay, N. Konstantinova, A. Panchenko [et al.]. — Cham : Springer, 2015. — P. 320–332. — DOI: https://doi.org/10.1007/978-3-319-26123-2_31.

- [3] Jurafsky, D. *Speech and Language Processing* / D. Jurafsky, J. H. Martin. — 3rd ed. — Hoboken, NJ : Prentice Hall, 2020. — 1024 p. — URL: <https://web.stanford.edu/~jurafsky/slp3/>.

Библиографическая ссылка

Константинова, Е. А. Морфологический анализ слов русского языка с использованием конечных преобразователей // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 87–92.

Сведения об авторах

КОНСТАНТИНОВА ЕЛЕНА АНДРЕЕВНА

Студентка магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 004.93'1, 004.932, 004.89

Разработка программы биометрической аутентификации сотрудников с помощью искусственного интеллекта

Корепанов М. В.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа С. В. Архиповым

Аннотация. В статье рассматривается программа биометрической аутентификации сотрудников по изображению лица. Описаны постановка задачи, свёрточная нейронная сеть с остаточными связями, функция потерь ArcFace, хранение биометрических признаков и сценарии регистрации и входа. Приведены результаты обучения и тестирования модели, а также меры защиты данных пользователей.

КЛЮЧЕВЫЕ СЛОВА: биометрическая аутентификация, распознавание лиц, свёрточная нейронная сеть, ArcFace, защита данных.

Введение

Развитие нейронных сетей привело к тому, что методы искусственного интеллекта стали применяться не только в исследовательских задачах, но и в прикладных системах безопасности. Одним из таких направлений является биометрическая аутентификация, при которой личность пользователя подтверждается по устойчивым физиологическим признакам. В отличие от пароля или пропуска, лицо сотрудника нельзя забыть дома, передать другому человеку без риска обнаружения или заменить простой копией учётной записи.

Актуальность задачи определяется ростом требований к контролю доступа на предприятиях и объектах с ограниченным режимом. Традиционные способы аутентификации зависят от человеческого фактора: пароль может быть раскрыт, карта доступа потеряна, а проверка вручную занимает время. Использование распознавания

лица позволяет автоматизировать проверку личности: камера получает изображение сотрудника, модель формирует числовой вектор признаков, а система сравнивает его с эталонами в базе данных и принимает решение о допуске.

В работе рассматривается программный модуль биометрической аутентификации сотрудников. Он предназначен для включения в систему идентификации персонала в помещении и решает практическую задачу безопасного входа по двум факторам: учётным данным и изображению лица. Теоретической основой послужили современные подходы к глубокому распознаванию лиц, в том числе свёрточные сети для извлечения признаков [3] и остаточное обучение, предложенное К. Хэ и соавторами [2].

1. Постановка задачи

Целью работы является разработка программы, которая выполняет регистрацию и последующую аутентификацию сотрудников по изображению лица. Пользователь должен иметь возможность создать запись в системе, сохранить пароль и набор биометрических признаков, а затем проходить проверку при входе. Программа должна быть реализована на языке Python, использовать модель свёрточной нейронной сети для формирования векторов признаков и хранить сведения о сотрудниках в документоориентированной базе MongoDB.

К системе предъявляются следующие функциональные требования. Во-первых, при регистрации сотрудник указывает табельный номер и пароль, после чего проходит биометрическую регистрацию с камеры. Во-вторых, авторизация должна быть двухфакторной: сначала проверяются табельный номер и пароль, затем выполняется сравнение лица с сохранёнными векторами. В-третьих, конфиденциальные данные должны храниться в защищённом виде: пароль сохраняется не как открытый текст, а как результат криптографического преобразования, а векторы признаков шифруются перед записью в базу. В-четвёртых, при успешной аутентификации система должна обновлять биометрический профиль сотрудника, чтобы учитывать естественные изменения внешности.

При проектировании учитывалось, что система работает не с произвольным каталогом изображений, а с ограниченным набором

сотрудников. Поэтому важны не только средние метрики модели, но и предсказуемое поведение интерфейса при ошибках: отсутствие лица, несколько лиц в кадре, неверный пароль или попытка повторной регистрации одного табельного номера должны приводить к понятному отказу без записи некорректных данных.

Основная вычислительная задача сводится к построению отображения

$$f: I \rightarrow \mathbb{R}^d, \quad (1)$$

которое каждому изображению лица I сопоставляет вектор признаков фиксированной размерности d . Далее для текущего вектора и эталонных векторов сотрудника вычисляется мера сходства. Если максимальное значение выше заданного порога, личность считается подтверждённой, иначе доступ блокируется.

2. Модель распознавания лица

В качестве основы модели выбрана свёрточная нейронная сеть с остаточными связями. Такой подход позволяет строить более глубокую архитектуру без резкого ухудшения качества обучения: внутри остаточного блока сеть обучает не всё преобразование целиком, а поправку к уже имеющемуся представлению. Вход блока передаётся к выходу напрямую и складывается с результатом нескольких свёрточных слоёв. Поэтому блок можно описать как

$$y = F(x) + x, \quad (2)$$

где x — входное представление, $F(x)$ — преобразование, изучаемое свёрточными слоями, а y — выход блока. Если размерности не совпадают, применяется дополнительная свёртка 1×1 , приводящая данные к нужной форме.

Модель последовательно извлекает признаки разного уровня. Начальные слои выделяют простые визуальные элементы, например границы и контуры. Более глубокие блоки формируют признаки, связанные с геометрией лица и устойчивыми особенностями изображения. После основных свёрточных этапов применяется глобальное усреднение, а затем линейный слой, который формирует итоговый вектор признаков. Этот вектор используется не как метка класса, а как компактное числовое описание лица сотрудника.

Для обучения применялась функция потерь ArcFace, предложенная Ц. Дэном и соавторами [1]. Её идея состоит в том, чтобы увеличить угловой зазор между классами и сделать векторы одного человека более компактными в пространстве признаков. Сначала вектор признаков x_i и матрица весов классов W_j нормализуются, после чего вычисляется косинус угла между ними:

$$\cos(\theta_j) = x_i^T W_j. \quad (3)$$

Для правильного класса вводится угловой отступ m , а итоговые логиты имеют вид

$$z_j = \begin{cases} s \cos(\theta_j + m), & j = y, \\ s \cos(\theta_j), & j \neq y, \end{cases} \quad (4)$$

где s — коэффициент масштабирования, y — индекс правильного класса. В реализации использовались значения $s = 30$ и $m = 0,5$. После вычисления логитов применяется перекрёстная энтропия. В результате сеть учится формировать векторы, для которых внутриклассовое расстояние уменьшается, а межклассовое расстояние возрастает.

При аутентификации сравнение выполняется с помощью косинусного сходства

$$\text{similarity} = \cos(x_1, x_2) = \frac{x_1^T x_2}{\|x_1\| \cdot \|x_2\|}, \quad (5)$$

где x_1 и x_2 — входные вектора признаков. Эта мера удобна для биометрических векторов, так как учитывает направление в пространстве признаков и слабо зависит от масштаба. В программе установлен порог подтверждения личности 70 %. Если ни один сохранённый вектор сотрудника не даёт достаточного сходства, вход считается неуспешным.

3. Программная реализация

Разработанная программа состоит из трёх основных блоков: графического интерфейса, нейросетевого модуля и базы данных. Интерфейс реализован на Python с использованием библиотеки

Регистрация

Первый фактор

* Табельный номер

* Пароль

* Повторение пароля

Биометрия

Рис. 1. Окно регистрации сотрудника в системе.

FreeSimpleGUI. В рабочем окне выделены области для видеопотока, регистрации и авторизации. Такой интерфейс позволяет оператору быстро запускать нужный сценарий, видеть сообщения об ошибках и контролировать результат проверки.

Регистрация сотрудника начинается с ввода табельного номера, пароля и подтверждения пароля. Табельный номер используется как уникальный идентификатор записи в базе. Затем включается камера, и в течение 40–50 секунд система формирует набор из 20 изображений лица. Снимки получают при разных небольших изменениях положения головы, мимики и освещения, что повышает устойчивость дальнейшего сравнения. Каждое изображение проходит предварительную обработку, передаётся в нейронную сеть, а полученный вектор временно хранится в памяти до завершения регистрации. На рис. 1 показано окно регистрации сотрудника с областью видеопотока и полями первого фактора проверки.

После успешного сбора данных запись сохраняется в MongoDB.

В базе хранятся табельный номер, данные для проверки пароля и набор зашифрованных векторов лица. Пароль обрабатывается функцией PBKDF2 с использованием соли и хэширования SHA-256. Биометрические векторы шифруются алгоритмом AES, а ключ не помещается в исходный код программы и задаётся через переменные окружения. Такой подход уменьшает риск раскрытия данных даже при несанкционированном доступе к базе.

Аутентификация выполняется в два этапа. Сначала сотрудник вводит табельный номер и пароль. Система находит запись, повторно применяет PBKDF2 с сохранённой солью и сравнивает полученное значение с хэшем из базы. Если проверка не пройдена, дальнейшая биометрическая процедура не запускается. При успешной проверке открывается окно распознавания лица, активируется камера и в течение нескольких секунд формируется текущий вектор признаков.

На втором этапе текущий вектор сравнивается с расшифрованными эталонными векторами данного сотрудника. Если в кадре обнаружено несколько лиц, процедура блокируется, поскольку в такой ситуации нельзя однозначно связать изображение с вводимыми учётными данными. Если лицо одно и максимальное косинусное сходство превышает порог, интерфейс сообщает, что личность подтверждена, и разрешает вход. Если порог не достигнут, выводится отказ, предотвращающий доступ постороннего лица. Пример успешной биометрической проверки приведён на рис. 2.

Важной особенностью реализации является адаптивное обновление биометрического профиля. После успешной аутентификации система может сохранить новый вектор признаков сотрудника. Если количество сохранённых векторов меньше установленного предела, новый вектор добавляется. Если предел достигнут, система заменяет наименее подходящий старый вектор. За счёт этого профиль постепенно отражает изменения внешности, освещения и условий съёмки, но не разрастается бесконтрольно.

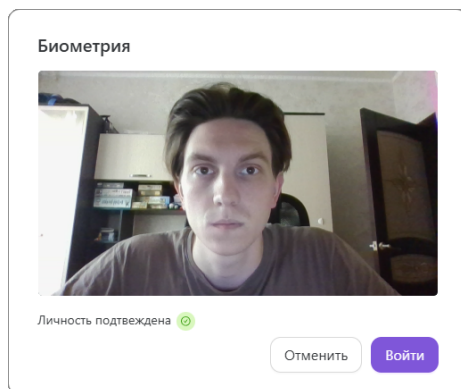


Рис. 2. Окно биометрической аутентификации при подтверждении личности.

4. Результаты обучения и проверки

Для обучения модели использовался набор данных VGGFace2. В рабочей выборке было 540 классов и 198 000 изображений. Данные были разделены на обучающую, валидационную и тестовую части в пропорции 80 %, 15 % и 5 %. Обучение выполнялось в течение 20 эпох при размере пакета 128 изображений. Общее время обучения составило приблизительно 9 часов.

В ходе обучения были получены следующие значения: потери на обучающей выборке составили 8,6 %, потери на валидационной выборке — 19,4 %. Точность на обучении достигла 99,4 %, а на валидации — 61,2 %. Разница между этими значениями показывает, что модель хорошо запоминает обучающие данные, но требует дальнейшей настройки для повышения устойчивости на новых изображениях. При этом тестирование сценария сравнения векторов при пороговом значении подтверждения в 70 % показало более высокое качество по метрикам точности, полноты и F_1 -меры.

В табл. 1 приведены итоговые показатели проверки модели в задаче подтверждения личности.

Таблица 1. Метрики качества биометрической проверки.

Метрика	Значение
Точность	91 %
Полнота	92 %
F_1 -мера	92 %

Полученные результаты позволяют сделать вывод, что выбранная архитектура подходит для прототипа системы контроля доступа. Остаточные связи обеспечивают эффективное извлечение признаков, ArcFace повышает разделимость классов, а хранение нескольких векторов на сотрудника снижает зависимость от одного эталонного снимка. Вместе с тем валидационная точность показывает необходимость дальнейшей оптимизации: увеличения разнообразия обучающих изображений, более тщательной настройки порога и проверки модели в реальных условиях пункта досмотра.

Заключение

В результате работы разработан программный модуль биометрической аутентификации сотрудников по изображению лица. В модуле реализованы регистрация пользователя, двухфакторная авторизация, извлечение биометрических признаков с помощью свёрточной нейронной сети, сравнение векторов по косинусному сходству и адаптивное обновление профиля сотрудника после успешной проверки. Для защиты данных используются хэширование паролей и шифрование биометрических векторов.

Практическая значимость программы состоит в том, что она может быть включена в систему контроля доступа предприятия и уменьшить зависимость от пропусков и паролей. Дальнейшее развитие работы связано с расширением обучающей выборки, тестированием на видеопотоке в разных условиях освещения, подбором более устойчивого порога, добавлением проверки живости лица и оптимизацией модели для работы на оборудовании пункта досмотра в режиме реального времени.

Список литературы

- [1] ArcFace: Additive Angular Margin Loss for Deep Face Recognition / J. Deng, J. Guo, N. Xue, S. Zafeiriou // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA : IEEE Computer Society, 2019. — P. 4690–4699. — DOI: <https://doi.org/10.1109/CVPR.2019.00482>.
- [2] Deep Residual Learning for Image Recognition / K. He, X. Zhang, S. Ren, J. Sun // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA : IEEE Computer Society, 2016. — P. 770–778. — DOI: <https://doi.org/10.1109/CVPR.2016.90>.
- [3] Parkhi, O. M. Deep Face Recognition / O. M. Parkhi, A. Vedaldi, A. Zisserman // Proceedings of the British Machine Vision Conference (BMVC). — Durham : BMVA Press, 2015. — P. 41.1–41.12. — DOI: <https://doi.org/10.5244/C.29.41>.

Библиографическая ссылка

Корепанов, М. В. Разработка программы биометрической аутентификации сотрудников с помощью искусственного интеллекта // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 93–101.

Сведения об авторах

КОРЕПАНОВ МАКСИМ ВЛАДИСЛАВОВИЧ

Студент магистратуры

Направление «Прикладная математика и информатика»

УДК 550.822.52, 550.822.53, 550.822.55, 519.246.8

Статистические методы и алгоритмы автоматической верификации результатов термобарокамерных испытаний геофизического оборудования

Кухар Н. В.

Тверской государственной университет

Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа С. В. Архиповым

Аннотация. В работе рассматривается задача автоматической верификации результатов термобарокамерных испытаний геофизического оборудования. Предложен программный подход, включающий определение состава прибора по структуре входного файла, выбор необходимых методов контроля и формирование итогового заключения. Реализованы алгоритмы проверки спектральных коэффициентов, гамма-каналов и нейтронных счётчиков, а также формирование человекочитаемого отчёта.

Ключевые слова: геофизическое оборудование, автоматическая верификация, статистический анализ, обработка данных, нейтрон-нейтронный каротаж.

Введение

Геофизические методы широко применяются при поиске и разведке полезных ископаемых, а также при бурении скважин. Достоверность получаемой информации в значительной степени определяется точностью и устойчивостью работы измерительной аппаратуры. Одним из видов такой аппаратуры являются приборы каротажа в процессе бурения, функционирующие в условиях повышенных температур и давлений и обеспечивающие получение данных непосредственно в процессе бурения [5, 6]. Поэтому перед вводом в эксплуатацию такие приборы проходят специальные испытания, позволяющие оценить их работоспособность в условиях, приближенных к реальным.

Результаты подобных испытаний представляют собой многоколлонные временные ряды, содержащие спектральные коэффициенты, гамма-каналы, нейтронные счётчики и служебные параметры. Для статистической обработки таких данных используются методы анализа временных рядов, общие статистические процедуры обработки экспериментальных данных и показатели вариабельности сигнала [1–4]. При этом для разных каналов применяются разные критерии оценки, а ручная проверка большого объёма данных оказывается трудоёмкой и зависит от опыта конкретного специалиста. В связи с этим возникает задача разработки методов и алгоритмов автоматической верификации результатов испытаний.

Целью данной работы является разработка программного средства, обеспечивающего автоматическое определение состава прибора, выбор необходимых методов контроля и формирование итогового заключения по результатам испытания. В рамках работы реализованы алгоритмы анализа спектральных коэффициентов, гамма-зондов и нейтрон-нейтронного каротажа, а также механизм формирования человекочитаемого отчёта.

1. Определение типа прибора и активных каналов

Первым этапом автоматического анализа является определение типа прибора и состава его рабочих каналов по структуре входного файла. В рассматриваемой задаче используются приборы 172 и 121 диаметра, отличающиеся по своей компоновке, набору датчиков и составу регистрируемых сигналов. Следовательно, перед применением методов контроля необходимо установить, с каким именно прибором ведётся работа и какие данные в файле являются информативными.

Для числового поля файла признаком информативности является ненулевая сумма значений:

$$\sum_{i=1}^N x_i \neq 0, \quad (1)$$

где x_i — значение анализируемого поля в i -й строке, а N — число строк в файле. Если условие не выполняется, поле не участвует в последующем анализе. Для нечисловых полей, например даты и времени, достаточно самого факта их наличия.

Дополнительная сложность связана с тем, что прибор делит окружающее пространство на 16 азимутальных секторов. В условиях барокамеры прибор находится в фиксированном положении, поэтому данные по каждому датчику и зонду записываются не во все возможные поля, а только в часть из них. По этой причине недостаточно просто знать общий тип прибора: необходимо определить, в какие именно поля файла записываются данные конкретных датчиков и зондов в данном испытании.

Для решения этой задачи был разработан отдельный программный класс, выполняющий функции классификатора. Его работа основана на анализе структуры входного файла и поиске ненулевых полей, относящихся к заданным семействам каналов. После этого найденные каналы группируются по типам, что позволяет определить состав прибора и выбрать набор методов, которые должны быть применены на последующих этапах анализа.

2. Проверка спектральных коэффициентов

Важной частью автоматической верификации является анализ спектральных коэффициентов, характеризующих положение и устойчивость спектра в процессе испытания. К числу таких параметров относятся коэффициент строба STRB, коэффициент бария BARD и коэффициент усиления AMP0. Для каждого из них используется собственный допустимый диапазон значений, соответствующий нормальному режиму работы спектрального тракта.

Общая логика проверки спектральных коэффициентов основана на двух последовательных этапах. На первом этапе определяется момент выхода параметра в рабочий режим. Для этого ищется первый участок длиной k , на котором значение коэффициента в течение k последовательных записей находится в допустимом диапазоне:

$$x_i \in [x_{\min}; x_{\max}], \quad i = i_0, \dots, i_0 + k - 1. \quad (2)$$

Такой подход позволяет исключить начальный переходный участок, на котором спектр ещё не стабилизировался и не должен участвовать в итоговой оценке.

На втором этапе анализируются все последующие записи. Если после выхода в рабочий режим коэффициент продолжает оставаться

в допустимом диапазоне,

$$x_i \in [x_{\min}; x_{\max}], \quad (3)$$

то запись считается корректной. Если же значение выходит за установленные пределы, строка относится к браку. В результате для каждого коэффициента формируется итоговая сводка, включающая число пропущенных строк до стабилизации, число проверенных строк и число строк с отклонениями.

С физической точки зрения коэффициенты STRB и BARD связаны с положением характерных участков спектра. При нормальной работе их значения должны находиться в пределах заранее заданных интервалов. Коэффициент АМРО характеризует работу усилительного тракта и также должен оставаться в допустимом числовом диапазоне. Таким образом, проверка спектральных коэффициентов позволяет автоматически выявлять нарушения энергетической калибровки и нестабильность спектрального тракта.

3. Проверка гамма-зондов и счётчиков

Для гамма-зондов и счётчиков используется температурный метод контроля, основанный на сравнении среднего уровня сигнала в двух характерных режимах испытания. В качестве первой опорной точки выбирается временной интервал, начинающийся после достижения заданной начальной температуры, а в качестве второй — интервал, соответствующий максимальной температуре, зафиксированной в рассматриваемом файле.

Если значения сигнала на первом окне обозначить через y_1, \dots, y_n , а на втором — через z_1, \dots, z_m , то для каждого канала вычисляются средние значения

$$\bar{y}_{\text{start}} = \frac{1}{n} \sum_{i=1}^n y_i, \quad \bar{y}_{\text{max}} = \frac{1}{m} \sum_{i=1}^m z_i. \quad (4)$$

Затем определяется относительное отклонение между этими двумя характеристиками:

$$\delta = \left| \frac{\bar{y}_{\text{max}} - \bar{y}_{\text{start}}}{\bar{y}_{\text{start}}} \right| \cdot 100 \%. \quad (5)$$

Если рассчитанное отклонение не превышает установленный допустимый процент, канал считается прошедшим проверку. Если же

$$\delta > \delta_{\max}, \quad (6)$$

то это свидетельствует о температурной нестабильности канала. В результате для каждого гамма-зонда или счётчика формируется отдельное заключение, содержащее средние значения на обеих температурных точках, рассчитанное отклонение и итог проверки.

Преимущество данного метода состоит в том, что он даёт формализованную оценку температурного дрейфа гамма-каналов. Тем самым обеспечивается сопоставление поведения канала в начале рабочего температурного диапазона и при максимальной температурной нагрузке. В прикладной геофизике подобные гамма- и гамма-гамма методы используются для оценки свойств разреза и контроля параметров среды, что подтверждает практическую значимость автоматизации их анализа [5].

4. Проверка нейтрон-нейтронного каротажа

Для каналов нейтрон-нейтронного каротажа используются два уровня анализа. Первый уровень представляет собой формальный критериальный контроль, основанный на расчёте минутных средних значений большого и малого счётчиков. Второй уровень является дополнительной статистической оценкой и предназначен для выявления подозрительных участков сигнала даже в тех случаях, когда формальные пороговые условия не нарушены.

На этапе критериального контроля исходные данные агрегируются по минутным интервалам. Для каждой минуты при числе записей n рассчитываются минутная сумма и минутное среднее:

$$S_t = \sum_{i=1}^n x_i, \quad \bar{x}_t = \frac{1}{n} \sum_{i=1}^n x_i. \quad (7)$$

Для большого счётчика используется порог

$$\bar{x}_t^{(B)} > 20, \quad (8)$$

а для малого счётчика — порог

$$\bar{x}_t^{(M)} > 5. \quad (9)$$

Если минутное среднее превышает соответствующее значение, канал считается не прошедшим проверку по формальному критерию.

Однако одного порогового контроля недостаточно, поскольку сигнал может сохраняться в допустимых пределах, но при этом демонстрировать нестабильное статистическое поведение. По этой причине в работе введён дополнительный этап анализа, в рамках которого исследуются характеристики временного ряда по минутным окнам. Для каждого минутного интервала дополнительно рассчитываются доли секунд, в которых наблюдаются два и более, а также три и более импульса:

$$p_{\geq 2,t} = \frac{\#\{x_i \geq 2\}}{n}, \quad p_{\geq 3,t} = \frac{\#\{x_i \geq 3\}}{n}. \quad (10)$$

Кроме того, определяется внутриминутный разброс:

$$\sigma_t = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_t)^2}. \quad (11)$$

Для оценки динамики сигнала между соседними минутами используется абсолютная разность минутных средних:

$$d_t = |\bar{x}_t - \bar{x}_{t-1}|. \quad (12)$$

Дополнительно применяется фактор Фано,

$$F_t = \frac{\text{Var}(S)}{\text{Mean}(S)}, \quad (13)$$

который рассматривается как показатель относительной вариабельности счётного процесса [7]. Для уменьшения влияния отдельных выбросов используется робастная нормировка, основанная на медиане и медианном абсолютном отклонении:

$$z_t = \frac{m_t - \text{median}(m)}{1,4826 \cdot \text{MAD}(m)}, \quad (14)$$

$$\text{MAD}(m) = \text{median}(|m_t - \text{median}(m)|).$$

Далее по нескольким статистическим признакам формируется суммарная оценка отклонения. Если она превышает заданный уровень, соответствующая минута помечается как подозрительная. Соседние подозрительные минуты объединяются в интервалы, которые затем включаются в итоговый отчёт как участки, требующие дополнительного внимания.

Таким образом, предложенный подход сочетает формальный критериальный контроль и дополнительную статистическую диагностику. Это позволяет не только фиксировать явные превышения допустимых уровней, но и выделять скрытые нестабильные участки сигнала, которые могут быть важны при интерпретации результатов испытаний.

5. Программная реализация и пример результата

Разработанное решение реализовано на языке Python. Для обработки входных данных используется библиотека `Pandas`, обеспечивающая чтение табличных файлов, преобразование данных к числовому виду, группировку по временным интервалам и вычисление необходимых статистических характеристик. Выбор данного инструментария обусловлен тем, что результаты испытаний представляют собой многоколонные временные ряды, для которых требуется удобная и надёжная обработка [1–4].

Логика программы построена по модульному принципу. Для каждого типа анализа реализован отдельный класс: определение состава прибора, проверка спектральных коэффициентов, анализ гамма-зондов и счётчиков, а также проверка каналов нейтрон-нейтронного каротажа. Работа программы начинается с анализа структуры входного файла. Далее главный управляющий модуль автоматически выбирает необходимые методы контроля и передаёт найденные каналы в соответствующие классы анализа. После завершения всех проверок результаты объединяются в единую структуру данных, содержащую как численные показатели, так и итоговые логические заключения.

Работа алгоритма была проверена на реальном файле испытания объёмом 83 646 строк. В ходе анализа программа определила прибор 172 диаметра, выделила активные каналы `RLDL8`, `RSDL8`, `RFTN8`,

RNTN8, GK18, GK28 и GK38, а также служебные поля STRB, BARD и AMP0. По итогам проверки коэффициенты STRB и AMP0 прошли контроль, а коэффициент BARD не прошёл проверку из-за 120 строк с выходом за допустимый диапазон. Все анализируемые гамма-каналы прошли двухточечный температурный контроль. Для нейтронных счётчиков формальный пороговый критерий также не был нарушен, однако дополнительная статистическая диагностика выявила 101 подозрительную минуту для большого счётчика и 16 подозрительных минут для малого счётчика, которые были объединены в интервалы и отражены в человекочитаемом отчёте.

Следовательно, программа формирует не только внутренний машинный отчёт, пригодный для дальнейшей обработки, но и человекочитаемый текстовый документ, в котором указываются состав найденных каналов, результаты проверки каждого параметра, наличие подозрительных участков и итоговое заключение по соответствующим методам контроля.

Заключение

В работе рассмотрена задача автоматической верификации результатов испытаний геофизического оборудования в условиях повышенной температуры и давления. Основное внимание было уделено формализации критериев проверки для различных типов каналов и параметров, присутствующих во входных файлах испытаний.

В ходе работы был реализован алгоритм определения типа прибора и активных каналов по структуре файла, что позволяет автоматически устанавливать состав используемых датчиков и выбирать необходимые методы контроля. Разработаны методы проверки спектральных коэффициентов, гамма-зондов и счётчиков, а также каналов нейтрон-нейтронного каротажа. Для нейтронных каналов, помимо формального порогового критерия, реализована дополнительная статистическая оценка, позволяющая выявлять подозрительные участки сигнала.

Все разработанные методы объединены в единое программное средство, формирующее как машинный, так и человекочитаемый итоговый отчёт. Полученный результат показывает, что автоматический анализ позволяет систематизировать проверку различных

параметров прибора и сделать итоговое заключение более воспроизводимым и удобным для практического использования. Дальнейшее развитие работы может быть связано с расширением набора критериев анализа, углублением обработки спектральных данных и созданием графической оболочки для более удобного взаимодействия с программой.

Список литературы

- [1] Афанасьев, В. Н. Анализ временных рядов и прогнозирование : учебник / В. Н. Афанасьев, М. М. Юзбашев. — М. : Финансы и статистика, 2001. — 228 с. — URL: [https://dl.booksee.org/genesis/47000/721dafb90d4980a71aa359ea5d5d287a/_as/\[Afanasev_V.N.,_YUzbashev_M.M.\]_Analiz_vremennueh_\(BookSee.org\).pdf](https://dl.booksee.org/genesis/47000/721dafb90d4980a71aa359ea5d5d287a/_as/[Afanasev_V.N.,_YUzbashev_M.M.]_Analiz_vremennueh_(BookSee.org).pdf).
- [2] Введение в анализ временных рядов : учебное пособие для вузов / Н. В. Артамонов, Е. А. Ивин, А. Н. Курбацкий, Д. Фантащини. — Вологда : ВолНИЦ РАН, 2021. — 134 с. — URL: <https://clc.li/AsRxx>.
- [3] Шорохова, И. С. Статистические методы анализа : учебное пособие / И. С. Шорохова, Н. В. Кисляк, О. С. Мариев. — Екатеринбург : Изд-во Урал. ун-та, 2015. — 300 с. — URL: https://elar.urfu.ru/bitstream/10995/36122/1/978-5-7996-1633-5_2015.pdf.
- [4] Осташев, В. Б. Первичные методы обработки результатов эксперимента: теория погрешности, графическое представление данных, аппроксимация зависимостей, оформление отчёта : учебное пособие / В. Б. Осташев, С. В. Хотунцова. — СПб. : СПбГТИ(ТУ), 2021. — 81 с. — URL: <http://ostashev vb.spb.ru/Download/Errors.pdf>.
- [5] Гусев, Е. В. Методы полевой геофизики : учебное пособие / Е. В. Гусев. — Томск : Изд-во Томского политехнического университета, 2012. — 216 с. — URL: https://portal.tpu.ru/files/departments/publish/IPR_Gusev.pdf.
- [6] Каротаж во время бурения: что это такое и как он работает // Sinodrills : [сайт]. — 2025. — 6 июля. — URL: <https://www.sinodrills.com/ru/logging-while-drilling/>.

- [7] Rajdl, K. Fano factor: a potentially useful information / K. Rajdl, P. Lansky, L. Kostal // *Frontiers in Computational Neuroscience*. — 2020. — Vol. 14. — Art. 569049. — DOI: <https://doi.org/10.3389/fncom.2020.569049>.

Библиографическая ссылка

Кухар, Н. В. Статистические методы и алгоритмы автоматической верификации результатов термобарокамерных испытаний геофизического оборудования // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 102–111.

Сведения об авторах

КУХАР НИКИТА ВЛАДИМИРОВИЧ

Студент магистратуры

Направление «Прикладная математика и информатика»

УДК 338.1, 004.9

Анализ структурной трансформации топливно-энергетического баланса России

Литвиненко А. С.

Тверской государственной университет

Рекомендовано к публикации доцентом кафедры математической статистики и системного анализа А. А. Смородовой

Аннотация. В статье рассматривается методологический подход к оценке структурных преобразований в топливно-энергетическом балансе России. На основе данных Росстата за 2005–2021 годы проведён анализ с использованием традиционных индексных методов, выявлены основные тенденции трансформации. Предложен авторский индекс структурной модернизации, позволяющий количественно оценивать направление и интенсивность изменений через соотношение долей преобразованной и сырьевой энергии. Разработан автоматизированный программный инструмент, интегрирующий расчёт традиционных показателей, авторского индекса и сценарное прогнозирование. Апробация инструмента подтвердила его эффективность для оперативной диагностики и поддержки управленческих решений.

Ключевые слова: топливно-энергетический баланс, структурные сдвиги, индекс структурной модернизации, программный анализ, энергетическая статистика.

Введение

Топливо-энергетический комплекс (ТЭК) выступает стратегическим фундаментом экономики Российской Федерации, обеспечивая энергетическую безопасность и значительную долю бюджетных доходов. В условиях глобальной энергетической трансформации и санкционного давления вопросы структурной адаптации топливно-энергетического баланса (ТЭБ) приобретают критическое значение [3]. Существующие подходы к анализу ТЭБ преимущественно носят описательный характер, ограничиваются сопоставлением абсолютных объёмов и не обеспечивают комплексной количественной

оценки направления структурных преобразований [1]. Отсутствие специализированных алгоритмических решений, адаптированных под формат российской статистической отчётности, создаёт методологический и технологический пробел, сдерживающий качество аналитической поддержки принятия управленческих решений. Цель исследования — разработка автоматизированного программного инструмента для анализа структурных изменений в ТЭБ России и выявление ключевых тенденций его трансформации на основе нового авторского индексного подхода.

1. Индексный подход анализа ТЭБ

В экономике используется большое количество индексов, но в рамках анализа ТЭБ они почти не применяются. Предложенный подход будет содержать лишь несколько подходящих индексов для анализа ТЭБ. Анализ базируется на расчёте структурных долей и их использовании в индексных показателях. В работе использованы данные Росстата за период 2005–2021 годы. Структура добычи анализируется через расчёт долей энергоресурсов:

$$S_{i,t} = \frac{Q_{i,t}}{\sum_j Q_{j,t}}, \quad \sum_i S_{i,t} = 1,$$

где $Q_{i,t}$ — объём добычи i -го ресурса в году t .

Для оценки интенсивности структурных изменений применён индекс Салаи:

$$I_S = \frac{1}{2} \sum_i |S_{i,t} - S_{i,0}|.$$

Чем ближе значение индекса к 1, тем существеннее произошедшие структурные изменения.

Для измерения концентрации — индекс Херфиндала – Хиршмана:

$$\text{HHI}_t = \sum_i S_{i,t}^2.$$

Чем выше концентрация, тем меньшее число конкурентов представлено на рынке, и тем ближе рынок к монополии.

Результаты применения индексного подхода показал следующее. За исследуемый период доля нефти в структуре добычи снизилась

на 3,4 %, тогда как доля угля возросла на 4,2 %, что свидетельствует о развитии угольной промышленности при относительной стабильности газовой отрасли. Экспортная зависимость по нефти сократилась с 54 % до 44 %, по газу — с 32 % до 27 %, однако доля экспортируемого угля увеличилась с 29 % до 54 %. Индекс Салаи составил 0,0417, что указывает на слабые структурные сдвиги, а снижение ННИ с 0,402 до 0,380 подтвердило тенденцию к диверсификации добычи. Несмотря на информативность, существующие индексы не позволяют оценить направление модернизации экономики и требуют значительных ручных усилий для обработки многолистных форм Росстата.

2. Авторский индекс структурной модернизации ТЭБ

Для преодоления описанных ограничений предложен авторский индекс структурной модернизации ($I_{см}$), который оценивает не только интенсивность, но и вектор структурных изменений через соотношение глубины переработки на этапах добычи и конечного потребления. Методика основана на разделении всех энергоресурсов ТЭБ на две категории: сырое топливо (первичные ресурсы: уголь, нефть, природный газ) и преобразованная энергия (вторичные ресурсы: электроэнергия, теплоэнергия, нефтепродукты, кокс).

На первом этапе рассчитываются коэффициенты преобразования в добыче ($R_{доб}$) и потреблении ($R_{потр}$):

$$R_{доб} = \frac{\sum(\text{Преобразованная энергия в производстве})}{\sum(\text{Сырое топливо в добыче})},$$

$$R_{потр} = \frac{\sum(\text{Преобразованная энергия в потреблении})}{\sum(\text{Сырое топливо в потреблении})}.$$

Авторский индекс определяется как отношение этих коэффициентов:

$$I_{см} = \frac{R_{потр}}{R_{доб}}.$$

Значение $I_{см} > 1$ указывает на высокую структурную модернизацию: экономика потребляет внутри страны больше преобразованной энергии, чем добывает в такой форме, что свидетельствует о

развитой перерабатывающей промышленности и технологическом развитии. Значение $I_{\text{CM}} \approx 1$ характеризует сбалансированную структуру, а $I_{\text{CM}} < 1$ — низкую структурную модернизацию и усиление сырьевой зависимости.

Применение авторского индекса к данным Росстата за 2005–2021 годы [2] показало устойчивое значение $I_{\text{CM}} > 1$, что подтверждает тенденцию к внутренней структурной модернизации: российская экономика направляет значительную часть добываемого сырья на экспорт, оставляя для внутреннего потребления более качественные преобразованные энергоносители. Динамика индекса во времени, как показано на рис. 1, позволяет выявлять точки роста перерабатывающих отраслей и объективно оценивать эффективность государственной энергетической политики, что невозможно при использовании исключительно описательных или традиционных индексных подходов.

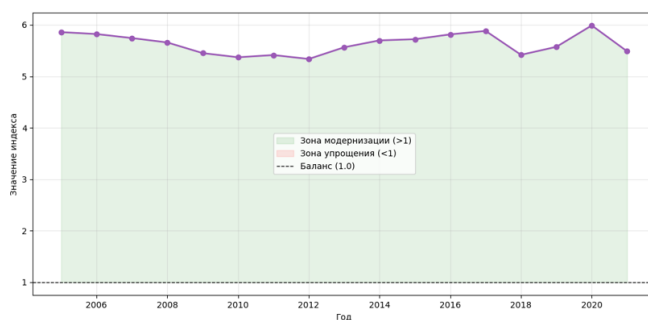


Рис. 1. Динамика индекса структурной модернизации.

3. Программная реализация аналитического инструмента

Для автоматизации предложенной методики и традиционных расчётов разработан специализированный программный комплекс «ТЭБ Анализатор» на языке Python. Архитектура решения построена по модульному принципу и включает четыре основных компонента: загрузка и парсинг данных, расчёт аналитических показателей, визуализация и генерация отчётов.

Модуль загрузки данных реализует автоматический импорт многолистовых Excel-книг Росстата, извлечение ключевых показателей по заданным маркерам и приведение данных к единому формату с проверкой корректности. Модуль расчётов интегрирует алгоритмы стандартного и индексного анализа (межотраслевой анализ, структурные доли, индексы Салаи и Херфиндаля – Хиршмана, индекс структурной модернизации, декомпозиция сдвигов и др.). Для прогнозирования реализовано ансамблевое моделирование, усредняющее результаты линейной регрессии, полиномиальной аппроксимации, экспоненциального сглаживания и Ridge-регрессии с фильтрацией выбросов по правилу $\pm 2\sigma$. Сценарное моделирование учитывает внешние шоки через коэффициенты тренда и волатильности:

$$F_s = F_{\text{base}} \cdot k_{\text{tr}} \pm \sigma \cdot k_{\text{vol}}.$$

Инструмент для анализа «ТЭБ Анализатор» представляется в виде программы с интуитивно понятным интерфейсом, отображённом на рис. 2, в котором: 1 – кнопка загрузки данных; 2 – поле для выбора построенного графика; 3 – поле для выбора метода; 4 – кнопки управления программой; 5 – поле для визуализации графиков; 6 – редактируемое поле для аналитических выводов.

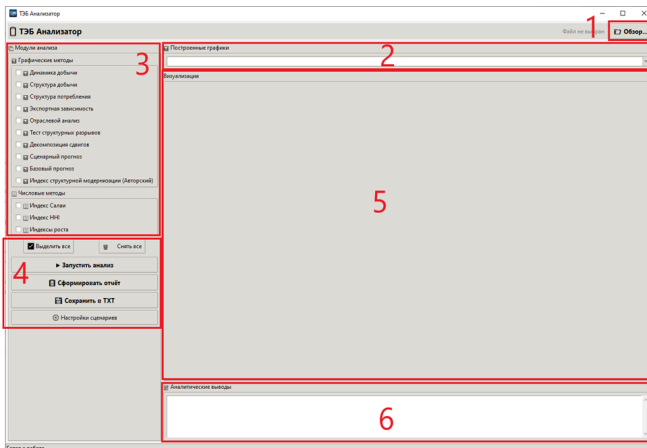


Рис. 2. Интерфейс программного инструмента.

Автоматическая генерация текстовых отчётов сокращает время проведения структурного анализа ТЭК с нескольких дней до нескольких часов. Апробация инструмента на данных за 2005–2021 годы подтвердила его работоспособность, воспроизводимость результатов и практическую применимость для органов государственного управления и научно-исследовательских организаций. Перспективы развития включают интеграцию модуля формирования готовых PDF-отчётов и расширение библиотеки прогнозных моделей.

Заключение

В работе решена задача перехода от описательной статистики к алгоритмизированной диагностике структурных преобразований в топливно-энергетическом балансе России. Анализ традиционными методами за 2005–2021 годы выявил снижение доли нефти, рост доли угля, снижение экспортной зависимости по нефти и газу при резком росте экспорта угля, а также слабую интенсивность структурных сдвигов. Предложен авторский индекс структурной модернизации, заполняющий существующий методологический пробел за счёт количественной оценки направления трансформации через соотношение долей преобразованной и сырьевой энергии. Разработанный программный инструмент успешно интегрирует все этапы анализа, обеспечивая автоматизацию расчётов, сценарное прогнозирование и наглядную визуализацию. Дальнейшие исследования могут быть направлены на расширение функционала комплекса, подключение источников данных в реальном времени и адаптацию алгоритмов под региональные энергетические балансы.

Список литературы

- [1] Проворная, И. В. Структурные сдвиги в мировой системе энергопотребления // Интерэкспо Гео-Сибирь. — 2018. — Т. 2. — С. 165–172. — DOI: <https://doi.org/10.18303/2618-981X-2018-2-165-172>.
- [2] Баланс энергоресурсов Российской Федерации, 2005–2021 годы : [статистические данные] / Федеральная служба государственной статистики (Росстат) [Электронный ресурс]. — URL: <https://>

rosstat.gov.ru/storage/mediabank/balans_energo_2005-2021.xlsx.

- [3] Энергетическая стратегия Российской Федерации на период до 2035 года: утверждена распоряжением Правительства РФ от 09.06.2020 № 1523-р. — Москва, 2020. — 93 с. — URL: <http://static.government.ru/media/files/w4sigF0iDjGVDYT4IgsApssm6mZRb7wx.pdf>.

Библиографическая ссылка

Литвиненко, А. С. Анализ структурной трансформации топливно-энергетического баланса России // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 112–118.

Сведения об авторах

ЛИТВИНЕНКО АЛЕКСЕЙ СЕРГЕЕВИЧ
Студент магистратуры
Направление «Прикладная информатика»

УДК 510.643, 510.52

Сложность логики ATL^* и её фрагментов

Льянов А. А.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры математического моделирования и вычислительной математики М. Н. Рыбаковым

Аннотация. В статье рассматривается фрагмент ATL без пропозициональных переменных. Показывается, что формулы в нём семантически эквивалентны \perp или \top . Этот факт, отмеченный в [3], получает развёрнутое обоснование, а также строится линейный алгоритм проверки выполнимости. Обсуждаются принципиальные отличия константного фрагмента логики K от аналогичных фрагментов ATL , ATL^* , CTL и CTL^* . Наконец, рассматривается погружение CTL^* в ATL^* , сохраняющее константный фрагмент. **Ключевые слова:** модели вычислений, логика альтернативного времени, логика ветвящегося времени.

Введение

Логика стратегического времени ATL и ATL^* являются инструментом формальной спецификации мультиагентных систем. Они позволяют описывать свойства, зависящие от возможностей взаимодействующих агентов, например: «группа серверов может гарантировать отказоустойчивость сети».

Интерес к формальной верификации обусловлен высокой стоимостью ошибок в важных системах: авария космического аппарата «Mars Climate Orbiter» была вызвана несогласованностью единиц измерения, ракета-носитель «Ariane 5» разрушилась на сороковой секунде полёта из-за ошибки в программном обеспечении, сбой медицинского ускорителя «Therac-25» привели к тяжёлым последствиям для пациентов, а ошибка всего в одном символе привела к уничтожению космического корабля «Mariner 1» с земли из-за отклонения с курса. Это демонстрирует необходимость строгих математических методов анализа программных и аппаратных систем.

Одной из фундаментальных задач логической верификации является задача выполнимости: требуется определить, существует ли

модель, удовлетворяющая заданной спецификации. Для логик **ATL** и **ATL*** данная задача EXPTIME-полна и 2EXPTIME-полна соответственно.

Тем не менее иногда в прикладных сценариях спецификации возможно обойтись без использования пропозициональных переменных. Соответствующие *константные* формулы интересны как с практической, так и с теоретической точек зрения. Они позволяют описывать глобальные свойства систем, независимые от конкретных состояний, например неизбежность отказа системы. Кроме того, изучение константных фрагментов помогает лучше понять границы сложности самой логики.

Данная работа посвящена константному фрагменту \mathcal{L}_{ATL}^0 . Основной результат был сформулирован в [3]; мы даём его развёрнутое доказательство, строим алгоритм проверки и проводим сравнительный анализ с другими логиками. Показано, что в рассматриваемом константном фрагменте стратегические операторы не приводят к усложнению алгоритма проверки выполнимости. Предлагается конструктивный алгоритм проверки выполнимости, анализируются его вычислительные характеристики и проводится сопоставление с логиками **K**, **CTL** и **CTL***. Также показывается, что стандартное вложение **CTL*** в **ATL*** согласуется с рассмотрением константных фрагментов.

Структура статьи следующая. В первом разделе вводятся необходимые понятия и определения. Во втором разделе показывается сводимость константных формул к \perp или \top и содержится описание алгоритма проверки с обсуждением его сложности. В третьем разделе сравнение с модальной логикой **K** и погружение **CTL*** в **ATL***. Заключение содержит обсуждение результатов.

1. Определения

ОПРЕДЕЛЕНИЕ 1. *Параллельная игровая структура (англ. concurrent game structure, сокр. CGS) — это кортеж*

$$\mathfrak{G} = (\mathbb{A}\mathbb{G}, S, Act, act, \delta),$$

где:

- \mathbb{AG} — конечное множество агентов, подмножества которого называют коалициями;
- S — множество состояний;
- Act — непустое множество действий;
- $act: \mathbb{AG} \times S \rightarrow \mathcal{P}(Act) \setminus \{\emptyset\}$ — функция, задающая набор действий, доступных для выполнения каждому игроку в каждом состоянии;
- δ — функция переходов, что сопоставляет каждому состоянию s и профилю действий $\alpha = (\alpha_1, \dots, \alpha_k)$, где $\alpha_a \in act(a, s)$, последующее состояние $\delta(s, \alpha)$.

ОПРЕДЕЛЕНИЕ 2. Параллельной игровой моделью (англ. *concurrent game model*, сокр. *CGM*) называют шкалу *CGS*, снабжённую оценкой $V: PROP \rightarrow \mathcal{P}(S)$, что сопоставляет каждой пропозициональной переменной множество состояний, в которых она истинна.

Отметим, что оценка не влияет на результаты, связанные с константными фрагментами.

ОПРЕДЕЛЕНИЕ 3. Формулы логики ATL* определяются относительно множества агентов \mathbb{AG} и счётно бесконечного множества $PROP$ атомарных переменных, рекурсивно с помощью следующей грамматики:

$$\varphi ::= p \mid \perp \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid X\varphi \mid G\varphi \mid (\varphi U\varphi) \mid \langle\langle C \rangle\rangle\varphi,$$

где p — произвольный элемент $PROP$, а C — произвольный элемент $\mathcal{P}(\mathbb{AG})$, множество всех подмножеств \mathbb{AG} .

ОПРЕДЕЛЕНИЕ 4. Подмножество $ATL(\mathbb{AG}, PROP)$ формул $ATL^*(\mathbb{AG}, PROP)$ определяется следующей грамматикой:

$$\varphi ::= p \mid \perp \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \langle\langle C \rangle\rangle X\varphi \mid \langle\langle C \rangle\rangle G\varphi \mid \langle\langle C \rangle\rangle (\varphi U\varphi),$$

где p — произвольный элемент $PROP$, а C — произвольный элемент $\mathcal{P}(\mathbb{AG})$.

ОПРЕДЕЛЕНИЕ 5. Формула φ называется выполнимой, если существует модель \mathfrak{M} и состояние s такие, что $\mathfrak{M}, s \models \varphi$.

Подробно определения и семантика изложены в [2].

Обозначим через \mathcal{L}_{ATL}^0 фрагмент ATL без пропозициональных переменных, в котором формулы строятся из констант \perp, \top , булевых связок, стратегического оператора и модальностей.

2. Сложность константного фрагмента

Сначала докажем две вспомогательные леммы.

ЛЕММА 1. Пусть для АТЛ-формул φ и ψ выполняется $\varphi \equiv \psi$, то есть для любой модели \mathcal{M} и состояния s :

$$\mathcal{M}, s \models \varphi \iff \mathcal{M}, s \models \psi.$$

Тогда для любой АТЛ-формулы $F(q)$ замена одного вхождения пропозициональной переменной q на φ или ψ сохраняет эквивалентность $F[\varphi/q] \equiv F[\psi/q]$.

ДОКАЗАТЕЛЬСТВО. Докажем утверждение структурной индукцией по формуле $F(q)$. В атомарном случае $F(q)$ либо не содержит рассматриваемого вхождения q , либо следует из $\varphi \equiv \psi$. Базис выполняется. Индукционное предположение говорит об эквивалентности подформул. Доказательство булевых связок следует из семантики и индукционного предположения. Для модальностей $\langle\langle A \rangle\rangle X\chi, \langle\langle A \rangle\rangle G\chi, \langle\langle A \rangle\rangle (\chi_1 U \chi_2)$ доказательство следует из факта: если подформулы истинны в одних и тех же состояниях, то совпадает и выполнение соответствующих условий на всех допустимых исходах стратегий. Значит, $F[\varphi/q] \equiv F[\psi/q]$. \square

ЛЕММА 2 (Эквивалентность формул в логике L_{ATL}^0). Для любой модели CGM и любой коалиции C выполняются эквивалентности:

$$\begin{array}{lll} (\top \wedge \top) \leftrightarrow \top; & \langle\langle C \rangle\rangle (\top U \top) \leftrightarrow \top; & \langle\langle C \rangle\rangle G\top \leftrightarrow \top; \\ (\top \wedge \perp) \leftrightarrow \perp; & \langle\langle C \rangle\rangle (\top U \perp) \leftrightarrow \perp; & \langle\langle C \rangle\rangle G\perp \leftrightarrow \perp; \\ (\perp \wedge \top) \leftrightarrow \perp; & \langle\langle C \rangle\rangle (\perp U \top) \leftrightarrow \top; & \langle\langle C \rangle\rangle X\top \leftrightarrow \top; \\ (\perp \wedge \perp) \leftrightarrow \perp; & \langle\langle C \rangle\rangle (\perp U \perp) \leftrightarrow \perp; & \langle\langle C \rangle\rangle X\perp \leftrightarrow \perp. \end{array}$$

Для отрицания и остальных булевых связок используются обычные таблицы истинности.

ДОКАЗАТЕЛЬСТВО. Булевы связки определяются согласно таблице истинности. $\langle\langle C \rangle\rangle G\top$ истинна всюду, потому что истина всегда выполнима. Аналогично доказываются эквивалентности остальных модальностей. \square

Данных эквивалентностей достаточно, чтобы свести любую константную формулу к \perp или \top .

ТЕОРЕМА 3. *Задача выполнимости \mathcal{L}_{ATL}^0 решается за линейное время относительно размера входной формулы. Помимо этого, каждая формула эквивалентна либо \perp , либо \top .*

ДОКАЗАТЕЛЬСТВО. Индукция по построению формулы. База \perp, \top выполняется тривиально. Индукционный шаг: пусть все подформулы приведены к \perp, \top . Тогда формула является либо булевой связкой, либо модальностью. Тогда, согласно лемме 2, формула заменяется на \perp или \top .

Подробно опишем алгоритм для проверки выполнимости формулы φ :

- строится синтаксическое дерево формулы φ ;
- листьям присваиваются значения \perp или \top ;
- для каждой внутренней вершины, включая корень, вычисляется значение согласно лемме 2.

Каждая вершина обрабатывается за время $O(1)$, а число вершин не превосходит длины формулы $|\varphi|$. Следовательно, задача выполнимости решается за время $O(|\varphi|)$, то есть принадлежит классу P . \square

Аналогичное рассуждение переносится на константные фрагменты ATL^* , CTL^* , CTL и LTL , поскольку в отсутствие пропозициональных переменных значение формул также сводится к двум классам эквивалентности.

Заметим, что сложность константных фрагментов логик ATL и ATL^* , CTL^* и CTL ниже, чем у минимальной нормальной логики K , в которой константный фрагмент является PSPACE-полным [1]. Это связано с тем, что шкалы данных темпоральных логик являются *серийными*: из каждого состояния исходит хотя бы один путь. Поэтому, например, формула $\langle\langle C \rangle\rangle X \perp$ не может быть истинной. В то же время в шкале для логики K нет ограничений на достижимость. Таким образом, формула $\square \perp$ может быть истинной, если нет достижимых миров. Помимо этого, вложенные модальности в K позволяют выражать нетривиальные ограничения на структуру шкалы достижимости даже без пропозициональных переменных. Комбинация этих двух факторов отличает константный фрагмент K от константных фрагментов ATL , ATL^* , CTL и CTL^* .

3. Погружение CTL^* в ATL^*

Известно, что CTL^* [3] естественно погружается в ATL^* .

УТВЕРЖДЕНИЕ 4. Существует полиномиально вычислимая функция f , сопоставляющая каждой формуле φ языка CTL^* формулу $f(\varphi)$ языка ATL^* , такая, что для всякой модели Крипке \mathcal{K} , соответствующей ей модели CGM \mathcal{M} , и любого состояния s ,

$$\mathcal{M}, s \models f(\varphi) \iff \mathcal{K}, s \models \varphi.$$

Идея погружения состоит в том, что квантор пути E моделирует стратегическую возможность коалиции всех агентов, а квантор пути A — возможностью пустой коалиции. Для модели Крипке можно построить соответствующую модель CGM , в которой выбор действия единственного агента соответствует выбору следующего состояния среди достижимых. Тогда формулы вида $E\alpha$ переводятся в $\langle\langle AG \rangle\rangle\alpha$, а формулы вида $A\alpha$ — в $\langle\langle \emptyset \rangle\rangle\alpha$. Булевы связки и темпоральные операторы X , G , U сохраняются.

ЗАМЕЧАНИЕ 1. Поскольку логика $S4$ погружается в CTL , CTL является фрагментом CTL^* , а CTL^* погружается в ATL^* , получаем полиномиально вычислимое погружение $S4$ в ATL^* . Следовательно, ATL^* наследует выразительные свойства CTL^* и связанных с ней логик. При этом для константных фрагментов рассмотренное выше рассуждение о сведении формул к \perp или \top сохраняет линейную оценку сложности.

Заключение

В работе рассмотрен константный фрагмент ATL без пропозициональных переменных. Каждая формула этого фрагмента сводится к одному из двух значений: \perp или \top , что позволяет проверять её выполнимость за линейное время. Также описано погружение CTL^* в ATL^* посредством замены кванторов пути на коалиционные модальности. Полученные результаты позволяют выделять тривиальные спецификации и могут использоваться при предварительном анализе формул в задачах проверки моделей. Дальнейшая работа может быть направлена на исследование фрагментов ATL^* и ATL с ограниченным числом переменных и на уточнение границ их вычислительной сложности.

Список литературы

- [1] Рыбаков, М. Н. Константные формулы в модальных логиках: проблема разрешения / М. Н. Рыбаков, А. В. Чагров // Логические исследования. — 2002. — Т. 9. — С. 202–220. — URL: <https://logicalinvestigations.ru/article/view/206?lang=en>.
- [2] Demri, S. Temporal Logics in Computer Science: Finite-State Systems / S. Demri, V. Goranko, M. Lange. — Cambridge : Cambridge University Press, 2016. — 744 p. — DOI: <https://doi.org/10.1017/CB09781139236119>.
- [3] Rybakov, M. Complexity of finite-variable fragments of propositional temporal and modal logics of computation / M. Rybakov, D. Shkatov // Theoretical Computer Science. — 2022. — Vol. 925. — P. 45–60. — DOI: <https://doi.org/10.1016/j.tcs.2022.04.056>.

Библиографическая ссылка

Льянов, А. А. Сложность логики ATL* и её фрагментов // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 119–125.

Сведения об авторах

Льянов Александр Александрович
Студент магистратуры
Направление «Прикладная информатика»

УДК 550.832, 004.42

Автоматическая настройка данных ГИС в рамках петрофизической интерпретационной модели

Майоршин Р. Н.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа И. А. Багровой

Аннотация. Рассматривается метод автоматической настройки данных геофизических исследований скважин, представленных в формате LAS. Метод включает унификацию кривых, робастную обработку выбросов, модельно-ориентированную калибровку признаков, вычисление производных индексов и регуляризованную оценку долей литотипов. Приведены результаты апробации на данных работающих нефтяных скважин.

Ключевые слова: данные ГИС, LAS-файл, петрофизическая модель, калибровка, литологическая интерпретация.

Введение

Данные геофизических исследований скважин являются одним из основных источников информации о литологическом составе и свойствах пород. На практике такие данные часто передаются в формате LAS, содержащем глубину, каротажные кривые, описание скважины и служебные параметры [3]. Несмотря на стандартизованную структуру файла, значения кривых требуют подготовки перед интерпретацией: разные приборы и программные комплексы используют различные шкалы, условные единицы измерения и мнемоники кривых; кроме того, в данных встречаются пропуски и выбросы. В результате точки на кросс-плотах могут быть плохо сопоставимы с узловыми областями петрофизической модели.

В классической петрофизической интерпретации используются представления о минералогическом и литологическом составе пород и об ожидаемых физических откликах различных типов разреза [2].

Однако ручная настройка кривых под такую модель является трудоёмкой и плохо воспроизводимой процедурой. Целью работы является разработка метода автоматической настройки данных ГИС, который приводит исходные кривые к пространству выбранной интерпретационной модели и позволяет оценивать доли литотипов по глубине. Литотипом далее называется тип породы, выделяемый в модели по совокупности вещественного состава и петрофизических признаков.

1. Постановка задачи

На вход метода поступает LAS-файл с глубинной сеткой и набором кривых $x_f(d)$, где d — глубина, а f — признак, например К, Th, Са, Mg, Si или S. Требуется получить новые кривые, пригодные для интерпретации: калиброванные признаки, производные индексы и доли литотипов. В работе используется интерпретационная модель, близкая по составу к модели литосканерной технологии ядерной спектрометрии [1]: песчаник, алевролит, аргиллит, мергель, известняк, доломит, ангидрит и гипс. Для каждого литотипа заданы модельные значения параметров, в том числе содержания К, Th, U, Са, Mg, Si, S и ряд петрофизических характеристик.

Пусть $M_{j,f}$ обозначает значение признака f для литотипа j в модели. Необходимо построить преобразованные признаки $y_f(d)$ и по ним оценить вектор долей литотипов

$$v(d) = (v_1(d), \dots, v_J(d)), \quad v_j(d) \geq 0, \quad \sum_{j=1}^J v_j(d) = 1. \quad (1)$$

Здесь $J = 8$. Ограничения (1) выражают физический смысл результата: доли пород не могут быть отрицательными и должны суммироваться в единицу, то есть в 100 %.

Целевой результат настройки понимается не как простая нормализация к диапазону $[0, 1]$, а как согласование исходных кривых с параметрами выбранной интерпретационной модели. Хорошо настроенные данные должны удовлетворять трём требованиям: рабочие диапазоны признаков должны соответствовать модельным диапазонам; кросс-плоты должны становиться интерпретируемыми относительно узлов модели; рассчитанные литотипные профили должны быть сопоставимы с результатами практической интерпретации.

2. Алгоритм настройки и интерпретации

Предложенный алгоритм состоит из нескольких этапов. Сначала LAS-файл разбирается на секции, а мнемоники кривых приводятся к внутренним именам признаков. Каналом далее называется отдельная каротажная кривая, то есть последовательность измерений одного признака по глубине. Например, различные варианты обозначения кальциевого канала сопоставляются с модельным параметром Ca . После этого выполняется робастная обработка выбросов. Для каждой кривой в скользящем окне W_i вычисляются локальная медиана и медианное абсолютное отклонение

$$\begin{aligned} m_i &= \text{med}\{x_f(k) : k \in W_i\}, \\ MAD_i &= \text{med}\{|x_f(k) - m_i| : k \in W_i\}. \end{aligned} \quad (2)$$

Точка помечается как выброс, если

$$\frac{|x_f(i) - m_i|}{MAD_i} > T, \quad (3)$$

где T — порог робастного фильтра. Выбросы не удаляются из исходного файла, но исключаются из подбора параметров калибровки.

В базовом режиме калибровка выполняется по квантилям. Квантилем q_p называется значение, ниже которого находится доля p наблюдений. Вместо минимума и максимума используются $q_{0,05}$ и $q_{0,95}$, так как они слабо зависят от одиночных выбросов. Если m_f^{min} и m_f^{max} — минимальное и максимальное значения соответствующего параметра в модели, то строится линейное преобразование

$$\begin{aligned} y_f(d) &= a_f + b_f x_f(d), \\ b_f &= \frac{m_f^{max} - m_f^{min}}{q_{0,95} - q_{0,05}}, \\ a_f &= m_f^{min} - b_f q_{0,05}. \end{aligned} \quad (4)$$

Тем самым рабочий диапазон реальной кривой сопоставляется с диапазоном модельного параметра.

Дополнительно исследовался вариант калибровки по контрольным литологическим профилям. Под контрольными профилями понимаются доли литотипов, полученные в рамках применяемой

на практике методики и используемые не как обязательный вход метода, а как средство апробации и настройки. По ним вычисляется ожидаемое модельное значение признака

$$y_f^*(d) = \frac{\sum_{j=1}^J V_j(d) M_{j,f}}{\sum_{j=1}^J V_j(d)}, \quad (5)$$

где $V_j(d)$ — контрольная доля литотипа j на глубине d . Далее для пары «исходная кривая — модельный параметр» подбирается регрессионное преобразование. Рассматривались линейная регрессия, полиномиальная регрессия степени не выше трёх и регрессия по квантильным бинам. Квантильным бином называется группа наблюдений, попавших в один интервал разбиения упорядоченных значений признака. В общем виде подбирается полином

$$P_f^* = \arg \min_{P \in \mathcal{P}_3} \sum_{d \in D_f} (P(x_f(d)) - y_f^*(d))^2, \quad (6)$$

где \mathcal{P}_3 — множество полиномов степени не выше трёх. При использовании бинов в формуле (6) берутся медианные представители интервалов. Это снижает влияние плотных кластеров, то есть областей кросс-плота или диапазонов признака, где сосредоточено большое число близких по значениям точек и которые при обычной регрессии могут чрезмерно влиять на параметры модели.

После калибровки вычисляются производные индексы

$$\begin{aligned} CI &= \frac{Ca}{Ca + Si + Mg}, \\ DI &= \frac{Mg}{Ca + Mg}, \\ SI &= \frac{Si}{Ca + Si + Mg}. \end{aligned} \quad (7)$$

Также рассчитываются отношения Th/K, Th/U и вычисленный гамма-каротаж. Эти признаки применяются вместе с калиброванными

ми кривыми. Для предварительного выбора конкурирующих литотипов вычисляется нормированное расстояние до узла модели

$$\Delta_j(d) = \left(\frac{1}{|F_d|} \sum_{f \in F_d} w_f \left(\frac{y_f(d) - M_{j,f}}{r_f} \right)^2 \right)^{1/2}, \quad (8)$$

где F_d — множество признаков, для которых на глубине d имеется корректное измеренное значение и задано соответствующее модельное значение, w_f — вес признака, r_f — диапазон параметра f по модели. На основе расстояний выбирается макрогруппа: силикатно-глинистая, карбонатная, эвапоритовая или смешанная.

Оценка долей литотипов выполняется как регуляризованная задача неотрицательной смесевой аппроксимации:

$$v^*(d) = \arg \min_{v \geq 0, \mathbf{1}^\top v = 1} \|W(Mv - y(d))\|_2^2 + \lambda \|v - v(d-1)\|_2^2. \quad (9)$$

Здесь $y(d)$ — вектор наблюдаемых калиброванных признаков, M — матрица модельных узлов, W — диагональная матрица весов, а второй член штрафует резкие изменения долей по глубине. Задача (9) решается проективным градиентным спуском с проекцией на симплекс (1). После этого выполняется сглаживание профилей и нормировка долей до 100%.

3. Пример работы и результаты апробации

Апробация выполнена на данных работающих нефтяных скважин. Рассчитанный литологический состав был сопоставлен с соответствующим литологическим составом, полученным в рамках применяемой на практике в настоящее время методики. Для количественной оценки использовались средняя корреляция по литотипным профилям и среднеквадратичная ошибка RMSE. В качестве простого базового метода применялось сопоставление с ближайшим модельным узлом без калибровки.

На рис. 1 показан пример кросс-плота после настройки. Узлы литотипов нанесены поверх облака точек. Ветвью здесь называется вытянутая область облака точек, соответствующая постепенному изменению состава пород в пространстве двух признаков. Карбонатная

ветвь связана с повышенным содержанием кальция и литотипами известняка и доломита, а сульфатная ветвь — с сочетанием кальциевого и серного признаков, характерным для ангидрита и гипса. Видно, что калиброванные признаки позволяют интерпретировать эти ветви относительно выбранной модели.

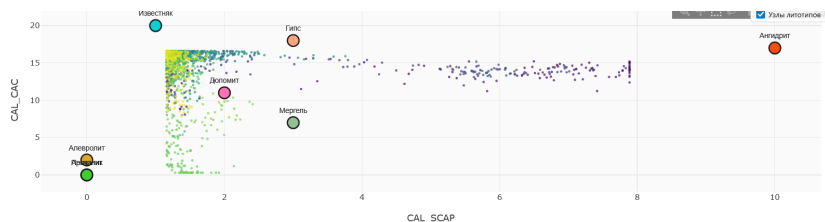


Рис. 1. Пример кросс-плота после настройки признаков.

На рис. 2 показан фрагмент основного рабочего окна системы с набором каротажных кривых по глубине. Пользователь может просматривать исходные и калиброванные признаки, сопоставлять интервалы по глубине и анализировать форму кривых перед расчётом литотипов.

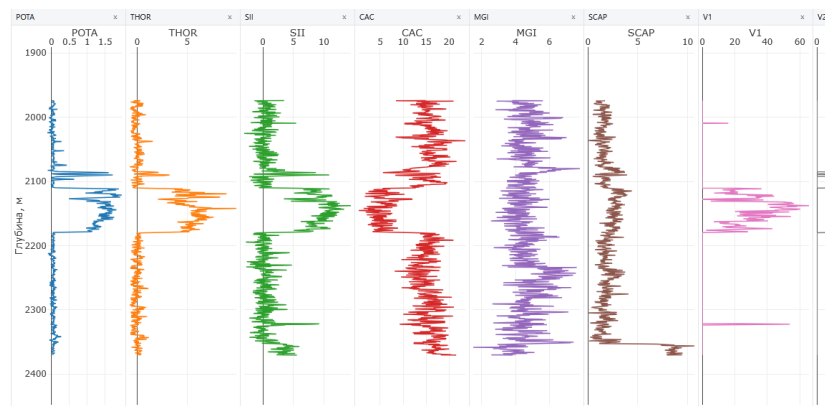


Рис. 2. Фрагмент основного рабочего окна системы с каротажными кривыми.

На рис. 3 приведены отдельные элементы интерфейса, используемые при анализе результата. Инспектор долей литотипов показывает средний вклад каждого литотипа в выбранном интервале. Панель качества калибровки отображает значения R^2 , RMSE и степень аппроксимирующего полинома по каналам. Панель параметров подготовки позволяет управлять нелинейной распаковкой, сглаживанием, автоматической подстройкой узлов и весами признаков в регуляризованной задаче смеси аппроксимации.

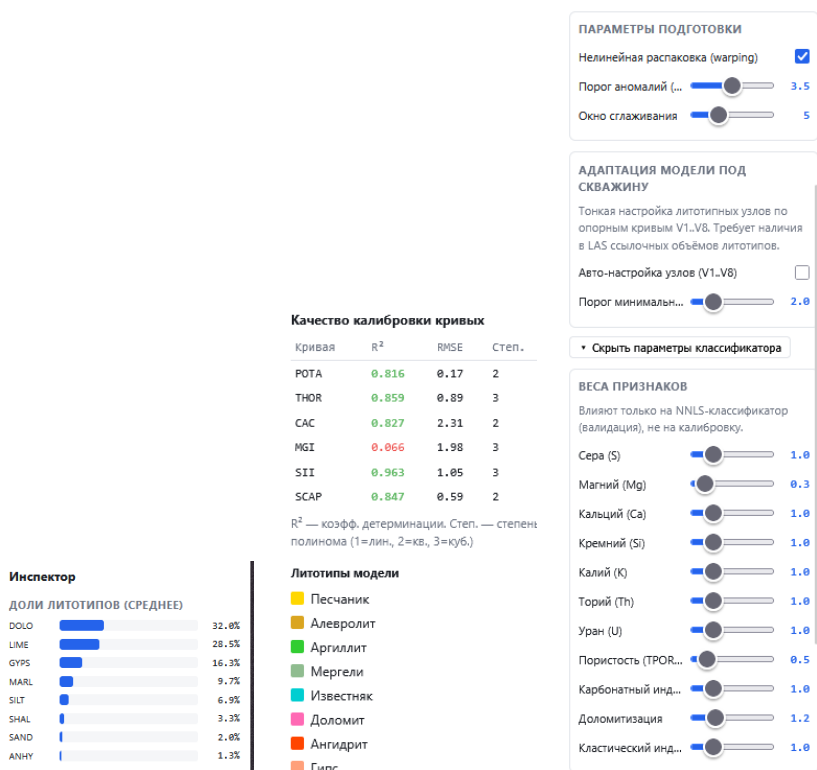


Рис. 3. Отдельные элементы интерфейса системы: инспектор средних долей литотипов, панель качества калибровки кривых и панель параметров подготовки данных.

В табл. 1 приведено сравнение режимов. В опыте без контрольных профилей эти профили не использовались в качестве входных данных и учитывались только после расчёта для оценки качества. Такой режим показывает, что метод способен строить литотипные профили только по исходным каротажным кривым и модели.

Таблица 1. Сравнение режимов интерпретации.

Файл	Базовый метод корр./RMSE	Без контрольных корр./RMSE	С доп. настройкой корр./RMSE
test_1	0,350/25,03	0,524/16,83	0,550/11,25
test_2	0,289/25,62	0,512/18,70	0,641/9,47

Результаты показывают, что базовый режим без использования контрольных профилей существенно превосходит ближайший узел модели. Для файла test_1 средняя корреляция возрастает с 0,350 до 0,524, а RMSE снижается с 25,03 до 16,83. Для файла test_2 корреляция возрастает с 0,289 до 0,512, а RMSE снижается с 25,62 до 18,70. Дополнительная настройка по контрольным профилям снижает RMSE до 11,25 и 9,47 соответственно.

Отдельно сравнивались методы калибровки признаков. Результаты приведены в табл. 2. Для обоих файлов полиномиальная регрессия оказалась предпочтительнее линейной. Квантильное биннирование было полезно как диагностический вариант, однако по средним RMSE уступило обычной полиномиальной регрессии.

Таблица 2. Сравнение методов калибровки признаков.

Файл	Линейная R^2 /RMSE	Полиномиальная R^2 /RMSE	Квантильная R^2 /RMSE
test_1	0,509/1,667	0,543/1,564	0,536/1,597
test_2	0,651/1,380	0,743/1,118	0,734/1,139

Наиболее устойчиво восстанавливались карбонатные и сульфатные интервалы. Слабым местом разработанного алгоритма является сложность определения гипса: его дифференциация от ангидрита по доступным признакам затруднена, так как для уверенного разделения этих литотипов необходим водородный признак, отсутствующий в использованных данных.

Заключение

Разработан метод автоматической настройки данных ГИС под выбранную петрофизическую модель. Метод включает робастную обработку выбросов, модельно-ориентированную калибровку признаков, вычисление производных индексов и регуляризованную оценку долей литотипов. Эксперименты показали, что режим без использования контрольных профилей превосходит простой метод ближайшего узла, а дополнительная настройка по контрольным профилям снижает ошибку интерпретации. Практический результат реализован в виде веб-системы, обеспечивающей загрузку LAS-файлов, визуализацию кросс-плотов, расчёт литотипных профилей и экспорт результата.

Дальнейшее развитие метода связано с расширением выборки скважин, сравнением с дополнительными методами калибровки и уточнением модели для слабо различимых литотипов, прежде всего гипса и ангидрита.

Список литературы

- [1] Литосканерная технология ядерной спектрометрии при исследовании сложнопостроенных геологических объектов / Т. В. Хисметов, О. С. Чернов, М. М. Мухамадеев [и др.] // Нефть. Газ. Новации. — 2024. — № 3 (280). — С. 32–39. — URL: <https://neft-gaz-novacii.ru/ru/archive/131-2024/2333--3-279-2024->.
- [2] Ellis, D. V. Well Logging for Earth Scientists / D. V. Ellis, J. M. Singer. — 2nd ed. — Dordrecht : Springer, 2007. — 708 p. — DOI: <https://doi.org/10.1007/978-1-4020-4602-5>.
- [3] Struyk, C. LAS Version 2.0: A Digital Standard for Logs, Update February 2017 / C. Struyk, D. Greenwood ; Canadian Well Logging Society. — 2017. — 16 p. — URL: https://kcpetrophysics.mynucleus.ca/LAS_2_Documentation_Feb2017.pdf.

Библиографическая ссылка

Майоршин, Р. Н. Автоматическая настройка данных ГИС в рамках петрофизической интерпретационной модели // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 126–135.

Сведения об авторах

МАЙОРШИН РУСЛАН НОВРУЗОВИЧ

Студент магистратуры

Направление «Прикладная математика и информатика»

УДК 621.396.96, 621.391.822

Алгоритмы защиты эхо-сигналов от несинхронных импульсных помех

Мальков А. С.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий М. Ю. Кудряшовым

Аннотация. Исследована задача защиты эхо-сигналов когерентно-импульсных РЛС от несинхронных импульсных помех (НИП). Реализованы и сравнены нелинейные, линейные и адаптивные методы фильтрации; разработаны робастный фильтр Винера с МАД-очисткой и доплеровский алгоритм подавления НИП. Ключевые слова: НИП, эхо-сигнал, робастный фильтр Винера, доплеровский фильтр, адаптивная классификация сигналов.

Введение

Когерентно-импульсные радиолокационные станции (РЛС) широко применяются для обнаружения и сопровождения целей, а также в метеорологических радарх, гидроакустических системах и медицинской ультразвуковой диагностике. Принятый сигнал представляет собой сумму эхо-сигнала, шума и различного рода помех.

Особую проблему составляют несинхронные импульсные помехи (НИП) — посторонние одиночные импульсы, попадающие в пачку зондирующих сигналов случайным образом и не синхронизированные с периодом повторения РЛС. Амплитуда НИП, как правило, значительно превышает амплитуду полезного сигнала, что приводит к искажению доплеровского спектра и ложным обнаружениям.

Целью данной работы являются исследование, реализация и сравнительный анализ алгоритмов защиты эхо-сигналов от НИП с разработкой адаптивной системы выбора оптимального метода фильтрации.

1. Математическая модель

Пусть в пачке из N импульсов принятый дискретный сигнал имеет вид

$$x[n] = s[n] + c[n] + \eta[n], \quad n = 0, \dots, N - 1,$$

где $s[n] = Ae^{j(2\pi f_d n + \varphi_0)}$ — эхо-сигнал с доплеровским сдвигом, $\eta[n]$ — аддитивный гауссовский шум, $c[n]$ — НИП.

Ключевое свойство НИП: помеха поражает один импульс пачки с номером m ,

$$c[n] = \begin{cases} A_{\text{НИП}} e^{j\varphi_{\text{НИП}}}, & n = m, \\ 0, & n \neq m, \end{cases}$$

где $A_{\text{НИП}} \gg A$, а m и $\varphi_{\text{НИП}}$ — случайные величины. ДПФ последовательности $c[n]$ равномерно распределяет энергию помехи по всем доплеровским каналам:

$$|Y_{\text{НИП}}[k]| = \frac{1}{N} \left| \sum_{n=0}^{N-1} c[n] e^{-j2\pi kn/N} \right| = \frac{A_{\text{НИП}}}{N} = \text{const},$$

$$k = 0, 1, \dots, N - 1.$$

2. Реализованные алгоритмы

В работе реализованы три группы методов подавления НИП на языке C++: нелинейные фильтры, линейные фильтры и адаптивные методы.

Нелинейные фильтры

Медианный фильтр заменяет каждый отсчёт медианой значений в скользящем окне шириной w , эффективно подавляя одиночные НИП.

Морфологический фильтр основан на операциях эрозии и дилатации и применяется для подавления выбросов с сохранением формы сигнала.

Фильтр на основе MAD обнаруживает и замещает выбросы, используя медиану абсолютных отклонений (MAD) в качестве робастной оценки разброса, устойчивой к аномальным значениям.

Линейные фильтры

Фильтр Винера минимизирует среднеквадратичную ошибку между выходом фильтра и эталонным сигналом. Оптимальные коэффициенты находятся из уравнения Винера – Хопфа:

$$\mathbf{R} \mathbf{w}_{\text{opt}} = \mathbf{p},$$

где \mathbf{R} — матрица автокорреляции, \mathbf{p} — вектор взаимной корреляции. Присутствие НИП в обучающей выборке приводит к систематическому искажению оценки матрицы \mathbf{R} .

Фильтр Савицкого – Голая осуществляет локальную полиномиальную аппроксимацию методом наименьших квадратов.

Фильтр Калмана является рекуррентным оптимальным фильтром для линейных динамических систем с гауссовским шумом.

Адаптивные методы

Робастный фильтр Винера устраняет чувствительность стандартного фильтра Винера к НИП за счёт трёх модификаций: предварительной очистки выбросов по критерию $|x[n] - \hat{x}| > k \cdot \text{MAD}(x)$ с адаптивным порогом k , робастной оценки входного сигнала, медианным сглаживанием и дополнения нулями на краях сигнала. Порог k выбирается адаптивно в зависимости от оценённого отношения сигнал/шум:

$$k = \begin{cases} 2,5, & \text{ОСШ} < 5 \text{ дБ}, \\ 3,5, & 5 \text{ дБ} \leq \text{ОСШ} \leq 15 \text{ дБ}, \\ 5,0, & \text{ОСШ} > 15 \text{ дБ}. \end{cases}$$

Длина фильтра определяется по формуле:

$$M = \left\lceil \frac{1}{2f_{95}} \right\rceil, \quad M \in [4; 128],$$

где f_{95} — частота, соответствующая 95 % накопленной энергии спектра.

Доплеровский алгоритм подавления НИП использует равномерность спектра одиночной НИП. Обнаружение выполняется по

коэффициенту вариации амплитуд спектральных бинов:

$$CV = \frac{\sigma_{|Y|}}{\mu_{|Y|}}.$$

Индекс поражённого импульса определяется через устойчивую фазовую оценку:

$$\hat{m} = \left[-\frac{N \arg(\Psi)}{2\pi} \right] \bmod N, \quad \Psi = \sum_{k=0}^{N-2} Y[k+1] \overline{Y[k]}.$$

Вклад помехи вычитается из каждого спектрального бина, после чего сигнал восстанавливается обратным ДПФ.

3. Адаптивная система классификации

Разработана система, автоматически выбирающая подходящий фильтр по входному сигналу $x[n]$. Вычисляются статистические признаки: куртозис κ , пик-фактор, частота пересечений нуля, разреженность, гладкость и нормированный размах. На основе правил классификации сигнал относится к одному из шести классов согласно табл. 1.

Таблица 1. Классы сигналов и назначаемые фильтры.

Класс	Критерии	Фильтр
Защумлённый	$\kappa > 10$; $RR > 3$	Робастный Винера
Эхо-сигнал	$s > 0,65$; $CF > 3$	Фильтр Калмана
Прямоуг.	$\kappa < 2,5$; $\sigma_s > 0,05$	Савицкого – Голая(11, 3)
Треугольный	$\kappa < 3,0$; $ZCR > 0,05$	Медианный(5)
Синусоида	$\kappa < 4,0$; $ZCR > 0,03$	Винера(16)
Неизвестный	иначе	Медианный(7)

4. Результаты вычислительных экспериментов

Алгоритмы протестированы на синтетических данных по четырём метрикам: отношению сигнал/шум (ОСШ, дБ), медиане абсо-

лютных отклонений (MAD), среднеквадратичной ошибке (СКО) и коэффициенту корреляции ρ . Результаты приведены в табл. 2.

Таблица 2. Сравнение алгоритмов по метрикам качества.

Алгоритм	ОСШ, дБ	MAD	СКО	ρ
Медианный(5)	$18,2 \pm 2,1$	0,48	$2,3 \cdot 10^{-4}$	0,891
Медианный(7)	$19,8 \pm 2,3$	0,44	$1,9 \cdot 10^{-4}$	0,905
Фильтр Винера(8,5)	$17,9 \pm 2,8$	0,51	$2,6 \cdot 10^{-4}$	0,878
Роб. Винера	$24,1 \pm 1,8$	0,37	$1,4 \cdot 10^{-4}$	0,941
Морфологический(5)	$16,1 \pm 2,6$	0,62	$3,8 \cdot 10^{-4}$	0,841
Детект. выбор. MAD	$21,4 \pm 1,9$	0,40	$1,6 \cdot 10^{-4}$	0,924
Савицкого-Голая(11,3)	$18,5 \pm 2,4$	0,46	$2,1 \cdot 10^{-4}$	0,894
Фильтр Калмана	$17,2 \pm 3,1$	0,53	$2,8 \cdot 10^{-4}$	0,862
Доплер. НИП	$24,7 \pm 1,2$	0,33	$1,1 \cdot 10^{-4}$	0,963
Адаптивный	$23,8 \pm 1,5$	0,36	$1,3 \cdot 10^{-4}$	0,951

Доплеровский фильтр показал наилучшие результаты по всем метрикам: ОСШ 24,7 дБ и корреляция 0,963. Робастный фильтр Винера обеспечивает наилучший баланс качества и универсальности: ОСШ 24,1 дБ против 17,9 дБ у стандартного фильтра Винера.

В табл. 3 показана зависимость ОСШ от нормированной амплитуды НИП при фиксированном порядке фильтра. При малых амплитудах оба фильтра близки, однако при стократном превышении выигрыш робастного варианта достигает 20,7 дБ.

Таблица 3. Зависимость ОСШ от амплитуды НИП для фильтров Винера и робастного Винера.

$A_{\text{НИП}}/\sigma_s$	ОСШ _{Винер} , дБ	ОСШ _{Роб.} , дБ	Δ ОСШ, дБ
1	37,3	38,9	+1,6
5	36,7	38,9	+2,2
10	35,3	38,9	+3,6
20	32,3	38,9	+6,6
50	12,4	27,3	+14,9
100	6,6	27,3	+20,7
200	1,1	27,1	+26,0

Заключение

В ходе работы реализованы и сравнены десять алгоритмов защиты эхо-сигналов от несинхронных импульсных помех. Разработаны два оригинальных алгоритма: робастный фильтр Винера с адаптивной MAD-очисткой и доплеровский фильтр НИП с фазовой оценкой индекса поражённого импульса. Создана адаптивная система классификации сигналов, автоматически выбирающая оптимальный метод фильтрации по шести статистическим признакам.

Экспериментально подтверждено, что доплеровский фильтр НИП достигает наилучших абсолютных показателей (ОСШ 24,7 дБ, $\rho = 0,963$), а робастный фильтр Винера обеспечивает лучшее соотношение качества и универсальности применения. Адаптивная система обеспечивает близкое к оптимальному качество без участия оператора, что делает её перспективной для практического применения.

Список литературы

- [1] Левин, Б. Р. Теоретические основы статистической радиотехники. — 3-е изд. — М. : Радио и связь, 1989. — 656 с. — URL: <https://urss.ru/cgi-bin/db.pl?lang=Ru&blang=ru&page=Book&id=61080>.
- [2] Оппенгейм, А. Цифровая обработка сигналов / пер. с англ. / А. Оппенгейм, Р. Шафер. — М. : Техносфера, 2006. — 856 с. — URL: <https://www.technosfera.ru/lib/book/317>.
- [3] Тихонов, В. И. Статистический анализ и синтез радиотехнических устройств и систем / В. И. Тихонов, В. Н. Харисов. — М. : Радио и связь, 1991. — 608 с. — URL: <https://djvu.online/file/dFuvVma6QB40E>.

Библиографическая ссылка

Мальков, А. С. Алгоритмы защиты эхо-сигналов от несинхронных импульсных помех // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 136–141.

Сведения об авторах

МАЛЬКОВ АРТЁМ СЕРГЕЕВИЧ

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 004.62, 004.8

Разработка рекомендательной системы для совершения покупок на основе анализа пользовательского поведения с использованием методов интеллектуального анализа данных

Москвичёв Я. А.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий Е. Н. Гришиной

Аннотация. Рассмотрена задача построения рекомендательной системы для интернет-магазина на основе методов поиска ассоциативных правил. Предложен комбинированный подход, объединяющий кластеризацию транзакций методом K-Means++ с построением FP-дерева на каждом кластере. Сравнение четырёх алгоритмов на датасете Instacart показало, что кластеризованный FP-Growth обеспечивает наивысшее качество правил с комплексной оценкой 95,5 %.

КЛЮЧЕВЫЕ СЛОВА: рекомендательная система, ассоциативные правила, FP-Growth, кластеризация, K-Means++.

Введение

Рост ассортимента товаров в интернет-магазинах приводит к информационной перегрузке, при которой пользователь не может самостоятельно выбрать подходящий товар. Рекомендательные системы решают эту задачу, формируя персонализированные предложения на основе анализа данных о покупках [1]. Среди методов построения рекомендательных систем алгоритмы поиска ассоциативных правил — Apriori и FP-Growth — позволяют выявлять устойчивые паттерны совместных покупок без необходимости в явных пользовательских оценках [2]. Однако классические алгоритмы имеют

ограничения при работе с гетерогенными данными, содержащими группы пользователей с различающимися паттернами поведения.

Целью настоящей работы является разработка рекомендательной системы для интернет-магазина, объединяющей предварительную кластеризацию транзакций методом K-Means++ с алгоритмами поиска ассоциативных правил, а также сравнительный анализ четырёх вариантов алгоритмов для выбора оптимального.

1. Математическая постановка задачи

Пусть $I = \{i_1, \dots, i_n\}$ — множество товаров, $D = \{T_1, \dots, T_m\}$ — множество транзакций, где каждая транзакция $T_k \subseteq I$ представляет собой набор товаров, приобретённых в рамках одного заказа. Ассоциативное правило имеет вид $X \rightarrow Y$, где $X \subset I$, $Y \subset I$, $X \cap Y = \emptyset$.

Качество правил оценивается двумя метриками. Поддержка определяется как доля транзакций, содержащих $X \cup Y$:

$$\text{support}(X \rightarrow Y) = \frac{|\{T \in D : X \cup Y \subseteq T\}|}{|D|}. \quad (1)$$

Достоверность определяется как условная вероятность наличия Y при наличии X :

$$\text{confidence}(X \rightarrow Y) = \frac{\text{support}(X \cup Y)}{\text{support}(X)}. \quad (2)$$

Правило считается значимым, если его поддержка (1) и достоверность (2) превышают заданные пороговые значения.

Задача рекомендации формулируется следующим образом: для заданного набора товаров $S \subseteq I$ (содержимого корзины покупок) определить множество товаров $R \subseteq I \setminus S$, связанных с S ассоциативными правилами с высокой достоверностью.

2. Алгоритмы поиска ассоциативных правил

Алгоритм Apriori выполняет поуровневый перебор наборов-кандидатов, используя свойство антимонотонности: если набор элементов нечаст (его поддержка ниже порога), то все его надмножества также нечасты. На каждой k -й итерации из $(k-1)$ -элементных частей

наборов L_{k-1} генерируются k -элементные кандидаты C_k , затем подсчитывается их поддержка. Основной недостаток — необходимость многократного сканирования базы транзакций [2].

Алгоритм FP-Growth [3] устраняет этап генерации кандидатов за счёт построения компактного FP-дерева. На первом проходе определяются частоты товаров, на втором — транзакции вставляются в дерево, где общие префиксы объединяются. Затем для каждого элемента строится условная база паттернов и условное FP-дерево, из которого рекурсивно извлекаются частые наборы. FP-Growth требует только два прохода по данным и демонстрирует существенно более высокую производительность.

3. Комбинирование кластеризации с поиском правил

Для повышения качества ассоциативных правил предложен комбинированный подход, объединяющий кластеризацию транзакций с последующим поиском правил внутри каждого кластера. Каждая транзакция представляется бинарным вектором размерности 100 (по наиболее частым товарам). Кластеризация выполняется методом K-Means с инициализацией K-Means++, минимизирующей функционал

$$\text{WCSS} = \sum_{k=1}^K \sum_{x \in C_k} \|x - \mu_k\|^2, \quad (3)$$

где C_k — множество объектов кластера k , μ_k — его центроид.

Оптимальное число кластеров определяется методом локтя: для различных значений K вычисляется функционал (3), строится прямая от первой до последней точки WCSS-кривой, и выбирается значение K , для которого перпендикулярное расстояние от точки кривой до прямой максимально. Экспериментально установлено, что $K = 4$ обеспечивает оптимальный баланс между качеством сегментации и вычислительными затратами.

Работа комбинированного алгоритма Clustered FP-Growth, схема которого представлена на рис. 1, состоит из четырёх этапов: векторизация транзакций, кластеризация методом K-Means++, построение FP-дерева и поиск частых наборов внутри каждого кластера, объединение результатов. Поскольку поиск правил выполняется на

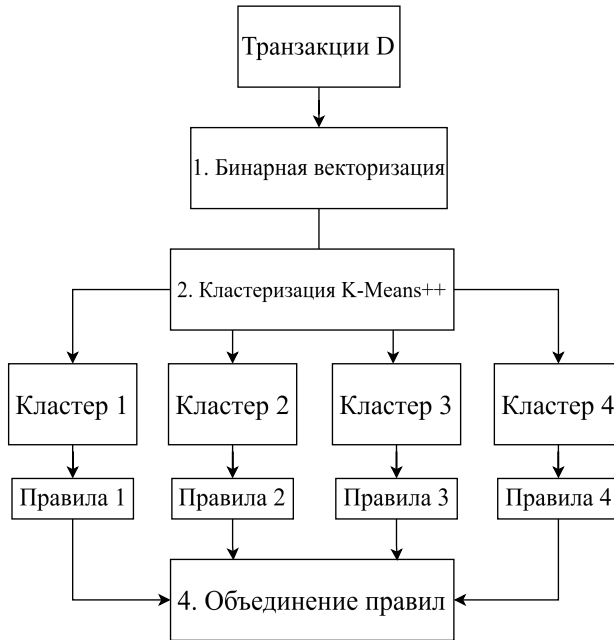


Рис. 1. Схема работы комбинированного алгоритма Clustered FP-Growth.

подвыборках, пороговое значение поддержки рассчитывается относительно размера кластера, что позволяет выявлять локальные паттерны с более низкой абсолютной частотой.

4. Архитектура и программная реализация

Система реализована на языке Java 21 с использованием фреймворка Spring Boot 3.2.0 и СУБД PostgreSQL. Архитектура построена на принципах многослойного разделения ответственности с паттерном «Интерфейс + Реализация»: слой представления (Thymeleaf, Bootstrap 5.3), слой контроллеров, слой сервисов с алгоритмическими модулями, слой репозитория (Spring Data JPA) и слой базы данных.

Таблица 1. Сравнение алгоритмов поиска ассоциативных правил.

Алгоритм	Правил	Средн. подд.	Средн. дост.	Уник. товаров	Время (сек)
Cl. FP-Growth	20020	0,53 %	25,01 %	301	0,75
Cl. Apriori	20020	0,53 %	25,01 %	301	3,30
FP-Growth	54	0,41 %	17,54 %	10	0,62
Apriori	54	0,41 %	17,54 %	10	2,33

Рекомендательная подсистема формирует рекомендации в двух контекстах. На карточке товара выполняется двухфазный поиск: прямой и обратный по ассоциативным правилам. В корзине покупок реализован трёхфазный поиск с последовательным ослаблением критериев: точное подмножество, пересечение, обратный поиск по консеквенту. Результаты кэшируются с TTL 30 минут.

5. Экспериментальное исследование

Эксперименты проведены на датасете Instacart Market Basket Analysis, содержащем 49 688 товаров и 1 384 617 записей о покупках. Базовые параметры: $\text{minSupport} = 0,003$, $\text{minConfidence} = 0,005$, $K = 4$.

Для комплексной оценки алгоритмов используется формула

$$S = 0,8 \times A + 0,2 \times V, \quad (4)$$

где компонента качества $A = 0,7 \times (\bar{c}/c_{\max}) + 0,3 \times (\bar{s}/s_{\max})$ учитывает среднюю достоверность \bar{c} и среднюю поддержку \bar{s} , а компонента скорости $V = 1 - t/t_{\max}$ учитывает время выполнения t .

Результаты сравнительного анализа четырёх алгоритмов представлены в табл. 1.

Анализ табл. 1 показал, что алгоритмы Apriori и FP-Growth при одинаковых параметрах генерируют идентичные наборы правил, различаясь только производительностью. Кластеризованные варианты демонстрируют более высокую среднюю достоверность правил за счёт поиска на однородных подвыборках, а также значительно более широкое покрытие каталога: 301 уникальный товар в правилах против 10 у некластеризованных алгоритмов.

Кластеризованный FP-Growth показал наилучшие результаты по формуле (4) с комплексной оценкой $S = 95,5\%$: при максимальных значениях средней достоверности (25,01 %) и средней поддержки (0,53 %) время выполнения составило 0,75 сек против 3,30 сек у кластеризованного Apriori. Кластеризованный Apriori при идентичном качестве правил получил оценку 80,0 %, что обусловлено исключительно различием в производительности.

Заключение

В ходе работы разработана рекомендательная система для интернет-магазина, объединяющая методы кластеризации и поиска ассоциативных правил. Реализованы и сравнены четыре алгоритма: Apriori, FP-Growth, кластеризованный Apriori и кластеризованный FP-Growth. Экспериментальное исследование на реальном датасете подтвердило, что предварительная кластеризация транзакций методом K-Means++ повышает среднюю достоверность ассоциативных правил и расширяет покрытие каталога товаров. Кластеризованный FP-Growth выбран в качестве оптимального алгоритма для продуктивной подсистемы.

Перспективными направлениями являются внедрение гибридных моделей, объединяющих ассоциативные правила с коллаборативной фильтрацией, а также разработка механизмов А/В-тестирования для оценки эффективности рекомендаций.

Список литературы

- [1] Ricci, F. Recommender Systems Handbook / F. Ricci, L. Rokach, B. Shapira. — New York : Springer, 2022. — 1060 p. — DOI: <https://doi.org/10.1007/978-1-0716-2197-4>.
- [2] Agrawal, R. Fast Algorithms for Mining Association Rules / R. Agrawal, R. Srikant // Proceedings of the 20th International Conference on Very Large Data Bases (VLDB). — San Francisco, CA : Morgan Kaufmann Publishers Inc., 1994. — P. 487–499. — URL: <https://dl.acm.org/doi/10.5555/645920.672836>.
- [3] Han, J. Mining Frequent Patterns without Candidate Generation / J. Han, J. Pei, Y. Yin // Proceedings of the 2000 ACM SIGMOD

International Conference on Management of Data. — New York : Association for Computing Machinery, 2000. — P. 1–12. — DOI: <https://doi.org/10.1145/342009.335372>.

Библиографическая ссылка

Москвичёв, Я. А. Разработка рекомендательной системы для совершения покупок на основе анализа пользовательского поведения с использованием методов интеллектуального анализа данных // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 143–149.

Сведения об авторах

Москвичёв Яков Александрович

Студент магистратуры

Направление «Прикладная информатика»

УДК 004.9, 658.15

Разработка инструментария для анализа и оптимизации структуры оборотных активов предприятия на основе теории временных предпочтений

Ознобихин Д. В.

Тверской государственной университет

Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа И. А. Багровой

Аннотация. Предложен программный прототип инструмента, формирующий отчёт по структуре оборотных активов с выявлением отклонений на основе данных предприятия. Для управления дебиторской задолженностью предложен сценарный анализ условной оплаты и расчёт пороговой скидки d^* . Проведена апробация на данных промышленного предприятия.

КЛЮЧЕВЫЕ СЛОВА: оборотные активы, дебиторская задолженность, оборачиваемость, временная стоимость денег, приведённая стоимость.

Введение

Оборотные активы (далее — ОА) характеризуют текущую ликвидность предприятия и скорость превращения ресурсов в денежные средства. На практике финансовому менеджеру важно не только рассчитывать показатели, но и выявлять отклонения, которые могут приводить к замораживанию средств в оборотных активах, росту дебиторской задолженности (далее — ДЗ) и увеличению операционного цикла. Также ряд параметров, такие как даты расчёта с поставщиками и покупателями, размер предоставляемых скидок, могут быть неопределёнными.

Цель работы — разработка и программная реализация прототипа инструментария анализа структуры оборотных активов с учётом теории временных предпочтений. Инструментарий включает в себя

структурный, динамический и коэффициентный анализ, расчёт валовой прибыли и валовой рентабельности, на основе которых происходит формирование индикаторов критических зон и рекомендаций, а также метод, который через сценарный анализ приведённых сумм дебиторской задолженности позволяет обосновать решение предоставления скидки дебиторам. Прототип апробирован на данных промышленного предприятия.

1. Входные данные и система показателей

Имеются данные бухгалтерской (финансовой) отчётности предприятия за несколько периодов. Требуется проанализировать текущее состояние и структуру оборотных активов, выявить критические участки и сформулировать рекомендации по улучшению и оптимизации структуры оборотных активов. Под критическими участками или слабыми местами в структуре оборотных активов здесь и далее подразумеваются активы, которые ухудшают скорость оборота средств, ликвидность предприятия и не приносят положительной финансовой отдачи. Для этого рассчитываются экономические показатели структуры, динамики, рентабельности и оборачиваемости, а также индикаторы отклонений. Дополнительно рассматривается задача выбора условий оплаты по дебиторской задолженности: сравниваются сценарии «позже без скидки» и «раньше со скидкой» через приведённую стоимость.

Для работы инструмента необходимы данные предприятия, сгруппированные в пять таблиц: «Дебиторская задолженность», «Структура оборотных активов», «Финансовые результаты», «Параметры сценария», «Дебиторы». Первые три листа содержат экономические данные предприятия по периодам и используются для расчёта показателей анализа оборотных активов, оборачиваемости. Четвёртый лист используется для оценки целесообразности предоставления заданной скидки дебитору. В пятом листе задаётся список контрагентов (сумма долга, сроки оплаты и ставка дисконтирования) для расчёта максимальной выгодной для предприятия (пороговой) скидки d^* по каждому дебитору.

Анализ текущего состояния структуры оборотных активов состоит из трёх основных этапов:

- по данным отчётности за периоды t рассчитываются базовые показатели структуры и эффективности использования оборотных активов (доли элементов в структуре ОА, динамика ключевых показателей по периодам, валовая прибыль и валовая рентабельность, показатели оборачиваемости);
- рассчитанные показатели сопоставляются с предыдущим периодом и на их основе формируются индикаторы отклонений и рекомендации;
- дополнительно выполняется сценарный анализ условий оплаты по дебиторской задолженности.

Далее приведены формулы расчёта базовых показателей с помощью которых в дальнейшем происходит поиск слабых активов [2, 3].

Структурный анализ выполняется через долю отдельного элемента оборотных активов в общей сумме оборотных активов:

$$w_{i,t} = \frac{OA_{i,t}}{OA_{1200,t}}, \quad (1)$$

где $OA_{i,t}$ — значение статьи оборотных активов с кодом i в период t , $OA_{1200,t}$ — итог по оборотным активам (строка 1 200 бухгалтерского баланса).

Динамический анализ основан на изменении и темпе роста показателя:

$$\Delta X_t = X_t - X_{t-1}, \quad T_t = \frac{X_t}{X_{t-1}}, \quad (2)$$

где X_t — анализируемый показатель в период t .

Расчёт валовой прибыли и валовой рентабельности:

$$GP_t = R_t - COGS_t, \quad GM_t = \frac{GP_t}{R_t}, \quad (3)$$

где R_t — выручка за период t (строка 2 110 бухгалтерского баланса), $COGS_t$ — себестоимость продаж (строка 2 120 бухгалтерского баланса).

Коэффициентный анализ включает показатели оборачиваемости оборотных активов и дебиторской задолженности:

$$K_t^{OA} = \frac{R_t}{OA_t}, \quad P_t^{OA} = \frac{D}{K_t^{OA}}, \quad (4)$$

$$K_t^{AR} = \frac{R_t}{AR_t}, \quad P_t^{AR} = \frac{D}{K_t^{AR}}. \quad (5)$$

где \overline{OA}_t и \overline{AR}_t — средние за период значения оборотных активов и дебиторской задолженности, D — число дней в периоде.

2. Индикаторы отклонений

На основе показателей из раздела 1 формируются индикаторы отклонений. Индикатор срабатывает (отображается в программной реализации), если фиксируется выполнение неравенства, связанного с этим индикатором (далее — условие срабатывания). Были выбраны индикаторы только основных показателей, наиболее влияющих на оборотные активы предприятия [1, 2]. Если базовое значение для расчёта отсутствует или равно нулю (и возникает деление на ноль), индикатор для данного периода не применяется.

Для описания условий срабатывания индикаторов введем следующие обозначения:

- t — отчётный период;
- INV_t — запасы (строка 1 210) в период t ;
- AR_t — общая дебиторская задолженность в период t ;
- R_t — выручка (строка 2 110) в период t ;
- T_{X_t} — темп роста показателя X ;
- DSO — средний срок погашения дебиторской задолженности (Days Sales Outstanding), дни;
- w_t^{over} — доля просроченной дебиторской задолженности;
- P_t^{OA} — период оборота оборотных активов, дни;
- GM_t — валовая рентабельность.

Далее опишем условия срабатывания и расшифровку индикаторов, используемых в работе.

РОСТ_ЗАПАСОВ — индикатор, показывающий что темп роста запасов выше темпа роста выручки. Срабатывает при выполнении

условия $T_{INV_t} > T_{R_t}$. Свидетельствует о накоплении избыточных или медленно реализуемых запасов (сырья, готовой продукции, незавершённой продукции и т. д.).

РОСТ_DSO_БОЛЬШЕ_ВЫРУЧКИ показывает, что темп роста дебиторской задолженности выше темпа роста выручки. Срабатывает при выполнении условия $T_{AR_t} > T_{R_t}$. Указывает на рост доли средств, отвлечённых в расчёты с дебиторами.

РОСТ_DSO — увеличился средний срок погашения дебиторской задолженности. Срабатывает при выполнении условия $DSO_t > DSO_{t-1}$.

РОСТ_ПРОСРОЧЕННОЙ_ДЗ — увеличилась доля просроченной дебиторской задолженности. Срабатывает при выполнении условия $w_t^{over} > w_{t-1}^{over}$.

РОСТ_ОБОРОТА_ОА — увеличился период оборота оборотных активов. Срабатывает при выполнении условия $P_t^{OA} > P_{t-1}^{OA}$. Индикатор может рассматриваться как сигнал замедления оборота средств и потенциального роста потребности в финансировании.

СНИЖЕНИЕ_РЕНТАБЕЛЬНОСТИ — снизилась валовая рентабельность. Срабатывает при выполнении условия $GM_t < GM_{t-1}$. Является сигналом ухудшения соотношения выручки и себестоимости.

Далее приведены формулы для расчёта показателей, используемых в правилах.

Темп роста показателя X рассчитывается по формуле

$$T_{X,t} = \frac{X_t}{X_{t-1}}. \quad (6)$$

Если срок погашения ДЗ задан во входных данных, он принимается без перерасчёта, иначе вычисляется как

$$DSO_t = \frac{\overline{AR}_t}{R_t} \cdot D, \quad \overline{AR}_t = \frac{AR_t + AR_{t-1}}{2}, \quad (7)$$

где \overline{AR}_t — средняя дебиторская задолженность за период t , AR_t — дебиторская задолженность на конец периода, D — число дней в периоде ($D = 365$).

Доля просроченной дебиторской задолженности:

$$w_t^{over} = \frac{AR_t^{over}}{AR_t}, \quad (8)$$

где AR_t^{over} — просроченная дебиторская задолженность, AR_t — общая дебиторская задолженность.

3. Управление дебиторской задолженностью с помощью сценарного анализа в теории временных предпочтений

Кроме расчёта финансовых показателей и индикаторов отклонений в работе также представлен один из возможных способов управления дебиторской задолженностью, которая является одним из ключевых элементов в структуре оборотных активов.

Предположим, что в отчётном периоде t предприятие имеет дебиторскую задолженность в размере S_t . Предприятию может быть выгодно предоставить дебитору скидку и получить оплату раньше, чем ожидать полную оплату в базовый срок. Рассматриваются две альтернативы:

- 1) получить полную оплату задолженности через t_1 дней;
- 2) предоставить скидку d_t и получить оплату раньше — через t_0 дней.

Для обоснования выбора альтернативы рассчитываются приведённые стоимости выплат задолженности (PV):

$$PV_{1,t} = \frac{S_t}{(1 + r_t)^{t_1/D}}, \quad (9)$$

$$PV_{2,t} = \frac{S_t(1 - d_t)}{(1 + r_t)^{t_0/D}}. \quad (10)$$

Критерий сравнения задаётся разницей приведённых стоимостей:

$$E_t = PV_{2,t} - PV_{1,t}, \quad (11)$$

где

- S_t — сумма задолженности, руб.;
- t_1 — базовый срок оплаты без скидки, дни;
- t_0 — срок досрочной оплаты со скидкой, дни;

- d_t — скидка, доли;
- r_t — ставка дисконтирования, доли;
- D — число дней в периоде ($D = 365$);
- $PV_{1,t}$ — приведённая стоимость оплаты без скидки;
- $PV_{2,t}$ — приведённая стоимость оплаты со скидкой;
- E_t — эффект от предоставления скидки по критерию PV.

Если $E_t > 0$, то предоставление скидки экономически оправдано; если $E_t < 0$, то скидка экономически невыгодна и альтернатива 2 отбрасывается.

При наличии списка дебиторов, для каждого рассчитывается пороговая скидка d_t^* - максимальная скидка, при которой досрочная оплата не менее выгодна, чем базовая оплата без скидки. Она находится из равенства $PV_{1,t} = PV_{2,t}$:

$$d_t^* = 1 - (1 + r_t)^{(t_0 - t_1)/D}. \quad (12)$$

В программной реализации расчёты по формулам (9)–(12) применяются как для сценарного анализа по периодам, так и для списка дебиторов, где для каждого рассчитывается индивидуальное значение d^* при заданных S , t_1 , t_0 и r .

4. Программная реализация

Прототип инструментария реализован в виде веб-сервиса, где серверная часть (бекэнд) написана на языке Python с использованием фреймворка FastAPI, а клиентская часть (фронтэнд) на Angular. На вход программы в блок загрузки файлов сайта подаётся один файл формата .xlsx с пятью листами с данными в определенном формате.

В процессе обработки выполняются чтение и проверка форматов данных, нормализация параметров сценария (ставка r и скидка d переводятся из процентов в доли), расчёт финансовых показателей, применение правил индикаторов отклонений и формирование рекомендаций.

Результат работы сервиса возвращается в формате JSON и ответном .xlsx файле, загружаемом автоматически. Файл содержит листы с рассчитанными финансовыми показателями, результатами PV-анализа, перечнем сработавших индикаторов, рекомендациями и предупреждениями по входным данным.

5. Апробация

Апробация инструментария выполнена на данных промышленного производственного предприятия за 2022–2024 гг., полученных из открытого источника, и включает диагностику по индикаторам отклонений и два сценарных эксперимента с модулем временных предпочтений.

Диагностика по индикаторам показала наличие отклонений в 2023 и 2024 гг. Перечень сработавших индикаторов приведён в табл. 1. В 2023 г. зафиксированы сигналы ускоренного роста запасов и дебиторской задолженности относительно выручки, а также ухудшение платёжной дисциплины (рост срока погашения и доли просроченной задолженности). В 2024 г. указанные выше сигналы сохраняются и дополнительно появляются замедление периода оборота ОА и снижение валовой рентабельности.

Таблица 1. Перечень сработавших индикаторов.

Периоды	Интерпретация
2023, 2024	Запасы растут быстрее выручки. Возможна заморозка средств в запасах.
2023, 2024	ДЗ растёт быстрее выручки. Рост неиспользуемых средств.
2023, 2024	Срок погашения ДЗ увеличивается. Замедление поступления денег.
2023, 2024	Растёт доля просроченной ДЗ. Ухудшение качества дебиторской задолженности.
2024	Период оборота ОА увеличивается. Замедление оборота средств.
2024	Валовая рентабельность снижается относительно прошлого периода.

Далее для демонстрации работы сценарного анализа приведены результаты проверки целесообразности скидки в размере $d = 1\%$ для

данных предприятия за 2022–2024 гг. В качестве ставки дисконтирования r_t принята средняя ключевая ставка ЦБ за соответствующий год. В качестве t_1 был взят средний срок погашения ДЗ из данных предприятия, t_0 задан как сценарный параметр, но не слишком маленький для приближения к реальным условиям. Исходные параметры по периодам приведены в табл. 2, результаты расчёта PV_1 , PV_2 , E_t и пороговой скидки d_t^* — в табл. 3.

Таблица 2. Параметры сценарного анализа по периодам.

Период	S , тыс. руб.	t_1 , дни	t_0 , дни	d , %	r , %
2022	64 435,00	70,9	60,0	1,00	10,50
2023	75 087,00	85,3	60,0	1,00	10,00
2024	92 654,00	96,5	60,0	1,00	17,50

Таблица 3. Результаты сценарного анализа по периодам.

Период	PV_1 , тыс. руб.	PV_2 , тыс. руб.	E , тыс. руб.	d^* , %	Итог
2022	63 197,35	62 752,20	–445,15	0,30	невыгодно
2023	73 433,01	73 180,55	–252,46	0,66	невыгодно
2024	88 786,58	89 327,73	541,15	1,60	выгодно

По итогам расчётов скидка в 1 % экономически оправдана только для 2024 г., что связано с более высокой ставкой дисконтирования и, как следствие, большей ценности раннего поступления денежных средств. Также из этого примера можно сделать вывод, что ставка дисконтирования r_t отражает «ценность» более раннего поступления денег для предприятия. Здесь, в качестве r_t использована средняя ключевая ставка ЦБ РФ за год, что является нижней границей «стоимости» денег. Однако при некоторых условиях, например при неотложных кредиторских задолженностях, предприятие может увеличить r_t , что будет обозначать более сильную заинтересованность в быстрых выплатах.

В качестве второго эксперимента над блоком обоснования скидки также проведён расчёт пороговой скидки d^* для отдельных дебиторов. Для набора контрагентов заданы S , t_1 , t_0 и r , после чего вычислено d^* . Результаты приведены в табл. 4. В исходных данных

первые три дебитора (Д1–Д3) составляют 60 % от общей дебиторской задолженности $S = 88\,786$ тыс. руб. предприятия за 2024 г. (см. табл. 2) и распределяют её в долях 10 %, 20 % и 30 % соответственно. Такое распределение отражает предположение, что существенная часть долга может приходиться на относительно небольшое число должников. Дебитор Д4 рассматривается как эталонный, а для остальных (Д5–Д9) меняется только один параметр относительно Д4. Параметр $r = 15\%$ — текущая ставка ЦБ РФ на момент проведения эксперимента.

Таблица 4. Расчёт пороговой скидки d^* для набора дебиторов.

Дебитор	S , тыс. руб.	t_1 , дни	t_0 , дни	r , %	d^* , %	Скидка, тыс. руб.
Д1	11 118,00	96,5	60,0	17,50	1,60	177,85
Д2	16 677,00	96,5	60,0	17,50	1,60	266,78
Д3	27 796,00	96,5	60,0	17,50	1,60	444,65
Д4	100,00	60,0	30,0	15,00	1,14	1,1422
Д5	100,00	60,0	15,0	15,00	1,71	1,7083
Д6	100,00	60,0	45,0	15,00	0,57	0,5727
Д7	100,00	75,0	30,0	15,00	1,71	1,7083
Д8	100,00	60,0	30,0	20,00	1,49	1,4874
Д9	100,00	60,0	30,0	50,00	3,28	3,2777

В данном примере подтверждается зависимость пороговой скидки от заданных параметров сценария. Пороговая скидка d^* возрастает при увеличении ставки дисконтирования r и при увеличении периода между t_0 и t_1 , поскольку более раннее поступление средств становится относительно ценнее. Для дебиторов Д7 и Д5 порог d^* наибольший именно из-за увеличенного периода $[t_0; t_1]$. Дебитору Д6 ожидаемо предлагается меньшая скидка из-за большего срока ранней оплаты t_0 . Дебиторы Д8 и Д9 позволяют пронаблюдать влияние ставки дисконтирования: в данных условиях рост r на 1 % повышает пороговую скидку d^* примерно на 0,07 %.

Заключение

В рамках работы был разработан и реализован прототип инструментария для анализа структуры оборотных активов. Инструментарий рассчитывает показатели структуры, динамики, валовой

рентабельности и оборачиваемости, а также формирует индикаторы отклонений. Дополнительно реализован сценарный модуль на основе теории временных предпочтений для сравнения условий оплаты через приведённую стоимость и пороговой скидки d_t^* . Проведена апробация инструментария на открытых данных производственного предприятия. После доработки (добавления шифрования при передаче данных предприятия, системы аутентификации, усовершенствования способов ввода/вывода данных) инструментарий можно использовать на реальных предприятиях для анализа структуры ОА и управления дебиторской задолженностью.

Список литературы

- [1] Дранко, О. И. Методология управления развитием промышленных предприятий с использованием комплекса математических моделей и методов прогнозирования : автореф. дис. . . . д-ра техн. наук : 05.13.10 / О. И. Дранко. — Челябинск, 2018. — 43 с. — URL: https://www.susu.ru/sites/default/files/dissertation/avtoref_dranko.pdf.
- [2] Дранко, О. И. Финансовый менеджмент: Технологии управления финансами предприятия : учеб. пособие для вузов / О. И. Дранко. — Москва : ЮНИТИ-ДАНА, 2004. — 351 с. — URL: <https://api.libraryksu.kg/elibrary/books/Dranko%20Financialsovy%20menedjment8142.pdf>.
- [3] Талер, Р. Новая поведенческая экономика: почему люди нарушают правила традиционной экономики и как на этом заработать / Р. Талер ; пер. с англ. А. Прохоровой. — Москва : Издательство «Э», 2017. — 368 с. — URL: <https://avidreaders.ru/book/novaya-povedencheskaya-ekonomika-pochemu-lyudi-narushayut.html>.

Библиографическая ссылка

Ознобихин, Д. В. Разработка инструментария для анализа и оптимизации структуры оборотных активов предприятия на основе теории временных предпочтений // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 150–160.

Сведения об авторах

Ознобихин Дмитрий Владимирович

Студент магистратуры

Направление «Прикладная информатика»

УДК 004.9, 658.15

Управление хранением и унификацией справочной информации в системе 1С

Осокин А. Д.

Тверской государственной университет

Рекомендовано к публикации доцентом кафедры математической статистики и системного анализа А. А. Смородовой

АННОТАЦИЯ. В статье исследуется проблема обеспечения целостности данных в системах на платформе «1С:Предприятие». Проанализированы риски неструктурированного хранения информации и методы сопоставления записей при межбазовом обмене. На примере «1С:Управление холдингом» рассмотрены типовые механизмы унификации справочников. Введено понятие «критически значимые реквизиты» с классификацией по степени влияния на систему. Предложен алгоритм модификации подсистемы «Мастер-данные», позволяющий настраивать контроль изменений справочной информации без вмешательства в программный код.

КЛЮЧЕВЫЕ СЛОВА: унификация справочной информации, 1С:Предприятие, мастер-данные, критически значимые реквизиты.

Введение

В условиях стремительного роста объёма данных и усложнения организационных структур возникает проблема хранения и унификации данных. В современных экономических системах, в том числе и системах, основанных на платформе 1С:Предприятие, информация хранится в базе данных компьютера, благодаря этому её можно легко получать, изменять и обрабатывать. В 1С:Предприятие управление хранением данных можно представить как совокупность методов и механизмов платформы, которые обеспечивают размещение, целостность, доступность и эффективность использования хранимых данных.

Цель работы — исследование методов хранения и унификации справочной информации в системе 1С, разработка функционала для управления хранением и унификацией справочной информации

в системе. Авторская модификация позволит администраторам в пользовательском режиме устанавливать и снимать запрет на изменение критически значимых реквизитов справочной информации, для того чтобы пользователь не мог ошибочно заменить важное значение реквизита, что позволяет избежать дальнейших ошибок в работе информационной системы.

1. Понятия унификации и хранения информации

В работе по методическому обеспечению технико-экономической оценки мероприятий унификации А. В. Леонов и А. Ю. Пронин рассматривают понятие унификации как приведение заданного множества элементов к единообразию по установленным признакам или как сокращение числа элементов заданного множества [3, с. 5]. Наиболее полное описание проблемы унификации информации представили А. С. Копырин и И. Л. Макарова, полагая, что если экономические данные могут собираться из различных источников, то общий набор данных будет иметь проблемы с однородностью содержания. Авторы связывают унификацию информации с преодолением неоднородности экономической информации, описывая, что проблема в основном представлена несоответствиями в атрибутах данных, таких как имена атрибутов и единицы измерения, приводя в пример показатель затрат, который может быть назван «Затраты», «Издержки», «Costs» [2, с. 45]. На основании вышеизложенного по мнению авторов унификацию информации можно представить как процесс целенаправленного сокращения многообразия элементов заданного множества путём приведения их к единообразию по установленным правилам и признакам. В рамках ИС:Предприятие унификацию информации следует рассматривать как процесс приведения справочных данных к единому эталонному формату, исключающему дублирование и неоднозначность хранимых данных. Для эффективной организации хранения и обработки данных в экономических системах унификация играет важную роль, она способствует нормализации базы данных, исключению избыточности, обеспечивает целостность и непротиворечивость данных. Отсутствие унификации данных приводит к следующим проблемам:

- 1) дублирование — один и тот же объект хранится в нескольких подразделениях с разными реквизитами;
- 2) нарушение целостности и противоречивость данных — один и тот же объект может иметь противоречащие друг другу сведения;
- 3) избыточность — многократное хранение информации об одном и том же объекте приводит к увеличению объёма базы данных;
- 4) несогласованность данных — отсутствие единого формата хранения данных затрудняет формирование отчётности предприятия.

Перечисленные выше проблемы напрямую снижают эффективность управления экономической системой. Следовательно, внедрение унификации является важным условием для обеспечения целостности, достоверности и напрямую влияет на процесс обработки данных, анализу и принятию решений.

2. Методы унификации информации

Методика унификации напрямую влияет на качество обрабатываемой информации. В работе «Методика создания унифицированных справочников аналитического учёта бизнес-процессов компании» Т. В. Зырянова и Ю. С. Тарновская делают акцент на методике унификации системы справочников аналитического учёта в компании, показывая, что унификация оказывает существенную помощь при анализе большого объёма информационных потоков на предприятии. Учёные отмечают, что при внедрении системы управленческого учёта и бюджетирования (или оптимизации существующих на предприятии учётных систем) многие разработчики сталкиваются с необходимостью создания единых справочников для автоматической передачи информации из одной учётной системы в другие. [1, с. 3]. В данной статье управление хранением и унификацией будет рассмотрено на примере типовой конфигурации «1С:Управление холдингом» — это решение предназначенное для комплексной автоматизации управления финансами холдингов и крупных монокомпаний. В рассматриваемой типовой конфигурации

корпоративная структура настраивается в справочнике «Регламенты подготовки отчётности». В информационную базу справочная информация может попадать как с помощью обмена с связанными базами, так и создаваться пользователями вручную. Пользование конфигурации «1С:Управление холдингом» подразумевает управление холдинговой компанией с помощью консолидации отчётности входящих в неё организаций. При создании консолидированного отчёта необходима загрузка данных из всех организаций, входящих в периметр консолидации. При загрузке отчётной информации из большого списка организаций возникает проблема конфликта данных, которая может быть связана с различными наименованиями и кодами элементов справочников или другими объектами метаданных конфигурации. Так как контролировать ведение учёта во множестве организаций холдинга затруднительно, часто возникает ситуация, когда один товар или его описание имеют различия. Также в информационной базе для составления консолидированной отчётности работают сотрудники из разных организаций, это может привести к ошибочному изменению информации, не относящейся к организации сотрудника. Для составления отчётности используются общие для сотрудников всех организаций элементы информационной базы, например, справочник валют и периодов отчётности. При неверном исправлении таких элементов возникают ошибки в учёте и работе информационной системы. Дублирование элементов общих справочников также может привести к ошибкам, так как формирование консолидированной отчётности в разрезе нескольких элементов информационной базы с одинаковым смысловым значением (например, элементы справочника валюты «рубль» и «руб.», созданные пользователями из разных организаций) является некорректным. Управление хранением и унификацией справочной информации необходимо для информационных баз типа «Управление холдингом», так как информация в эти базы зачастую попадает через обмен с другими информационными базами холдинговой компании. При загрузке такой информации необходимо чётко разграничивать создания и изменения элементов справочной информации для избегания дублирования и неверного редактирования элементов. Механизм унификации позволяет создать элемент справочной информации один раз, а затем распространить его на все связанные информационные базы, а механизм управления хранением справочной информации

обеспечивает создание единой системы ведения справочников во всех информационных базах предприятия. Для управления хранением и унификацией справочной информации в системе 1С могут быть использованы следующие решения и механизмы:

- 1) контроль уникальности – стандартная настройка, присутствующая во всех современных конфигурациях на платформе 1С. Контроль уникальности определяет необходимость контроля не повторяемости номеров средствами системы перед записью объекта;
- 2) программная проверка при записи элемента – разработчики в конфигурации информационной базы 1С разрабатывают программные проверки на уникальность названия, номера и прочих необходимых реквизитов, в случае обнаружения не уникального значения пользователю выводится сообщение с ошибкой записи объекта;
- 3) мастер-данные – различные расширения для контроля справочной информации рекомендуемые и распространяемые специалистами фирмы «1С». Расширения позволяют настраивать контроль записываемых данных в информационную базу, запрещать и согласовывать изменения в элементах справочной информации.

Описанные выше решения и механизмы имеют различный функционал для обеспечения управления хранением и унификации информации. Выбор конкретного решения и механизма или их комбинация зависит от требований к достоверности данных и интенсивности работы пользователей в информационной базе. Наиболее функциональным решением для унификации информации в системах на платформе 1С:Предприятие является внедрение подсистемы «Мастер-данные». В отличие от контроля уникальности и программной проверки, применение подсистемы «Мастер-данные» позволяет интерактивно настаивать доступ к созданию и изменению справочной информации без необходимости написания и отладки кода с последующим обновлением информационной базы.

3. Описание методики доработки управления хранением справочной информации

В информационных системах на платформе 1С предприятия можно выявить реквизиты справочной информации, изменения которых могут влиять как на логику работы, так и на работоспособность информационной системы. Например, изменение флага «Участник ЕАС» у элемента справочника «Страны мира» может повлиять на формирование отчётности, так как этот флаг часто применяется в отборах. К ошибкам в работе программы может привести изменение реквизитов, содержащих опознавательный код (например, цифровой или символьный код у справочника валюты), поскольку такой код может быть использован в коде алгоритмов информационной системы. Из вышесказанного авторами вводится понятие «критически значимые реквизиты» – реквизиты, изменения которых может привести к изменениям в работе информационной системы. Критически значимые реквизиты можно разделить на следующие типы в соответствии со степенью их влияния на работу информационной системы:

- изменяемый — изменения реквизитов не отразится на логике работы информационной системы;
- неизменяемый — изменения реквизитов отражаются на логике работы информационной системы;
- фатальные — изменения реквизитов может привести к ошибкам в работе информационной системы.

Для ликвидации проблемы неконтролируемого изменения критически значимых реквизитов принято решение расширения возможностей подсистемы, выполнена модификация механизма подсистемы «Мастер-данные», которая предоставляет пользователям возможность изменять параметры проверки изменений реквизитов для объектов, добавленных в подсистему «Мастер-данные» без изменения программного кода и обновления информационной базы. Модификация выполнена с использованием расширения подсистемы для управляемых форм конфигураций «Управление холдингом 3.2» 1С предприятия, расширения позволяют модифицировать информационные

базы без снятия объектов с поддержки для последующего обновления с помощью официальных релизов от 1С. В подсистему добавлен справочник «Критически значимые реквизиты», в который можно вводить объекты, включённые в подсистему «Мастер-данные». Реквизиты добавленных справочников возможно настраивать для проверки перед непосредственной записью элементов справочника. Отслеживаемые реквизиты имеют следующие типы настроек, которые рекомендуется проставлять по степени влияния реквизита на логику работы информационной системы:

- **изменяемый** — при записи элемента изменения реквизита успешно сохраняются;
- **неизменяемый** — при попытке записи изменения реквизита с данной настройкой, изменение будет отменено. У пользователя отобразится окно с сообщением «Попытка изменить критически значимый реквизит (неизменяемый)» и перечислены значения до изменения и значение, которое пользователь пытался сохранить;
- **фатальный** — при записи элемента с измененным реквизитом с этой настройкой, запись будет запрещена и пользователю отобразится сообщение об ошибке с текстом «Попытка изменить критически значимый реквизит (фатальный)» и указано измененное значение реквизита и значение до его изменения.

Алгоритм работы модификации по отслеживанию критически значимых реквизитов подсистемы «Мастер-данные» в виде блок-схемы представлена ниже (рис. 1).

Методика доработки предполагает проверять изменения отслеживаемых реквизитов указанных справочников перед непосредственной записью их элементов, при обнаружении изменений система действует в соответствии с типом реквизита.

Заключение

Подводя итог, можно отметить, что унификация и управление хранением играют ключевую роль при работе с информационными базами на платформе 1С. Наиболее эффективным способом управления хранением и унификации справочной информации в системе

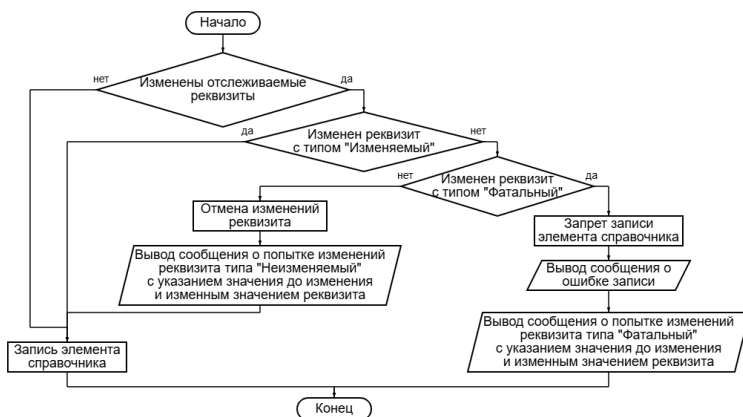


Рис. 1. Алгоритм работы модификации.

ИС является интеграция подсистемы «Мастер-данные», которая позволяет интерактивно управлять настройками хранения данных и способствует унификации информации при обмене и загрузке данных. Также выявлено, что важной частью обеспечений управления хранением является контроль изменения реквизитов, влияющих на работу информационной базы. В связи с этим введено понятие «Критически значимые реквизиты» с классификацией по степени влияния изменения значения реквизитов на логику и работоспособность информационной базы. Для обеспечения контроля изменения критически значимых реквизитов предложен алгоритм интерактивного ограничения запрета изменений значений реквизитов.

Список литературы

- [1] Зырянова, Т.В. Методика создания унифицированных справочников аналитического учёта бизнес-процессов компании / Т.В. Зырянова, Ю.С. Тарновская // Международный бухгалтерский учёт. — 2012. — № 36. — С. 2–14. — URL: <https://www.fin-izdat.ru/journal/interbuh/detail.php?ID=52988>.

- [2] Копырин, А. С. Алгоритм препроцессинга и унификации временных рядов на основе машинного обучения для структурирования данных / С. С. Копырин, И. Л. Макарова // Программные системы и вычислительные методы. — 2020. — № 3. — С. 40–50. — DOI: <https://doi.org/10.7256/2454-0714.2020.3.33958>.
- [3] Леонов, А. В. Методическое обеспечение технико-экономической оценки мероприятий по унификации / А. В. Леонов, А. Ю. Пронин // Компетентность. — 2021. — № 7. — С. 4–15. — DOI: <https://doi.org/10.24412/1993-8780-2021-7-04-15>.

Библиографическая ссылка

Осокин, А. Д. Управление хранением и унификацией справочной информации в системе 1С // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 162–170.

Сведения об авторах

Осокин Алексей Дмитриевич
Студент магистратуры
Направление «Прикладная информатика»

УДК 004.89, 658.5.012.1

Системный анализ производственных процессов на основе искусственного интеллекта

Родионов Д. А.

Тверской государственный университет
Рекомендовано к публикации профессором кафедры
математической статистики и системного анализа Г. М. Соломахой

Аннотация. В статье предложен подход к системному анализу производственных процессов на основе искусственного интеллекта. Рассмотрена структура комплексного мониторинга, включающая анализ состояния оборудования, контроль качества модели и выявление дрейфа данных. Показана связь результатов мониторинга и многокритериальной оценки моделей с ключевыми показателями эффективности производства.

Ключевые слова: искусственный интеллект, системный анализ, производство, мониторинг, эффективность.

Введение

Искусственный интеллект всё активнее применяется в промышленном производстве. С его помощью решаются задачи прогнозирования отказов оборудования, контроля качества продукции, анализа производственных данных и поддержки принятия решений. Общие подходы к построению и применению моделей машинного обучения представлены в работе К. Бишопа [2], а при описании систем искусственного интеллекта необходимо учитывать нормативную терминологию и классификацию таких систем [1].

Однако само наличие модели машинного обучения ещё не делает производственный процесс более управляемым. Модель может показывать хорошие результаты на этапе обучения, но постепенно терять качество в реальной эксплуатации. Причинами этого являются износ оборудования, изменение характеристик сырья, смена режима работы, появление новых типов дефектов или дрейф входных данных.

Для количественной оценки таких изменений может использоваться расстояние Кульбака – Лейблера [3].

На практике интеллектуальные решения часто внедряются разрозненно: один модуль анализирует вибрацию оборудования, другой выявляет дефекты продукции, третий отслеживает изменение входных параметров. Каждый из них может быть полезен сам по себе, но без общей логики трудно понять, где возникло отклонение и как оно влияет на производственный результат.

Целью работы является исследование и разработка алгоритмов искусственного интеллекта для системного анализа и оптимизации производственного процесса. Под разработкой алгоритмов в данной работе понимается не создание новой архитектуры модели машинного обучения, а разработка процедур их системного применения, мониторинга и оценки в составе единой производственной системы.

1. Производственный процесс как объект анализа

Производственный процесс можно рассматривать как систему, в которой связаны оборудование, технологические параметры, входные данные, модели машинного обучения, результаты анализа и ключевые показатели эффективности. Если меняется один элемент, это может отразиться на работе всей системы.

Например, ухудшение состояния оборудования приводит к изменению вибрации или температуры. Эти параметры поступают на вход модели. Если модель обучалась на других режимах работы, качество её прогноза или классификации может снизиться. Значит, ошибка не всегда возникает только из-за самой модели: причина может быть на уровне оборудования, данных или условий эксплуатации.

В рамках системного анализа в работе выделяются мониторинг состояния оборудования, контроль качества модели машинного обучения, анализ дрейфа входных данных, многокритериальная оценка моделей и связь результатов анализа с ключевыми показателями эффективности производства. Такой подход позволяет перейти от набора частных метрик к более целостному описанию состояния производственной системы.

2. Комплексный мониторинг системы искусственного интеллекта

Первый уровень анализа связан с мониторингом состояния оборудования. В промышленной среде это могут быть температура, вибрация, давление, токовые характеристики и другие параметры, отражающие состояние производственного объекта. Смысл этого уровня состоит в выявлении признаков деградации до наступления критического отказа.

Для оценки качества прогноза применяется среднеквадратичная ошибка, обозначаемая как MSE:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (1)$$

где y_i — фактическое значение параметра, \hat{y}_i — прогнозируемое значение параметра, n — число наблюдений.

Также используется средняя абсолютная ошибка, обозначаемая как MAE:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (2)$$

Среднеквадратичная ошибка сильнее реагирует на крупные ошибки прогноза, а средняя абсолютная ошибка удобна для практической интерпретации, так как показывает среднее отклонение в тех же единицах измерения, что и исходный параметр. Рост этих ошибок является поводом для анализа, но сам по себе ещё не объясняет причину ухудшения.

Второй уровень связан с контролем качества модели машинного обучения. Для задач классификации, например при выявлении дефектов продукции, используются точность положительных срабатываний, чувствительность модели и F-мера. В формулах для этих показателей применяются общепринятые обозначения Precision, Recall и F1-score:

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (3)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (4)$$

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (5)$$

где TP — истинно положительные срабатывания, FP — ложноположительные срабатывания, FN — ложноотрицательные срабатывания.

Метрика Precision показывает надёжность положительного сигнала модели. Метрика Recall характеризует способность модели обнаруживать реальные дефекты. F1-score используется как обобщающий показатель, учитывающий баланс между Precision и Recall. В задачах промышленного контроля качества особое значение имеет метрика Recall, поскольку пропуск дефектной продукции обычно опаснее ложного срабатывания.

Третий уровень связан с анализом дрейфа данных, то есть изменения статистических характеристик входных данных по сравнению с обучающей выборкой. Для оценки изменения распределений используется расстояние Кульбака – Лейблера: Третий уровень связан с анализом дрейфа данных, то есть изменения статистических характеристик входных данных по сравнению с обучающей выборкой. Для оценки изменения распределений используется расстояние Кульбака – Лейблера:

$$D_{KL} = \sum_{i=1}^r P_i \ln \frac{P_i}{Q_i}, \quad (6)$$

где P_i — вероятность попадания значения признака в i -й интервал эталонного распределения, Q_i — вероятность попадания значения признака в i -й интервал текущего распределения, r — количество интервалов, на которые разбивается область значений анализируемого признака.

Рост значения D_{KL} показывает, что распределение текущих данных отличается от эталонного распределения, сформированного по данным обучения или нормального режима работы. Сам по себе этот показатель не означает ошибку модели, но вместе со снижением качества её работы становится важным диагностическим сигналом. Например, рост дрейфа данных и снижение метрики Recall могут указывать на изменение условий эксплуатации или появление новых типов дефектов.

3. Многокритериальная оценка моделей

При выборе модели машинного обучения для производственного процесса недостаточно учитывать только точность. В реальных условиях модель должна быть понятной, достаточно оперативной и устойчивой к искажениям данных. Поэтому в работе используется многокритериальный подход.

Пусть рассматривается множество моделей

$$M = \{M_1, M_2, \dots, M_n\}, \quad (7)$$

где M_i — i -я модель машинного обучения.

Для каждой модели вводится вектор критериев

$$C_i = (C_{i1}, C_{i2}, C_{i3}, C_{i4}), \quad (8)$$

где C_{i1} — качество решения целевой задачи, C_{i2} — интерпретируемость, C_{i3} — вычислительная эффективность, C_{i4} — устойчивость к искажениям данных.

Задача выбора модели формулируется как

$$M^* = \arg \max_{M_i \in M} I_i, \quad (9)$$

где I_i — интегральный показатель эффективности i -й модели.

Устойчивость модели определяется через сохранение качества при искажениях данных. Пусть Q_0 — качество модели на исходных данных, Q_s — качество модели при s -м сценарии искажения, S — число сценариев. Тогда

$$R = \frac{1}{S} \sum_{s=1}^S \frac{Q_s}{Q_0}. \quad (10)$$

Поскольку критерии имеют различную размерность, перед агрегированием выполняется их нормализация. Для критерия, подлежащего максимизации, используется выражение

$$C_{ij}^{norm} = \frac{C_{ij} - C_j^{min}}{C_j^{max} - C_j^{min}}, \quad (11)$$

где C_{ij}^{norm} — нормированное значение j -го критерия для i -й модели, C_{ij} — исходное значение критерия, C_j^{min} и C_j^{max} — минимальное и максимальное значения данного критерия среди сравниваемых моделей.

Для критерия, подлежащего минимизации, применяется обратная нормализация:

$$C_{ij}^{norm} = \frac{C_j^{max} - C_{ij}}{C_j^{max} - C_j^{min}}. \quad (12)$$

Интегральная оценка модели определяется как взвешенная сумма нормализованных критериев:

$$I_i = \sum_{j=1}^m w_j C_{ij}^{norm}, \quad (13)$$

где w_j — вес j -го критерия, m — количество критериев. Для весов выполняются условия

$$w_j \geq 0, \quad \sum_{j=1}^m w_j = 1. \quad (14)$$

Такая оценка позволяет сравнивать модели не по одному показателю, а по совокупности характеристик. Самая точная модель не всегда является лучшей для промышленной эксплуатации: более простая модель может быть предпочтительнее, если она понятнее для специалиста, быстрее работает и меньше теряет качество при изменении данных.

4. Связь результатов анализа с ключевыми показателями эффективности

Результаты мониторинга и многокритериальной оценки должны быть связаны с ключевыми показателями эффективности производства. Иначе система будет выдавать только технические метрики, но не будет ясно, что они означают для производства.

В работе рассматриваются уровень брака, уровень внеплановых простоев, надёжность оборудования, стабильность качества продукции и общая эффективность производственной системы. В общем

виде зависимость отдельного ключевого показателя эффективности от состояния системы может быть представлена как

$$K_k = f_k(I, D, E), \quad (15)$$

где K_k — k -й ключевой показатель эффективности, I — интегральная оценка качества модели, D — показатель дрейфа данных, E — показатель состояния оборудования.

Для практического применения такая зависимость может быть аппроксимирована линейной моделью

$$K_k = \beta_{0k} + \beta_{1k}I - \beta_{2k}D + \beta_{3k}E, \quad (16)$$

где β_{0k} , β_{1k} , β_{2k} , β_{3k} — коэффициенты влияния факторов для k -го ключевого показателя эффективности.

Такая запись задаёт общий вид связи между диагностическими показателями системы искусственного интеллекта и производственными показателями. Для разных ключевых показателей эффективности коэффициенты модели могут отличаться.

Предложенный подход был применён для анализа трёх типовых производственных сценариев. В задаче прогнозирования деградации подшипникового узла рост ошибки прогноза, ухудшение состояния оборудования и наличие дрейфа данных интерпретировались как признаки повышения риска внепланового простоя; после корректирующих действий итоговый показатель производственной ситуации увеличился с 0,582 до 0,822. В задаче контроля качества сварных соединений снижение метрики Recall рассматривалось как сигнал риска скрытого брака; после корректировки условий применения модели итоговая оценка увеличилась с 0,696 до 0,811. В задаче мониторинга станочного узла при изменении нагрузки рост расстояния Кульбака – Лейблера вместе со снижением качества модели указывал на изменение входных условий и необходимость обновления данных или переобучения модели.

Предложенный подход реализован в виде программного комплекса на языке программирования Python с использованием библиотек NumPy, Pandas, Matplotlib, scikit-learn и TensorFlow/Keras. Программная реализация использовалась для расчёта метрик мониторинга, анализа дрейфа данных, многокритериальной оценки моделей и формирования диагностических выводов.

Заключение

В статье рассмотрен подход к системному анализу производственных процессов на основе искусственного интеллекта. Предложенный подход объединяет мониторинг состояния оборудования, контроль качества модели, анализ дрейфа данных и многокритериальную оценку моделей машинного обучения.

Показано, что результаты анализа могут быть связаны с ключевыми показателями эффективности производства: уровнем брака, внеплановыми простоями, надёжностью оборудования, стабильностью качества продукции и общей эффективностью производственной системы. Практическая применимость подхода показана на задачах прогнозирования деградации оборудования, контроля качества продукции и анализа изменения режима производственной нагрузки.

Таким образом, системный анализ производственных процессов на основе искусственного интеллекта позволяет перейти от разрозненного применения отдельных моделей к более целостной системе мониторинга, оценки и поддержки принятия решений.

Список литературы

- [1] ГОСТ Р 59277–2020. Системы искусственного интеллекта. Классификация систем искусственного интеллекта. — М. : Стандартинформ, 2021. — 12 с. — URL: <https://protect.gost.ru/gost/details/e3a4849d-c935-4a41-84fd-5740a098cb49>.
- [2] Bishop, C. M. Pattern Recognition and Machine Learning / C. M. Bishop. — New York : Springer, 2006. — 738 p. — URL: <https://www.microsoft.com/en-us/research/wp-content/uploads/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf>.
- [3] Kullback, S. On Information and Sufficiency / S. Kullback, R. A. Leibler // The Annals of Mathematical Statistics. — 1951. — Vol. 22, № 1. — P. 79–86. — DOI: <https://doi.org/10.1214/aoms/1177729694>.

Библиографическая ссылка

Родионов, Д. А. Системный анализ производственных процессов на основе искусственного интеллекта // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 171–179.

Сведения об авторах

Родионов Д. А.

Студент магистратуры

Направление «Прикладная математика и информатика»

УДК 004.852, 004.891.3

Система рекомендаций для медиаконтента

Свистунов А. Н.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры информатики

В. Л. Волушковой

Аннотация. В статье рассматривается система рекомендаций для медиаконтента на примере аниме-каталога. Основное внимание уделено гибридной модели, объединяющей контентно-ориентированную фильтрацию, коллаборативную фильтрацию по пользовательским взаимодействиям и сигнал популярности. Описаны данные, архитектура, формулы ранжирования и сценарий экспериментальной проверки.

Ключевые слова: система рекомендаций, медиаконтент, коллаборативная фильтрация, гибридная модель, холодный старт.

Введение

Рост объёма цифрового медиаконтента приводит к тому, что пользователю становится всё сложнее самостоятельно находить произведения, соответствующие его интересам. Простая сортировка по рейтингу или популярности не учитывает индивидуальные предпочтения и не позволяет объяснить, почему один и тот же объект может быть релевантен одному пользователю и неинтересен другому. Поэтому в современных информационных системах важную роль играют системы рекомендаций, позволяющие формировать персонализированную выдачу на основе данных о контенте и поведении пользователя.

В данной работе рассматривается система рекомендаций для медиаконтента, апробированная на предметной области аниме. Такой выбор обусловлен тем, что аниме-контент обладает выраженной жанровой и тематической спецификой: при выборе произведения пользователь учитывает не только общую оценку, но и жанры, студии, длительность, сезон выхода, формат и собственную историю просмотров. Кроме того, в подобных системах возникает проблема

холодного старта: новый пользователь ещё не имеет истории взаимодействий, а значит, алгоритм не может сразу опираться только на оценки и просмотры.

Целью исследования является разработка и реализация системы рекомендаций, способной формировать персональные рекомендации на основе гибридного подхода. В отличие от витринного каталога, где рекомендации являются дополнительным элементом интерфейса, в данной работе центральным объектом является именно рекомендательный модуль. Веб-интерфейс, каталог, личный список и административные инструменты рассматриваются как инфраструктура, необходимая для сбора данных, демонстрации результата и проверки качества рекомендаций.

1. Постановка задачи и используемые данные

Пусть имеется множество пользователей U и множество объектов медиаконтента A . Для каждого пользователя необходимо сформировать упорядоченный список объектов, которые с наибольшей вероятностью будут ему интересны. В рассматриваемой системе объектом рекомендаций является аниме, однако сама постановка допускает расширение на другие виды медиаконтента, если для них доступны метаданные и пользовательские взаимодействия.

Задача рекомендации для пользователя $u \in U$ заключается в построении функции ранжирования

$$S : U \times A \rightarrow \mathbb{R}, \quad (1)$$

которая сопоставляет паре «пользователь – объект» числовую оценку интереса. После вычисления значений функции $S(u, a)$ система сортирует кандидаты по убыванию оценки и возвращает первые K элементов. При этом из выдачи исключаются объекты, с которыми пользователь уже взаимодействовал, так как повторная рекомендация просмотренного или оценённого произведения ухудшает пользовательский опыт.

Для построения функции ранжирования используются три группы данных. Первая группа — метаданные об объектах. В реализованной системе они загружаются из набора anime-offline-database. В базу данных импортируются название, тип, статус, сезон и год

выхода, количество эпизодов, длительность, изображения, внешняя оценка, синонимы, студии, продюсеры, связи между произведениями и жанры.

Вторая группа данных — явные предпочтения пользователя. После регистрации пользователь может выбрать любимые жанры, студии, предпочтительную длительность и несколько начальных произведений, которые ему нравятся. Эти данные важны для решения проблемы холодного старта: система получает первичный профиль даже до появления оценок и истории просмотров.

Третья группа данных — пользовательские взаимодействия. К ним относятся оценки, статусы просмотра и признак добавления в избранное. Именно эти взаимодействия образуют основу для коллаборативной фильтрации, так как позволяют сравнивать пользователей по фактическому поведению, а не только по анкете предпочтений.

Для хранения данных применяется реляционная база PostgreSQL. Она содержит сведения об аниме, пользователях, профилях, оценках, статусах просмотра, избранном и задачах импорта. Для ускорения повторной выдачи результатов используется Redis, в котором кэшируются публичные подборки и персональные рекомендации.

2. Гибридная модель рекомендаций

Для системы медиаконтента был выбран гибридный подход. Его смысл заключается в том, что итоговая рекомендация строится не по одному признаку, а по совокупности нескольких сигналов. Контентно-ориентированная часть позволяет учитывать метаданные произведения и начальный профиль пользователя, коллаборативная фильтрация выявляет сходство пользователей по действиям, а популярность стабилизирует выдачу и используется в публичных подборках.

Для описания пользовательской активности введём агрегированный сигнал взаимодействия пользователя u с объектом a :

$$r(u, a) = \alpha \cdot q(u, a) + \beta \cdot t(u, a) + \gamma \cdot f(u, a), \quad (2)$$

где $q(u, a)$ — нормализованная оценка пользователя, $t(u, a)$ — численное представление статуса просмотра, $f(u, a)$ — бинарный признак избранного, а α , β и γ — веса соответствующих факторов. Рейтинг

нормализуется так, чтобы оценки выше середины шкалы формировали положительный вклад:

$$q(u, a) = \max \left(0, \frac{\text{rating}(u, a) - 5}{5} \right). \quad (3)$$

Например, статус «просмотрено» имеет больший вес, чем статус «запланировано», так как он отражает более сильный и завершённый интерес пользователя к объекту.

Контентная составляющая описывает соответствие объекта профилю пользователя:

$$S_{cb}(u, a) = \sum_{p \in P} w_p \cdot x_p(u, a), \quad (4)$$

где P — множество учитываемых признаков, w_p — вес признака, а $x_p(u, a)$ — степень соответствия объекта a предпочтению пользователя u по признаку p . В реализованной системе к таким признакам относятся совпадение по жанрам, студиям, предпочтительной длительности, а также близость к начальным и положительно оценённым аниме.

Для коллаборативной фильтрации необходимо определить сходство пользователей. Оно вычисляется по векторам агрегированных взаимодействий. В качестве меры используется косинусное сходство:

$$\text{sim}(u, v) = \frac{\sum_{a \in A} r(u, a) \cdot r(v, a)}{\sqrt{\sum_{a \in A} r(u, a)^2} \cdot \sqrt{\sum_{a \in A} r(v, a)^2}}. \quad (5)$$

Чем больше совпадают положительные взаимодействия двух пользователей, тем выше значение $\text{sim}(u, v)$. После этого для целевого пользователя u и кандидата a вычисляется коллаборативная оценка:

$$S_{cf}(u, a) = \frac{\sum_{v \in N_k(u)} \text{sim}(u, v) \cdot r(v, a)}{\sum_{v \in N_k(u)} |\text{sim}(u, v)|}, \quad (6)$$

где $N_k(u)$ — множество k наиболее похожих пользователей. Если похожие пользователи положительно взаимодействовали с некоторым аниме, отсутствующим у целевого пользователя, то это аниме получает высокий коллаборативный балл.

Итоговая оценка рекомендации задаётся формулой

$$S(u, a) = \lambda_1 S_{cb}(u, a) + \lambda_2 S_{cf}(u, a) + \lambda_3 S_{pop}(a), \quad (7)$$

где $S_{pop}(a)$ — сигнал популярности объекта, а λ_1 , λ_2 и λ_3 — веса компонентов. Такой подход позволяет не сводить рекомендацию только к популярности или только к совпадению жанров. Для нового пользователя основную роль играет контентная часть, а по мере накопления оценок, статусов и избранного возрастает значение коллаборативной составляющей.

3. Реализация системы рекомендаций

Программная реализация была выполнена в виде веб-системы, где все прикладные компоненты подчинены задаче сбора сигналов и формирования рекомендаций. Серверная часть реализована на Java с использованием Spring Boot и организована как модульный монолит. Такой подход позволяет логически выделить модули каталога, профиля, взаимодействий, импорта, администрирования и рекомендаций, но не вводит инфраструктурную сложность микросервисной архитектуры.

Клиентская часть реализована на Vue 3 и TypeScript. Она обеспечивает пользовательские сценарии, необходимые для работы системы рекомендаций: регистрацию, начальную профилизацию, просмотр каталога, выставление оценок, изменение статуса просмотра, добавление в избранное, просмотр личного списка и получение персональных рекомендаций. Таким образом, интерфейс выполняет не только демонстрационную функцию, но и служит источником данных для последующего ранжирования.

Особое значение имеет механизм начальной профилизации. После регистрации пользователь может выбрать жанры, студии, предпочтительную длительность и несколько понравившихся аниме.

Для персональной выдачи используется отдельный маршрут прикладного программного интерфейса. Публичные подборки недавних, популярных и трендовых произведений доступны без авторизации, а персональные рекомендации строятся только для авторизованного пользователя. После изменения оценки, статуса просмотра, избранного или профиля пользователя кэш персональной выдачи

Таблица 1. Пользовательские сигналы для рекомендательной модели.

Сигнал	Смысл в модели	Вклад
Оценка	Численное выражение интереса пользователя к аниме	высокий
Статус просмотра	Стадия взаимодействия: запланировано, смотрю, просмотрено, отложено	средний
Избранное	Явное выделение объекта как особенно интересного	высокий
Начальный профиль	Жанры, студии, длительность и выбранные аниме для холодного старта	базовый

сбрасывается, чтобы последующие рекомендации отражали новое состояние данных.

В табл. 1 приведена интерпретация основных пользовательских сигналов, используемых при формировании вектора взаимодействий.

4. Экспериментальная проверка

Для проверки работоспособности системы использовались контролируемые сценарии. Такой формат выбран потому, что на этапе исследования отсутствует большая реальная пользовательская база, необходимая для строгой статистической оценки. В то же время сценарная проверка позволяет показать, что модель реагирует на изменения профиля и взаимодействий в соответствии с ожидаемой логикой.

Первый сценарий связан с холодным стартом. Для нового пользователя без оценок и статусов заполнялся начальный профиль: выбирались жанры, студии и несколько понравившихся произведений. После завершения профилизации система формировала непустую персональную выдачу, близкую к указанным предпочтениям. Это показывает, что контентно-ориентированная часть действительно позволяет получить первые рекомендации без истории взаимодействий.

Таблица 2. Сценарий проверки коллаборативной фильтрации.

Пользователь	Базовый набор	Дополнительные аниме
U_1	B	X, Y
U_2	B	X, Y
U_3	B	X, Z
U_t	B	отсутствуют

Второй сценарий проверял вклад коллаборативной фильтрации. Были созданы три пользователя U_1 , U_2 и U_3 с одинаковым базовым набором положительных взаимодействий B . Затем каждому из них были добавлены дополнительные произведения, отсутствующие у контрольного пользователя U_t . Структура эксперимента приведена в табл. 2.

После построения рекомендаций для контрольного пользователя объекты X , Y и Z появились в верхней части выдачи. Наиболее высокую позицию получил объект X , поскольку он встречался у большего числа пользователей с близкой историей. При этом у U_t не было собственных взаимодействий с этими объектами. Следовательно, их появление объясняется не фактом просмотра самим пользователем, а поведением похожих пользователей.

Для описания результата использовалась метрика попадания в верхнюю часть выдачи:

$$HitRate@K = \begin{cases} 1, & \text{если } Rec_u^K \cap Rel_u \neq \emptyset, \\ 0, & \text{иначе,} \end{cases} \quad (8)$$

где Rec_u^K — первые K рекомендаций пользователя, а Rel_u — множество ожидаемо релевантных объектов. В эксперименте роль Rel_u выполняли объекты X , Y и Z . Появление хотя бы одного из них в первых позициях подтверждает наличие коллаборативного вклада. Более детальный анализ может дополнительно использовать точность $Precision@K$:

$$Precision@K = \frac{|Rec_u^K \cap Rel_u|}{|Rec_u^K|}. \quad (9)$$

Проведённая проверка показала, что система реагирует на пользовательские взаимодействия динамически. После выставления оценки

или добавления объекта в избранное персональные рекомендации обновляются, а уже обработанные произведения исключаются из последующей выдачи. Это подтверждает согласованность между моделью взаимодействий, кэшированием и интерфейсом пользователя.

Заключение

В статье рассмотрена система рекомендаций для медиаконтента, реализованная на примере аниме-каталога. Основной результат работы заключается в разработке гибридной модели, объединяющей контентно-ориентированную фильтрацию, коллаборативную фильтрацию по пользовательским взаимодействиям и сигнал популярности. В отличие от систем, основанных только на общей популярности или только на метаданных, предложенный подход учитывает как свойства произведений, так и действия пользователей.

Реализованная система поддерживает работу в условиях холодного старта за счёт начальной профилизации, а после накопления оценок, статусов просмотра и избранного уточняет выдачу с помощью коллаборативной составляющей. Контролируемый эксперимент с группой похожих пользователей показал, что система способна рекомендовать контрольному пользователю объекты, отсутствующие в его собственной истории, но характерные для пользователей с близкими взаимодействиями.

Дальнейшее развитие работы может быть связано с количественной оценкой на большом наборе пользовательских логов, автоматическим подбором весов гибридной модели и исследованием более сложных методов, таких как матричная факторизация и модели латентных представлений.

Список литературы

- [1] Burke, R. Hybrid Recommender Systems: Survey and Experiments // User Modeling and User-Adapted Interaction. — 2002. — Vol. 12, № 4. — P. 331–370. — DOI: <https://doi.org/10.1023/A:1021240730564>.
- [2] Koren, Y. Matrix Factorization Techniques for Recommender Systems / Y. Koren, R. Bell, C. Volinsky // Computer. — 2009. —

Vol. 42, № 8. — P. 30–37. — DOI: <https://doi.org/10.1109/МС.2009.263>.

- [3] Lops, P. Content-based Recommender Systems: State of the Art and Trends / P.Lops, M. de Gemmis, G.Semeraro // Recommender Systems Handbook. — Boston, MA : Springer, 2011. — P. 73–105. — DOI: https://doi.org/10.1007/978-0-387-85820-3_3.

Библиографическая ссылка

Свистунов, А.Н. Система рекомендаций для медиаконтента // Студенческая конференция факультета ПМИК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 180–188.

Сведения об авторах

СВИСТУНОВ АЛЕКСЕЙ НИКОЛАЕВИЧ
Студент магистратуры
Направление «Прикладная информатика»

УДК 004.85, 004.93

Сравнение функций потерь Triplet Loss и ArcFace для задачи One-Shot распознавания лиц

Терешкин М. В.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий И. С. Солдатенко

Аннотация. В работе исследуется задача идентификации личности по изображению лица в условиях крайне ограниченного числа эталонных снимков (*one-shot learning*). На базе датасета Olivetti (40 человек) реализованы и экспериментально сравнены два подхода: Triplet Loss и функция потерь с угловым отступом ArcFace. Обучение проводилось на одном случайном изображении каждого человека с применением агрессивных аугментаций. Лучший результат достигнут моделью с ArcFace: точность идентификации по принципу ближайшего соседа составила 63,6 %, что в 25 раз выше случайного уровня.

Ключевые слова: распознавание лиц, *one-shot learning*, ArcFace, Triplet Loss, ограниченная выборка.

Введение

Системы биометрической идентификации личности широко применяются в пропускных системах, криминалистике, мобильных сервисах. Однако на практике часто доступно лишь одно или несколько эталонных изображений каждого человека, что делает невозможным применение классических глубоких нейросетей, требующих тысяч примеров на класс. Задача распознавания при наличии всего одного обучающего снимка на класс известна как *one-shot learning* и вызывает активный интерес исследователей [2, 3]. Современные свёрточные архитектуры, включая ResNet, позволили добиться прогресса в этой области, в том числе благодаря тренду на обучение с малым числом примеров [1].

Целью настоящей работы является экспериментальное сравнение двух подходов — метрического обучения Triplet Loss и функции потерь с аддитивным угловым отступом ArcFace — в условиях одного эталонного фото на человека. Особое внимание уделяется качеству формируемых эмбедингов и точности идентификации по ближайшему соседу.

1. Методика экспериментов

Данные и аугментация. Эксперименты проводились на открытом датасете Olivetti Faces, содержащем 40 человек по 10 черно-белых изображений размером 92×112 пикселей. Для имитации one-shot режима было выполнено случайное разбиение: 1 изображение каждого человека помещалось в обучающую выборку, оставшиеся 9 — в тестовую. Таким образом, обучающий набор содержал всего 40 снимков.

Чтобы компенсировать нехватку данных, применялся широкий спектр аугментаций на этапе обучения:

- случайное кадрирование (RandomResizedCrop) с масштабом от 0,6 до 1,0;
- горизонтальное отражение (RandomHorizontalFlip);
- изменение яркости, контрастности, насыщенности и оттенка (ColorJitter);
- случайный поворот до $\pm 20^\circ$;
- гауссово размытие (GaussianBlur);
- случайное стирание участков изображения (RandomErasing).

Все изображения приводились к размеру 224×224 и нормализовались средними значениями $[0,485, 0,456, 0,406]$ и стандартными отклонениями $[0,229, 0,224, 0,225]$, принятыми для моделей, предобученных на ImageNet.

Архитектура модели. В качестве базового энкодера использовалась предобученная на ImageNet сеть ResNet50 [1], к которой добавлен проекционный слой, преобразующий выход сети в 512-мерный

вектор признаков с последующей L2-нормализацией. Размораживались только параметры последнего свёрточного блока (layer4), что позволяло адаптировать высокоуровневые признаки к конкретным лицам, сохраняя низкоуровневые свойства, унаследованные от ImageNet.

Функции потерь. Исследовались два варианта функции потерь.

- 1) **Triplet Loss** [3] обучает модель минимизировать евклидово расстояние между эмбедами одного человека и максимизировать расстояние до эмбеддингов других людей. Использовался *margin* $m = 0,5$, тройки (anchor, positive, negative) формировались случайным образом внутри батча.
- 2) **ArcFace** [2] добавляет угловой отступ в классификационный слой, трансформируя логиты как $\cos(\theta + m)$ для правильного класса. Это приводит к более компактным и хорошо разделимым кластерам в пространстве эмбеддингов. Параметры: масштаб $s = 64,0$, отступ $m = 0,4$.

Обучение. Модель обучалась 160 эпох с оптимизатором AdamW (learning rate 10^{-4} , weight decay $5 \cdot 10^{-4}$) и косинусным планировщиком скорости обучения, уменьшающим её до 10^{-6} . Применялась техника смешанной точности (AMP) для ускорения вычислений на GPU (NVIDIA RTX 4070 Ti), а также градиентный клипсинг с максимальной нормой 1,0 для предотвращения взрывов градиента.

Оценка. Для оценки качества идентификации применялся протокол *one-shot распознавания*:

- вычислялись эталонные эмбединги для всех 40 обучающих изображений;
- для каждого тестового снимка находился ближайший сосед по косинусному сходству среди эталонов;
- предсказанием считался класс ближайшего соседа.

2. Результаты

В табл. 1 приведены итоговые метрики для обоих подходов. Модель с ArcFace продемонстрировала значительно более высокую точность идентификации.

Таблица 1. Сравнение подходов на Olivetti (точность one-shot, макро-усреднённые точность и полнота).

Метод	Точность	Прес. (macro)	Рес. (macro)
Triplet Loss	39,7 %	0,52	0,40
ArcFace (160 эпох)	63,6 %	0,72	0,64

Анализ матрицы ошибок показал, что для 10 из 40 классов модель с ArcFace обеспечивает практически безошибочную идентификацию (F1-мера $> 0,8$). В то же время выявлены классы с нулевой точностью, что объясняется низким качеством или нехарактерным ракурсом единственного эталонного снимка. Это подчёркивает критическую важность выбора качественного эталонного изображения для практических систем.

Дополнительно стоит отметить, что ArcFace продемонстрировал устойчивый рост точности на валидации даже после того, как точность на обучении достигла 100 %, что свидетельствует о хорошей обобщающей способности и отсутствии переобучения.

Заключение

В работе исследована задача идентификации личности в условиях одного эталонного снимка. Экспериментально подтверждено, что функция потерь ArcFace превосходит Triplet Loss по стабильности сходимости и итоговой точности. Достигнута one-shot точность 63,6 % на датасете Olivetti, что более чем в 25 раз превышает случайный уровень (2,5 %).

Дальнейшие исследования направлены на масштабирование подхода на датасет CASIA-WebFace ($> 10\,000$ классов), а также на замену энкодера на более мощные архитектуры (ResNet100, EfficientNet) с целью повышения точности в условиях большого числа классов.

Список литературы

- [1] Кочетугов, В. В. Свёрточные нейронные сети в системах обнаружения и распознавания лиц / В. В. Кочетугов, О. И. Федяев //

- Информатика и кибернетика. — 2025. — №1. — С. 26–32. — URL: <https://ojs.donntu.ru/infcyb/article/view/571>.
- [2] ArcFace: Additive Angular Margin Loss for Deep Face Recognition / J. Deng, J. Guo, N. Xue, S. Zafeiriou // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA : IEEE Computer Society, 2019. — P. 4690–4699. — DOI: <https://doi.org/10.1109/CVPR.2019.00482>.
- [3] Holkar, A. Few-shot learning for face recognition in the presence of image discrepancies for limited multi-class datasets / A. Holkar, R. Walambe, K. Kotecha // Image and Vision Computing. — 2022. — Vol. 120. — P. 104420. — DOI: <https://doi.org/10.1016/j.imavis.2022.104420>.

Библиографическая ссылка

Терешкин, М. В. Сравнение функций потерь Triplet Loss и ArcFace для задачи One-Shot распознавания лиц // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 189–193.

Сведения об авторах

ТЕРЕШКИН МИХАИЛ ВЛАДИМИРОВИЧ

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 004.85, 631.559

Разработка программно-интеллектуальной системы прогнозирования урожайности сельскохозяйственных культур с применением алгоритма случайного леса

Тимошенко Д. В.

Тверской государственной университет

Рекомендовано к публикации доцентом кафедры математической статистики и системного анализа С. В. Архиповым

Аннотация. В статье рассматривается задача прогнозирования урожайности сельскохозяйственных культур на основе агроэкологических данных полигона «Губино». Разработана программно-интеллектуальная система в среде R Shiny, обеспечивающая автоматизацию обучения, калибровки и применения моделей машинного обучения. Система позволяет выполнять обработку данных и формировать прогнозные оценки урожайности. Для построения прогнозов использован алгоритм случайного леса (Random Forest).
Ключевые слова: машинное обучение, случайный лес, прогнозирование урожайности, R Shiny, агроэкологические данные.

Введение

Прогнозирование урожайности сельскохозяйственных культур является фундаментальной задачей агрономии, напрямую связанной с продовольственной безопасностью и эффективностью управления земельными ресурсами. Сложность данной задачи обусловлена многофакторностью объекта исследования: итоговая продуктивность (Y) представляет собой сложную функцию от генетического потенциала культур, экологических факторов (климат, рельеф), свойств почвы и агротехнических мероприятий.

Традиционные статистические методы, такие как множественная линейная регрессия, часто оказываются неэффективными при работе с агроэкологическими данными. Это связано с наличием сильных

нелинейных связей между факторами и явлением мультиколлинеарности (например, корреляция высоты рельефа с температурой и влажностью), что ведёт к нестабильности классических моделей.

Применение методов машинного обучения, в частности алгоритма случайного леса (Random Forest) [2], позволяет моделировать эти сложные зависимости без жёстких априорных допущений о распределении данных. Благодаря механизму бэггинга и случайному выбору подмножества признаков, данный алгоритм обеспечивает высокую робастность к зашумлённым данным и выбросам, что критически важно при анализе результатов многолетних полевых экспериментов.

Актуальность данной работы заключается в необходимости автоматизации процесса построения таких моделей для оперативного принятия решений в агрономии. Целью исследования является разработка программного комплекса в среде R Shiny, позволяющего проводить предиктивный анализ урожайности на основе ландшафтных и микроклиматических показателей агрополигона «Губино».

1. Объект исследования и система предикторов

Экспериментальной базой исследования послужили данные многолетнего мониторинга (1998–2024 гг.) на агрополигоне «Губино» (Тверская область). Объектом прогнозирования выступает урожайность смеси многолетних трав (клевер луговой и тимофеевка луговая). Для построения модели была сформирована система независимых переменных, разделённая на три иерархических блока:

- морфометрические параметры рельефа: высота местности над уровнем моря (Height), крутизна склонов (Steepness), кривизна поверхности (Curvature);
- микроклиматические показатели: солнечная радиация (Radiation);
- почвенно-агрохимические характеристики: активная кислотность почвы (pH), содержание подвижных форм фосфора (P) и калия (K), содержание гумуса (Humus).

Специфика данных заключается в их высокой пространственной вариабельности: коэффициент вариации урожайности достигает 34,7 %

в неблагоприятные годы. Важным этапом подготовки данных стала классификация вегетационных периодов за весь срок наблюдений (1998–2024 гг.). С помощью иерархического кластерного анализа были выделены четыре типа лет, различающихся по совокупности гидротермических условий. Данная классификация позволила разделить гетерогенную выборку на более однородные группы, что является необходимым условием для корректного обучения ансамблевых алгоритмов. Для каждого кластера была сформирована специализированная модель, адаптированная к специфике конкретного типа вегетационного периода.

2. Методика обучения и калибровки предиктивных моделей

В основу вычислительного ядра системы положена оптимизированная реализация алгоритма Random Forest (библиотека `ranger`), обеспечивающая высокую скорость обработки больших массивов агроэкологических данных. Особенностью предложенного подхода является автоматизированный цикл «обучение – калибровка», позволяющий минимизировать риски переобучения модели. Процесс подготовки прогностического модуля включает следующие этапы.

- 1) Формирование обучающей и контрольной выборок. В интерфейсе реализован механизм динамического разделения данных (Train/Test Split). Это позволяет оценивать прогностическую способность алгоритма на данных, которые не участвовали в построении деревьев.
- 2) Настройка ансамбля. Алгоритм генерирует совокупность декоррелированных решающих деревьев. Итоговый прогноз урожайности (\hat{y}) вычисляется как среднее значение предсказаний всех деревьев ансамбля:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^B T_b(x),$$

где B — количество деревьев, $T_b(x)$ — прогноз отдельного дерева.

- 3) Оценка точности (калибровка). Для верификации модели в системе рассчитывается комплекс метрик [1]: средняя квадратичная ошибка (RMSE), коэффициент детерминации (R^2) и средняя абсолютная ошибка в процентах (MAPE). Использование MAPE особенно актуально для агрономии, так как позволяет интерпретировать погрешность в терминах отклонения от фактического сбора урожая.

Важной практической составляющей является возможность сохранения метаданных модели. Каждая обученная модель жёстко связывается со списком предикторов и целевой переменной, что исключает ошибки при последующем мультимодельном прогнозировании. Внедрение этапа калибровки непосредственно в модуль обучения позволяет пользователю отсеивать неэффективные конфигурации факторов до их сохранения в библиотеку моделей.

3. Архитектура и функциональные возможности программного комплекса

Разработанный программный комплекс на базе фреймворка Shiny реализует модульную архитектуру, обеспечивающую полный цикл работы с предиктивными моделями: от предварительной обработки данных до формирования отчётности. В отличие от существующих узкоспециализированных решений, данная система позволяет интегрировать экспертные знания агронома в процесс машинного обучения.

Логическая структура приложения включает четыре функциональных блока.

- 1) Модуль обучения и калибровки. Пользователь осуществляет загрузку экспериментальных данных, настраивает параметры алгоритма Random Forest (через библиотеку ranger) и определяет состав предикторов. Важной особенностью является этап калибровки: система автоматически оценивает качество обученных моделей на отложенной тестовой выборке, позволяя оперативно корректировать состав переменных. После завершения процесса модель сохраняется в сериализованном формате вместе с метаданными.

- 2) Система управления библиотекой моделей. В данном блоке реализован реестр всех созданных моделей. Для каждой записи доступна детальная информация: список используемых входных переменных, целевой показатель урожайности и статистические характеристики. Предусмотрена возможность добавления текстовых примечаний («заметок»), что позволяет документировать специфические условия эксперимента или особенности конкретного года обучения.
- 3) Модуль мультимодельного прогнозирования. Этот раздел предназначен для работы с новыми данными. Система поддерживает одновременную работу с несколькими моделями, что позволяет проводить сравнительный анализ прогнозов. Если загружаемый файл содержит фактические значения урожайности, программный комплекс автоматически вычисляет метрики точности (RMSE, MAPE) и строит диагностические графики (диаграммы рассеяния). Результаты сессии прогнозирования могут быть моментально сохранены в архив.
- 4) Архив прогнозов и экспорт данных. Финальный модуль обеспечивает хранение истории предиктивного анализа. Пользователь может просмотреть результаты прошлых расчётов, визуализировать графики сравнения моделей и выполнить экспорт данных в формат CSV для последующей обработки в сторонних ГИС-системах или статистических пакетах.

Техническая реализация опирается на использование библиотеки `bslib` для создания современного интерфейса и `thematic` для адаптивной визуализации графиков `ggplot2` [3]. Реактивное управление состоянием в Shiny гарантирует консистентность данных при переключении между модулями.

4. Анализ эффективности обученных моделей

Для каждого выделенного климатического кластера в разработанном программном комплексе была обучена индивидуальная модель `Random Forest`. Проверка предсказательной способности проводилась на отложенных выборках (`Test Split`), а внутренняя стабильность оценивалась через ошибку «`Out-of-Bag`» (ООБ). Результаты оценки качества представлены в таблице 1.

Таблица 1. Метрики качества моделей по климатическим кластерам.

Метрика	R^2	MAPE, %	RMSE	OOB Error
Кластер 1	0,929	4,89	0,222	0,3167
Кластер 2	0,923	5,84	0,241	0,3760
Кластер 3	0,951	6,73	0,232	0,2940
Кластер 4	0,902	6,96	0,285	0,3085

Анализ таблицы показывает, что наиболее высокую точность продемонстрировала модель для Кластера 3 ($R^2 = 0,951$). Высокие показатели метрик во всех группах объясняются тем, что предварительная кластеризация данных позволила алгоритму Random Forest эффективнее выявлять закономерности внутри статистически близких погодных сценариев. Ошибка MAPE для всех моделей находится в диапазоне 4,89–6,96 %, что подтверждает высокую прогностическую способность разработанного программного комплекса и адекватность выбранного метода разбиения данных.

Заключение

В ходе работы был разработан программно-методический комплекс на базе фреймворка R Shiny, обеспечивающий автоматизацию полного цикла предиктивного анализа урожайности. Внедрение модулей управления жизненным циклом моделей и архивации прогнозов позволяет трансформировать разрозненные расчётные скрипты в полноценную систему поддержки принятия решений.

Дальнейшее развитие программного комплекса планируется в направлении расширения библиотеки доступных алгоритмов машинного обучения (в частности, интеграции методов градиентного бустинга) для проведения сравнительного анализа эффективности ансамблей. Также перспективным вектором является разработка модуля пространственной визуализации данных, который позволит сопоставлять результаты прогноза с электронными картами полей, обеспечивая более наглядную интерпретацию результатов для конечного пользователя-агронома.

Список литературы

- [1] Гмурман, В.Е. Теория вероятностей и математическая статистика : учебник для вузов / В.Е.Гмурман. — 12-е изд. — Москва : Издательство Юрайт, 2026. — 479 с. — URL: <https://urait.ru/bcode/598377>.
- [2] Breiman, L. Random Forests / L. Breiman // Machine Learning. — 2001. — Vol. 45. — P. 5–32. — DOI: <https://doi.org/10.1023/A:1010933404324>.
- [3] Wickham, H. ggplot2: Elegant Graphics for Data Analysis / H. Wickham. — 2nd ed. — Cham : Springer, 2016. — 260 p. — DOI: <https://doi.org/10.1007/978-3-319-24277-4>.

Библиографическая ссылка

Тимошенко, Д. В. Разработка программно-интеллектуальной системы прогнозирования урожайности сельскохозяйственных культур с применением алгоритма случайного леса // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 194–200.

Сведения об авторах

ТИМОШЕНКО ДЕНИС ВИКТОРОВИЧ
Студент магистратуры
Направление «Прикладная информатика»

УДК 004.85, 004.932

Влияние индуктивных смещений архитектур ResNet, Vision Transformer и ConvNeXt на обобщающую способность моделей классификации изображений

Тонмее Е. Л.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий Р. Н. Гордеевым

АННОТАЦИЯ. Сопоставляется влияние индуктивных смещений ResNet-50, ViT-S/16 и ConvNeXt-T на обобщающую способность моделей классификации изображений. По данным, систематизированным в диссертации, анализируются CIFAR-100 и CIFAR-100-C. Показано, что свёрточные модели эффективнее при дефиците данных, тогда как ViT сокращает отставание при росте выборки.
КЛЮЧЕВЫЕ СЛОВА: глубокое обучение, классификация изображений, индуктивные смещения, ResNet, Vision Transformer, ConvNeXt, обобщающая способность.

Введение

Практическая значимость задачи классификации изображений определяется тем, что модели компьютерного зрения всё чаще работают в условиях ограниченных обучающих выборок и изменчивого качества входных данных. В таких условиях выбор архитектуры нельзя сводить только к числу параметров или вычислительной сложности. Существенную роль играют индуктивные смещения — априорные предположения о структуре изображения, заложенные в модель до начала обучения.

Свёрточные архитектуры используют локальность, разделение весов и трансляционную эквивариантность как встроенные ограничения. Поэтому они обычно эффективнее используют малые и средние выборки. Vision Transformer (ViT), напротив, переносит

изображение в последовательностное представление и в меньшей степени опирается на заранее заданную пространственную структуру [2]. Это повышает гибкость модели, но увеличивает зависимость от объёма данных.

Цель статьи — на основе результатов, систематизированных в магистерской диссертации, сопоставить архитектурные семейства ResNet-50, ViT-S/16 и ConvNeXt-T с точки зрения обобщающей способности, эффективности использования обучающих данных и устойчивости к типовым искажениям.

1. Архитектуры и индуктивные смещения

ResNet-50 относится к классическим свёрточным архитектурам. Её ключевая идея — остаточное обучение: блок сети аппроксимирует не целевое отображение напрямую, а остаточную функцию, которая складывается с входом блока. Такая схема облегчает оптимизацию глубоких сетей и стала одной из причин успеха ResNet в задачах распознавания изображений [3]. С точки зрения индуктивных смещений ResNet-50 наследует основные свойства CNN: локальные рецептивные поля, повторное использование одних и тех же фильтров в разных областях изображения и устойчивую обработку сдвигов.

ViT-S/16 строит представление иначе. Изображение разбивается на патчи, каждый патч переводится в вектор, после чего последовательность векторов обрабатывается трансформером. Механизм самовнимания позволяет учитывать глобальные связи между фрагментами уже на ранних слоях. Однако локальность и трансляционная эквивариантность не заданы так жёстко, как в свёрточных сетях. Пространственные отношения должны быть извлечены из данных через позиционные кодирования и обучение параметров модели [2].

ConvNeXt-T занимает промежуточную позицию. Это свёрточная архитектура, модернизированная с учётом решений, характерных для эпохи Vision Transformer и Swin Transformer: используются изменённый входной блок, крупные depthwise-свёртки, LayerNorm, GELU и инвертированная структура блока. При этом ConvNeXt остаётся чистой ConvNet-моделью и сохраняет свёрточные индуктивные смещения [1].

Табл. 1 показывает содержательный ряд архитектур: от классической CNN с выраженными пространственными предположениями

Таблица 1. Сравнение архитектурных семейств.

Модель	Основа	Смещения	Малые данные
ResNet-50	свёртки	сильные	устойчиво
ViT-S/16	внимание	слабые	требует больше
ConvNeXt-T	модерн. CNN	сильные	лучшее поведение

через модель с минимальными пространственными ограничениями к современной ConvNet, которая сохраняет преимущества свёрток, но использует более поздние архитектурные решения.

2. Методика сравнительного анализа

Сравнение выполняется по опубликованным результатам и данным, систематизированным в диссертации. Поэтому далее используются формулировки «сопоставление показывает» и «анализ опубликованных данных позволяет предположить», а не утверждение о полном новом обучении всех моделей.

Основой анализа обобщающей способности служит CIFAR-100 — набор из 100 классов изображений, удобный для оценки качества на чистой тестовой выборке. Центральная метрика — Тор-1 Accuracy:

$$\text{Acc@1} = \frac{N_{correct}}{N_{total}} \cdot 100 \%.$$

Здесь $N_{correct}$ — число правильно классифицированных изображений, а N_{total} — объём тестовой выборки. Дополнительно учитывается Generalization Gap как разность между качеством на обучающей и тестовой выборках: большой разрыв указывает на переобучение.

Для анализа устойчивости к данным вне исходного распределения используется CIFAR-100-C — вариант CIFAR-100 с типовыми искажениями: шумом, размытием, погодными и цифровыми нарушениями. Подход Hendrycks и Dietterich позволяет оценивать качество моделей на common corruptions, а показатель mCE агрегирует ошибку по типам искажений [4]:

$$\text{mCE} = \frac{1}{15} \sum_{c=1}^{15} CE_c.$$

Таблица 2. Top-1 Accuracy на CIFAR-100, %.

Доля	ResNet-50	ViT-S/16	ConvNeXt-T
10 %	$55,2 \pm 0,8$	$43,1 \pm 1,4$	$57,6 \pm 0,7$
20 %	$63,8 \pm 0,6$	$55,6 \pm 1,1$	$65,9 \pm 0,5$
50 %	$71,4 \pm 0,5$	$68,9 \pm 0,7$	$73,2 \pm 0,4$
100 %	$76,3 \pm 0,4$	$75,8 \pm 0,5$	$78,1 \pm 0,3$

В настоящей статье устойчивость обсуждается кратко: CIFAR-100-S важен как индикатор поведения вне исходного распределения, но полная матрица искажений не включается из-за ограниченного формата статьи.

3. Результаты и обсуждение

В табл. 2 приведены значения Top-1 Accuracy для четырёх долей обучающей выборки CIFAR-100. Значения отражают среднее и стандартное отклонение по данным, систематизированным в диссертации.

При 10 % обучающих данных свёрточные архитектуры имеют явное преимущество: ResNet-50 достигает 55,2 %, ConvNeXt-T — 57,6 %, тогда как ViT-S/16 показывает 43,1 %. Такое поведение согласуется с представлением о «data-hungry» характере ViT: закономерности, которые в CNN заданы архитектурно, трансформер должен восстанавливать из обучающих примеров.

При увеличении объёма данных разрыв сокращается. На 50 % выборки ViT-S/16 приближается к ResNet-50, а при полном объёме CIFAR-100 различие между ними становится небольшим: 76,3 % против 75,8 %. Согласно выводам диссертации, для этой пары при 100 % данных различие статистически незначимо ($p = 0,2035$). Поэтому корректнее говорить не о безусловном превосходстве одной архитектуры, а о зависимости результата от объёма обучающих данных.

ConvNeXt-T превосходит ResNet-50 на всех уровнях выборки. Это важно методологически: выигрыш ConvNeXt нельзя свести только к наличию свёрточных индуктивных смещений. Он указывает на самостоятельную ценность архитектурных улучшений: свёрток 7×7 ,

иной нормализации, изменённой структуры блока и современной схемы построения ConvNet [1].

Устойчивость к искажениям требует осторожной интерпретации. CIFAR-100-C и mCE позволяют оценить, как модели ведут себя при нарушении исходного распределения данных [4]. Анализ диссертации указывает, что свёрточные модели чувствительны к шумам, разрушающим локальные текстуры, тогда как ViT может сильнее страдать от искажений, нарушающих пространственную когерентность патчей. ConvNeXt-T в этом сопоставлении занимает промежуточное положение: расширенное рецептивное поле помогает учитывать более широкий контекст, но модель сохраняет свёрточную природу.

Заключение

Сравнительный анализ показывает, что различие между ResNet-50, ViT-S/16 и ConvNeXt-T связано не только с размером моделей, но и с характером индуктивных смещений. ResNet-50 и ConvNeXt-T используют локальность и разделение весов, поэтому оказываются предпочтительнее при ограниченном объёме обучающих данных. ViT-S/16 обладает более слабыми пространственными смещениями и сильнее зависит от числа примеров, но при росте выборки приближается к ResNet-50.

Практический вывод состоит в следующем: при дефиците данных разумнее выбирать ConvNeXt-T или ResNet-50, а при большем объёме обучающей выборки ViT-S/16 становится конкурентоспособным. ConvNeXt-T выглядит особенно сильным компромиссом, поскольку сохраняет преимущества CNN и одновременно использует архитектурные решения, характерные для трансформерной эпохи.

Ограничение анализа состоит в его опоре на опубликованные результаты и данные, систематизированные в диссертации. Для окончательной проверки численных выводов необходимы независимые воспроизводимые эксперименты с единым кодом обучения, одинаковыми режимами аугментации и открытыми логами запусков.

Список литературы

- [1] A ConvNet for the 2020s / Z.Liu, H.Mao, C.-Y. Wu [et al.] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA : IEEE Computer Society, 2022. — P. 11976–11986. — DOI: <https://doi.org/10.1109/CVPR52688.2022.01167>.
- [2] An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale / A.Dosovitskiy, L.Beyer, A.Kolesnikov [et al.] // 9th International Conference on Learning Representations. — ICLR, 2021. — DOI: <https://doi.org/10.48550/arXiv.2010.11929>.
- [3] Deep Residual Learning for Image Recognition / K.He, X.Zhang, S.Ren, J.Sun // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — Los Alamitos, CA : IEEE Computer Society, 2016. — P. 770–778. — DOI: <https://doi.org/10.1109/CVPR.2016.90>.
- [4] Hendrycks, D. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations / D.Hendrycks, T.Dietterich // 7th International Conference on Learning Representations. — ICLR, 2019. — DOI: <https://doi.org/10.48550/arXiv.1903.12261>.

Библиографическая ссылка

Тонмеу, Е. Л. Влияние индуктивных смещений архитектур ResNet, Vision Transformer и ConvNeXt на обобщающую способность моделей классификации изображений // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 201–206.

Сведения об авторах

ТОНМЕУ ЕЛВАДАС ЛОРИС

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 519.766

Выразительные возможности классов категориальных грамматик

Федоров И. В.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры информатики

Б. Н. Карловым

Аннотация. В работе исследуются выразительные возможности классов категориальных грамматик. Рассматриваются $(*, *)$ -жёсткие категориальные грамматики и их связь с КС-грамматиками. Доказывается, что для любой однозначной контекстно-свободной грамматики существует $(*, *)$ -жёсткая категориальная грамматика, порождающая тот же язык.

Ключевые слова: классические категориальные грамматики, КС-грамматики, нормальная форма Грейбах, $(*, *)$ -жёсткие грамматики.

Введение

Изучение формальных грамматик и языков является одной из фундаментальных областей теоретической информатики и компьютерной лингвистики, закладывающей основы для широкого круга практических применений. Формальные грамматики служат математической моделью синтаксиса языков программирования и естественных языков, что делает их неотъемлемой частью компиляторов, интерпретаторов и систем автоматического разбора текстов. Разработка более выразительных классов грамматик и эффективных алгоритмов их обработки напрямую влияет на производительность современных средств разработки ПО и на качество систем машинного перевода.

Одним из классов категориальных грамматик являются классические категориальные грамматики [1]. Классические категориальные грамматики, в которых категория, приписываемая каждому символу, определяется этим символом и его m левыми и n правыми

соседями, — это (m, n) -жёсткие грамматики, они определены в статье [3]. Также в этой статье определяется понятие $(*, *)$ -жёсткой грамматики, в которой контекст слова позволяет определить его синтаксическую роль в предложении. Символ $*$ означает, что при определении категории учитываются все символы, стоящие по заданную сторону от текущего символа. Данное понятие аналогично понятию однозначной КС-грамматики, а потому теоретический интерес представляет исследование связи языков, порождаемых этими двумя типами грамматик.

1. Основные определения

ОПРЕДЕЛЕНИЕ 1. Пусть \mathbf{C} — произвольное конечное множество (множество элементарных категорий), $(,), /, \backslash$ — символы, не принадлежащие \mathbf{C} . Множество категорий над \mathbf{C} (обозначается $\text{Cat}(\mathbf{C})$) — это наименьшее множество слов в алфавите $\mathbf{C} \cup \{ (,), /, \backslash \}$ такое, что:

- 1) $\mathbf{C} \subseteq \text{Cat}(\mathbf{C})$;
- 2) если $\Phi, \Psi \in \text{Cat}(\mathbf{C})$, то $(\Phi/\Psi) \in \text{Cat}(\mathbf{C})$ и $(\Psi \backslash \Phi) \in \text{Cat}(\mathbf{C})$.

Элементы множества $\text{Cat}(\mathbf{C})$ называются категориями.

Пусть A и B_1, \dots, B_k — элементарные категории. Будем записывать категорию $((A/B_k)/\dots)/B_1$ в виде сокращенной записи без внутренних скобок — $(A/B_k/\dots/B_1)$.

Обозначим через $\text{Cat}(\mathbf{C})^*$ множество всех конечных последовательностей (или строк) категорий. На множестве $\text{Cat}(\mathbf{C})^*$ определяется отношение сокращения \vdash .

ОПРЕДЕЛЕНИЕ 2. Для любых двух категорий Φ и Ψ и любых двух строк категорий Γ_1 и Γ_2 имеет место $\Gamma_1(\Phi/\Psi)\Psi\Gamma_2 \vdash \Gamma_1\Phi\Gamma_2$ и $\Gamma_1\Psi(\Psi\backslash\Phi)\Gamma_2 \vdash \Gamma_1\Phi\Gamma_2$. Через \vdash^* обозначается рефлексивное транзитивное замыкание отношения \vdash .

ОПРЕДЕЛЕНИЕ 3. Классическая категориальная грамматика — это четвёрка $G = (\Sigma, \mathbf{C}, S, \delta)$, где:

- Σ — конечный алфавит,
- \mathbf{C} — конечное множество элементарных категорий,
- $S \in \text{Cat}(\mathbf{C})$ — главная категория,

- $\delta: \Sigma \rightarrow \mathcal{P}(\text{Cat}(\mathbf{C}))$ — отображение алфавита в множество всех подмножеств множества категорий такое, что для любого $a \in \Sigma$ множество $\delta(a)$ конечно (словарь).

Для слова $w = a_1 \dots a_n \in \Sigma^*$ через $\delta(w)$ будем обозначать множество всех последовательностей категорий, которые могут быть приписаны слову w :

$$\delta(w) = \{\Phi_1 \dots \Phi_n \mid \Phi_i \in \delta(a_i) \text{ для } i = 1, \dots, n\}.$$

ОПРЕДЕЛЕНИЕ 4. Язык, порождаемый грамматикой $G = (\Sigma, \mathbf{C}, S, \delta)$, состоит из всех слов $a_1 \dots a_n \in \Sigma^+$, для которых существует последовательность категорий $\Phi_1 \dots \Phi_n$ такая, что $\Phi_i \in \delta(a_i)$ для всех i и $\Phi_1 \dots \Phi_n \vdash^* S$. Язык, порождаемый грамматикой G , обозначается $L(G)$.

ОПРЕДЕЛЕНИЕ 5. Категориальная грамматика $G = \langle \Sigma, \mathbf{C}, S, \delta \rangle$ называется $(*, *)$ -жёсткой, если для любого слова $u = w_1 a w_2$, для любых последовательностей категорий $\Gamma_1, \Gamma'_1 \in \delta(w_1), \Gamma_2, \Gamma'_2 \in \delta(w_2)$ и для любых двух категорий $\Phi, \Psi \in \delta(a)$ из $\Gamma_1 \Phi \Gamma_2 \vdash^* S$ и $\Gamma'_1 \Psi \Gamma'_2 \vdash^* S$ следует $\Phi = \Psi$.

Напомним также определение нормальной формы Грейбах.

ОПРЕДЕЛЕНИЕ 6. КС-грамматика $G = \langle N, \Sigma, P, S \rangle$ называется КС-грамматикой в нормальной форме Грейбах, если G — грамматика без ε -правил и все её правила имеют один из следующих видов:

- 1) $S \rightarrow \varepsilon$;
- 2) $A \rightarrow a\alpha$, где $A \in N, a \in \Sigma, \alpha \in N^*$.

2. $(*, *)$ -жёсткость однозначных КС-языков

Нашим основным результатом является следующее утверждение.

ТЕОРЕМА 1. Пусть G — однозначная КС-грамматика, тогда существует $(*, *)$ -жёсткая грамматика G' такая, что $L(G) = L(G')$.

Можно ограничиться рассмотрением КС-грамматик без бесполезных нетерминалов, поскольку при их удалении сохраняется однозначность грамматики. Известно также, что по однозначной КС-грамматике можно построить однозначную КС-грамматику в нормальной форме Грейбах, используя преобразования, описанные в статье [2].

Пусть G'' — однозначная КС-грамматика в нормальной форме Грейбах. По G'' можно построить эквивалентную ей категориальную грамматику G' следующим образом: если для a в G'' существует правило $A \rightarrow aB_1 \dots B_k$, то этому a сопоставляется категория $(A/B_k / \dots / B_1)$ [1].

Покажем сначала, что для строки категорий, содержащей только правые деления, дерево сокращений будет единственным.

ЛЕММА 2. Пусть Γ — строка категорий, в которой все категории имеют вид $(A/B_1 / \dots / B_r)$, где $A, B_i \in \mathcal{C}$. Тогда дерево сокращений для Γ будет единственным.

ДОКАЗАТЕЛЬСТВО. Так как мы ограничиваемся только правым делением, то некоторая категория $(A/B_1 / \dots / B_r)$ последовательно сокращается с примитивными категориями B_r, \dots, B_1 , например, $(A/B/C)CB \vdash (A/B)V \vdash A$. Это единственно возможный способ сокращений, так как левых делений нет. Следовательно, не существует альтернативы применения правила сокращения для некоторых (A/B) и B , стоящего правее. Таким образом, существует не более одного способа сократить Γ при правостороннем делении. \square

Далее определим глубину категории и докажем лемму о единственности строк категорий, сокращающихся до категории γ для непустого слова w .

ОПРЕДЕЛЕНИЕ 7. Пусть Γ — строка категорий такая, что $\Gamma \vdash^* S$. Глубина категории в строке Γ определяется следующим образом.

- 1) Если $\Gamma = \gamma\Gamma'$, где $\gamma = (S / \dots)$, то γ имеет глубину 0.
- 2) Пусть

$$\Gamma = \dots (A/B_1 / \dots / B_k) (B_k / \dots) \dots (B_{k-1} / \dots) \dots (B_1 / \dots)$$

является строкой категорий, где категории (B_j / \dots) сокращаются с $(A/B_1 / \dots / B_k)$. Тогда если $(A/B_1 / \dots / B_k)$ является категорией глубины i , то категории $(B_k / \dots), (B_{k-1} / \dots), \dots, (B_1 / \dots)$ имеют глубину $i + 1$.

Это определение корректно, так как в данном случае в Γ есть только правые деления, а по лемме 2 строка категорий, в которой все категории имеют вид $(A/B_1 / \dots / B_r)$, сокращается единственным способом.

ЛЕММА 3. Если $\Gamma_1, \Gamma_2 \in \delta(w)$, $\Gamma_1, \Gamma_2 \vdash^* S$, то для любого i в Γ_1 и в Γ_2 совпадают позиции категорий глубины i и сами категории.

ДОКАЗАТЕЛЬСТВО. Докажем индукцией по глубине категории.

Базис. Пусть $i = 0$. Пусть $\Gamma_1 = \gamma_1 \gamma_2 \dots \gamma_n$, при этом $\Gamma_1 \vdash^* S$ и $\gamma_1 = (S/A_k / \dots / A_1)$. Тогда существуют строки категорий $\Gamma'_1, \dots, \Gamma'_k$ такие, что $\Gamma'_i \vdash^* A_i$ и $\Gamma'_1 \dots \Gamma'_k = \gamma_2 \dots \gamma_n$. Категория γ_1 и только она имеет глубину 0. Предположим, что первая категория строки Γ_2 отлична от γ_1 . Поскольку в категориях нет левых делений, то она имеет вид $(S/B_m / \dots / B_1)$ для некоторых B_j . Тогда для слова w существуют деревья вывода в КС-грамматике G'' , в которых сыновья корня помечены символами a_1, A_1, \dots, A_k и a_1, B_1, \dots, B_m соответственно. Но такие деревья различны, что противоречит однозначности грамматики G'' . Таким образом, категории глубины 0 должны совпадать.

Индукционный шаг. Допустим, что для i утверждение доказано. Тогда возьмём строку категорий

$$\gamma_u \gamma_{u+1} \dots \gamma_v \vdash^* (A/B_r / \dots / B_1) B_1 \dots B_r \vdash^* A,$$

где $\gamma_u = (A/B_r / \dots / B_1)$ имеет глубину i . По предположению индукции γ_u имеет одну и ту же позицию в обеих строках категорий Γ_1 и Γ_2 и может быть определена только единственным образом. Существует r категорий $\gamma^1, \gamma^2, \dots, \gamma^r$ глубины $i + 1$ таких, что

$$\gamma^1 \Gamma^1 = \gamma_{u+1} \Gamma^1 \vdash^* B_1,$$

$$\gamma^2 \Gamma^2 \vdash^* B_2,$$

...

$$\gamma^r \Gamma^r \vdash^* B_r.$$

Здесь Γ^s — это некоторая строка категорий с глубиной большей $i + 1$.

В грамматике G' данной строке категорий будет соответствовать вывод, начинающийся применением правила $A \rightarrow aB_1B_2 \dots B_r$. B_1, B_2, \dots, B_r имеют свои поддеревья вывода. Пусть позиция символа a в слове w равна k . Тогда поддеревья с корнями B_1, \dots, B_r порождают последовательные участки слова, следующие за позицией k . Пусть $w[i, j]$ — участок слова w , начинающийся в позиции i и заканчивающийся в позиции j . Обозначим номера первых символов

этих участков в первом дереве вывода через

$$p_1, p_2, \dots, p_r,$$

а конец последнего участка — через p_{r+1} . Тогда $p_1 = k + 1$, и поддерево с корнем B_j порождает участок $w[p_j, p_{j+1} - 1]$.

Во втором дереве вывода получаем аналогичные позиции

$$p'_1, p'_2, \dots, p'_r, p'_{r+1}.$$

Так как по предположению индукции категория $\gamma_u = (A/B_r/\dots/B_1)$ имеет одну и ту же позицию в обеих строках, а A порождает тот же участок слова, что и в первом дереве, имеем

$$p_1 = p'_1, \quad p_{r+1} = p'_{r+1}.$$

Предположим, что в строке Γ_2 категория $\gamma^{j'}$ глубины $i+1$ отличается от γ^j из строки $\gamma_u \gamma_{u+1} \dots \gamma_v$, но тогда деревья вывода с корнем B_j в G''' будут различными, что невозможно в силу однозначности G''' .

Рассмотрим теперь случай, когда сами категории совпадают, но хотя бы одна из них стоит в другой позиции. Пусть существует j , для которого

$$p_j \neq p'_j.$$

Возьмём наименьшее такое j . Тогда $j > 1$ и

$$p_{j-1} = p'_{j-1}.$$

Отсюда

$$p_j - p_{j-1} \neq p'_j - p'_{j-1},$$

то есть $(j-1)$ -е поддерево, имеющее корень B_{j-1} , в двух деревьях вывода порождает участки слова различной длины. Получается, что граница между поддеревьями B_{j-1} и B_j проходит в разных позициях. Из этого следует, что два дерева вывода слова w в грамматике G''' различны, что, опять же, невозможно из-за однозначности G''' . Таким образом, категории глубины $i+1$ должны совпадать. \square

ЛЕММА 4. *Каково бы ни было непустое слово w и категория γ , не существует двух различных строк категорий для w , сокращающихся до γ , то есть если $\Gamma_1, \Gamma_2 \in \delta(w)$, $\Gamma_1 \vdash^* \gamma$, $\Gamma_2 \vdash^* \gamma$, то $\Gamma_1 = \Gamma_2$.*

ДОКАЗАТЕЛЬСТВО. Является непосредственным следствием доказанной ранее леммы 3. \square

ТЕОРЕМА 5. Пусть G'' — однозначная КС-грамматика в нормальной форме Грейбах. Тогда существует $(*, *)$ -жёсткая грамматика G' такая, что $L(G'') = L(G')$.

ДОКАЗАТЕЛЬСТВО. Построенная выше грамматика G' является искомой. Так как, согласно лемме 4, двух различных строк категорий для w , сокращающихся до γ не существует, то G' является $(*, *)$ -жёсткой грамматикой. \square

Теперь мы можем доказать наш основной результат.

ДОКАЗАТЕЛЬСТВО ТЕОРЕМЫ 1. Для начала преобразуем G в КС-грамматику в нормальной форме Хомского G^* . Как известно, такое преобразование сохраняет однозначность. Затем преобразуем G^* в КС-грамматику в нормальной форме Грейбах G'' , используя преобразования, представленные в [2]. Данные преобразования, как описано в статье, сохраняют однозначность, а поэтому грамматика G'' будет однозначной. Согласно теореме 5, для однозначной КС-грамматики в нормальной форме Грейбах существует $(*, *)$ -жёсткая грамматика G' такая, что $L(G'') = L(G')$, а значит, $L(G) = L(G')$. \square

Заключение

В данной работе была исследована связь между однозначными контекстно-свободными грамматиками и $(*, *)$ -жёсткими категориальными грамматиками. Основным результатом является доказательство того, что для однозначной КС-грамматики G существует $(*, *)$ -жёсткая грамматика G' такая, что $L(G) = L(G')$.

Полученный результат уточняет выразительные возможности рассматриваемого класса категориальных грамматик и может быть использован при дальнейшем исследовании ограничений жёсткости и свойств однозначности в формальных грамматиках.

Список литературы

- [1] Bar-Hillel, Y. On categorial and phrase structure grammars / Y. Bar-Hillel, H. Gaifman, E. Shamir // Bulletin of the Research Council of Israel. — 1960. — Vol. 9F. — P. 1–16. — URL: <https://cris.huji>.

[ac.il/en/publications/on-categorical-and-phrase-structure-grammars/](https://www.researchgate.net/publications/on-categorical-and-phrase-structure-grammars/).

- [2] Hotz, G. Normal-form transformations of context-free grammars // Acta Cybernetica. 1978. — Vol. 4, № 1. — P. 65–84. — URL: https://www.researchgate.net/publication/220123168_Normal-form-transformations_of_context-free_grammars.
- [3] Карлов, Б. Н. (m, n) -жёсткие категориальные грамматики // Вестник ТвГУ. Серия: Прикладная математика. — 2017. — № 4. — С. 7–23. — DOI: <https://doi.org/10.26456/vtpmk185>.

Библиографическая ссылка

Федоров, И. В. Выразительные возможности классов категориальных грамматик // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 207–214.

Сведения об авторах

ФЕДОРОВ ИВАН ВЛАДИМИРОВИЧ

Студент магистратуры

Направление «Прикладная математика и информатика»

УДК 004.932

Метод распознавания числовых показаний из сканированных анкетных документов

Харинова С. А.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
информационных технологий А. Б. Семёновым

Аннотация. В статье рассматривается метод распознавания рукописных цифр из сканированных анкетных документов. В основе метода лежит использование свёрточных нейронных сетей (CNN) для классификации изображений и алгоритм Рамера – Дугласа – Пекера для векторной аппроксимации треков рисования. Описан подход к синтетической генерации сбалансированного датасета, включающий аугментацию данных и деформацию векторных контуров для повышения устойчивости модели к реальным артефактам сканирования. Представлена архитектура CNN и результаты экспериментов, демонстрирующие точность распознавания 98,1 % на тестовой выборке.

КЛЮЧЕВЫЕ СЛОВА: распознавание цифр, свёрточные нейронные сети, анкетные документы, аугментация данных, балансировка датасета.

Введение

Распознавание информации из сканированных документов играет важную роль в процессах перевода документов в цифровой вид. Ежедневно обрабатываются миллионы документов, в которых содержится важная числовая информация: даты рождения, суммы, коды, номера документов. Современные системы оптического распознавания символов (OCR), такие как FineReader или Tesseract, демонстрируют хорошие результаты при обработке печатных текстов, однако при работе с анкетными документами они сталкиваются с рядом трудностей. К примеру, сложность структуры анкет, наличия разнообразного почерка, низкое качество сканов или наличие помарок или искажений.

Таким образом, разработка специализированного метода распознавания числовой информации является актуальной задачей.

1. Описание

Разрабатываемый метод ориентирован на работу с анкетными бланками, имеющими чётко выраженную структуру. При сканировании с разрешением 300 dpi клетка размером 10×10 мм преобразуется в изображение размером примерно 118×118 пикселей. Это значение выбрано как стандартное для OCR и является достаточным для распознавания отдельных цифр.

На вход модели подаётся изображение, содержащее одну рукописную цифру, выделенную из соответствующего поля анкеты. Задача метода — определить, какая цифра (от 0 до 9) на нём изображена.

2. Синтетическая генерация и аугментация данных

В работе применяется метод синтетической генерации, позволяющий контролировать разнообразие и сложность примеров. Особенность подхода — сохранение не растрового изображения, а последовательности координат точек трека рисования. Это даёт возможность гибко аугментировать данные на векторном уровне.

Для аппроксимации исходного трека используется алгоритм Рамера – Дугласа – Пекера, позволяющий уменьшить число точек кривой при сохранении её формы.

На рис. 1 представлен графический интерфейс для рисования цифр. Пользователь создаёт цифру в специальном окне с помощью мыши, а программа фиксирует каждую точку с координатами в порядке движения, формируя исходный векторный трек. В разработанной программе также предусмотрена возможность загрузки внешних растровых изображений для последующего преобразования их в векторный трек.

Аугментация выполняется на уровне векторного трека с использованием деформации ограничивающего прямоугольника (bounding box). Задавая различные положения углов, можно получать реалистичные варианты цифр: наклон, растяжение, сжатие, перспективные искажения, смещение внутри поля анкеты.

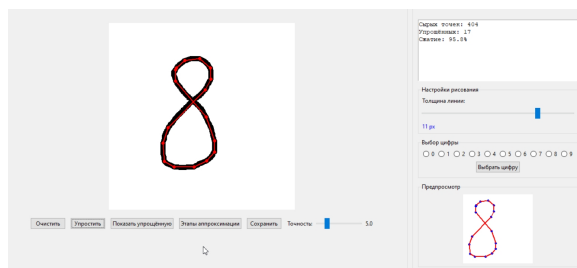


Рис. 1. Пример создания цифры 8.

На рис. 2 показан пример деформации одной и той же цифры при изменении углов ограничивающего прямоугольника. Такое преобразование позволяет имитировать наклон, растяжение, сжатие и перспективные искажения, характерные для рукописного ввода в анкетах.

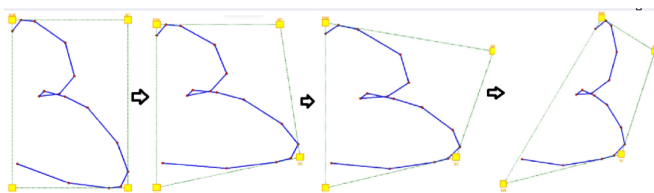


Рис. 2. Этапы деформации цифры.

На рис. 3 приведены примеры сгенерированных данных. Из одного исходного трека с помощью описанных преобразований создаются десятки новых, реалистичных вариантов написания цифры, отличающихся наклоном, размером, положением в поле и малыми искажениями.

В результате получен датасет для обучения модели. Все сгенерированные изображения представляют собой уникальные варианты, отличные от исходных благодаря применённым преобразованиям.

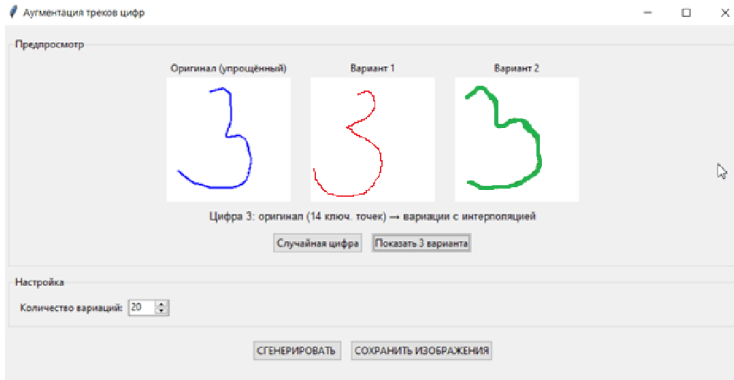


Рис. 3. Интерфейс для аугментации треков.

3. Архитектура свёрточной нейронной сети

Для классификации цифр разработана специализированная CNN-архитектура. Входные изображения имеют размер 118×118 пикселей в градациях серого.

Модель состоит из трёх свёрточных блоков.

- 1) Первый блок: свёртка с 32 фильтрами $3 \times 3 \rightarrow$ Batch Normalization \rightarrow ReLU \rightarrow MaxPooling (2×2). Размер карт признаков уменьшается с 118×118 до 59×59 .
- 2) Второй блок: свёртка с 64 фильтрами $3 \times 3 \rightarrow$ Batch Normalization \rightarrow ReLU \rightarrow MaxPooling (2×2). Размер карт — 29×29 .
- 3) Третий блок: свёртка с 128 фильтрами $3 \times 3 \rightarrow$ Batch Normalization \rightarrow ReLU \rightarrow MaxPooling (2×2). Размер карт — 14×14 .

Для демонстрации процесса извлечения признаков на рис. 4 показана карта активации первого свёрточного слоя (сумма всех 32 фильтров) при подаче на вход изображения цифры 7.

После свёрточной части карты признаков преобразуются в одномерный вектор ($128 \times 14 \times 14 = 25088$ значений) и подаются в полносвязную часть.

- Полносвязный слой: $25088 \rightarrow 512$, ReLU, Dropout (0,3).

- Полносвязный слой: $512 \rightarrow 256$, ReLU, Dropout (0,3).
- Полносвязный слой: $256 \rightarrow 128$, ReLU, Dropout (0,3).
- Выходной слой: $128 \rightarrow 10$, Softmax.

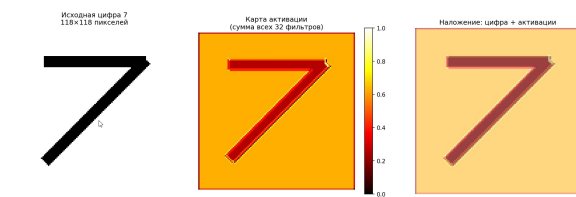


Рис. 4. Визуализация карты активации первого свёрточного слоя для цифры 7.

Для предотвращения переобучения используются Dropout и Batch Normalization, оказывающая регуляризующий эффект.

4. Экспериментальные результаты

Обучение модели проводилось на сбалансированном датасете из 12 500 изображений (1 000 на каждую цифру). Датасет разделялся на обучающую (80 %), валидационную (10 %) и тестовую (10 %) выборки. Функция потерь — CrossEntropyLoss, оптимизатор — Adam (learning rate = 0,001, weight decay = 10^{-4}), размер батча — 32.

Разработанный графический интерфейс позволяет визуализировать процесс распознавания. На рис. 5 представлен пример работы модели: загруженное изображение рукописной цифры распознаётся с высокой вероятностью. Также есть возможность не только загружать изображение, но и рисовать его в соответствующем окне.

В ходе тестирования выявлено, что наибольшая путаница может возникнуть между цифрами 3 и 5, 7 и 1, а также 8 и 6, что соответствует их схожести в рукописном написании. Однако благодаря высокой точности модели ошибки на тестовых примерах единичны.

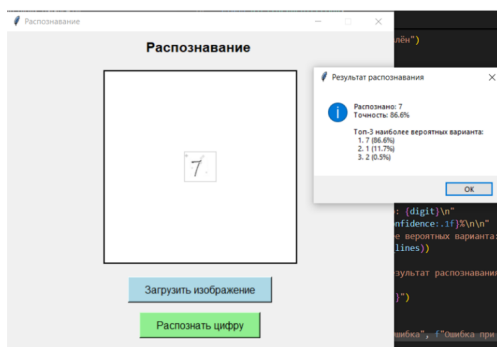


Рис. 5. Результат распознавания цифры 7.

Результаты распознавания на тестовой выборке представлены в табл. 1.

Таблица 1. Результаты распознавания цифр на сбалансированном датасете.

Цифра	Точность распознавания, %
0	97,8
1	98,2
2	98,9
3	97,3
4	98,1
5	97,9
6	97,8
7	99,2
8	98,2
9	97,7
Средняя	98,1

Заключение

В работе предложен метод распознавания числовых показаний из сканированных анкетных документов, основанный на свёрточных

нейронных сетях. Перечислим основные результаты.

- 1) Разработан подход к синтетической генерации сбалансированного датасета с использованием векторного представления треков и алгоритма Рамера – Дугласа – Пекера, позволяющего сократить объём данных на 85–90 % при сохранении формы цифр.
- 2) Предложена архитектура CNN, обеспечивающая точность распознавания 98,1 % на тестовой выборке.
- 3) Реализованы механизмы аугментации на уровне векторных контуров, имитирующие реальные искажения при сканировании.

Разработанный метод может быть использован в системах автоматической обработки анкет, экзаменационных бланков и других структурированных документов с числовыми полями. Дальнейшие исследования могут быть направлены на распознавание букв и числовых последовательностей произвольной длины.

Список литературы

- [1] Гонсалес, Р. С. Цифровая обработка изображений / Р. С. Гонсалес, Р. Э. Вудс ; пер. с англ. Л. И. Рубанова, П. А. Чочиа. — 3-е изд., испр. и доп. — М. : Техносфера, 2012. — 1104 с. — URL: <https://djvu.online/file/IqY5yzPA8F2sK>.
- [2] Николенко, С. Глубокое обучение. Погружение в мир нейронных сетей / С. Николенко, А. Кадури, Е. Архангельская. — СПб. : Питер, 2018. — 480 с. — ISBN 978-5-496-00805-3. — URL: <https://djvu.online/file/hvNHupS7Aoyui>.
- [3] Хмелев, Р. В. Итеративная аппроксимация последовательностей по максимуму периметра и с использованием неравенства треугольников / Р. В. Хмелев // Компьютерная оптика. — 2005. — № 27. — С. 155–164. — URL: <https://computeroptics.ru/KO/Annnot/KO27/27027.html>.

Библиографическая ссылка

Харинова, С. А. Метод распознавания числовых показаний из сканированных анкетных документов // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 215–222.

Сведения об авторах

ХАРИНОВА СОФЬЯ АЛЕКСАНДРОВНА

Студент магистратуры

Направление «Фундаментальная информатика и информационные технологии»

УДК 330.4, 004.42

Прогнозирование влияния рекомендательных систем на экономические показатели интернет-коммерции

Шварц Н. А.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа Н. А. Мансуровой

Аннотация. В работе предложена экономико-математическая модель, связывающая метрики качества рекомендательных алгоритмов с бизнес-показателями интернет-коммерции. Разработан программный сервис, позволяющий пройти полный цикл тестирования рекомендательного алгоритма — от расчёта метрик качества до прогнозирования экономического эффекта от внедрения. Проведена валидация модели, приведён пример использования сервиса. **Ключевые слова:** рекомендательные системы, метрики качества рекомендаций, экономико-математическое моделирование, прогнозирование бизнес-показателей, интернет-коммерция.

Введение

Рекомендательные системы (РС) обеспечивают пользователям персонализированные предложения товаров и контента и сегодня рассматриваются компаниями сферы интернет-коммерции как один из ключевых инструментов роста выручки, повышения конверсии и удержания клиентов. Вместе с тем разработка и эксплуатация РС требуют существенных затрат, а многообразие доступных архитектур и алгоритмов создаёт дополнительную сложность выбора. В этих условиях возникает прикладная потребность в инструментах, позволяющих заранее прогнозировать влияние планируемого внедрения РС на целевые бизнес-показатели.

Среди существующих подходов к оценке рекомендательных систем А/В-тестирование и пользовательские исследования обеспечи-

вают высокую достоверность, но требуют значительных временных и финансовых ресурсов, а также развёрнутой пользовательской базы. Имитационное моделирование предполагает построение сложных поведенческих моделей и высокую квалификацию исполнителей. На этом фоне офлайн-оценка на основе статических прогнозов по заранее подготовленным наборам данных оказывается наиболее доступным инструментом, особенно для малых и средних предприятий, а также для разработчиков, тестирующих свои наработки в условиях ограниченных ресурсов.

Целью работы является разработка экономико-математической модели и программного сервиса для прогнозирования влияния рекомендательных систем на экономические показатели интернет-коммерции на основе метрик качества алгоритмов рекомендаций. Научная новизна работы заключается в построении модели, связывающей метрики качества рекомендательных алгоритмов с ключевыми бизнес-показателями, а также в реализации программного сервиса, обеспечивающего полный цикл тестирования рекомендательного алгоритма — от расчёта метрик качества до прогнозирования бизнес-показателей.

1. Метрики качества и бизнес-показатели рекомендательных систем

В современной классификации выделяют три основные категории метрик качества рекомендательных систем [2]:

- 1) метрики точности прогнозирования (predictive metrics);
- 2) метрики качества ранжирования (ranking metrics);
- 3) поведенческие метрики (behavioral metrics).

Метрики точности прогнозирования. Суть этой группы метрик состоит в измерении средней ошибки (MAE или RMSE) между реальными оценками пользователя и оценками, которые рекомендательная система выставила как потенциальную релевантность товара для него. Однако на практике данная группа метрик плохо применима к системам, не формирующим явный прогноз оценок, и к тому же слабо отражает реальный рост качества рекомендаций.

Поэтому в настоящее время метрики точности прогнозирования редко используются при оценке качества работы РС.

Метрики качества ранжирования. Рекомендательная система в результате своей работы формирует для пользователя список товаров, которые должны представлять для него наибольший интерес. Однако пользовательский интерфейс платформы, реализующей рекомендации, не позволяет отобразить сотни позиций, сохраняя при этом удобство восприятия. Поэтому длина рекомендуемого списка оказывается ограниченной. Чем строже это ограничение, тем важнее место наиболее релевантного товара в выдаче. Метрики качества ранжирования оценивают, насколько верно элементы упорядочены по их истинной релевантности. Ниже представлены основные метрики качества ранжирования, применяемые для оценки РС:

- 1) средний обратный ранг (MRR);
- 2) коэффициент попаданий (HR);
- 3) точность (PREC);
- 4) полнота (REC);
- 5) F_1 -мера (F_1);
- 6) дисконтированный кумулятивный выигрыш (DCG).

Поведенческие метрики. Данная категория метрик частично заменила собой классический подход к оценке рекомендательных систем, используемый в метриках точности прогнозирования. В отличие от узкой направленности на точность предсказания релевантности, поведенческие метрики позволяют сформировать представление о характеристиках рекомендательной системы в целом. Ниже представлены основные поведенческие метрики, используемые для оценки рекомендательных систем:

- 1) новизна (NOV);
- 2) разнообразие (DIV);
- 3) сюрпризность (SER);
- 4) покрытие каталога (COV);
- 5) равномерность распределения рекомендаций (DC).

Комбинирование метрик качества ранжирования и поведенческих метрик обеспечивает комплексный подход к оценке эффективности рекомендательных систем. Сопоставление результатов, полученных в рамках этих подходов, позволяет перейти к анализу того, каким образом выявленные характеристики работы системы отражаются на результатах бизнеса. Ниже приведены пять ключевых бизнес-показателей интернет-коммерции, на которые рекомендательные системы оказывают наиболее существенное влияние [3]:

- 1) коэффициент «кликабельности» (CTR);
- 2) принятие рекомендаций и конверсия (AC);
- 3) объём продаж и доход (SR);
- 4) структура продаж (SD);
- 5) поведение и вовлечённость пользователей (UEB).

2. Модель прогнозирования влияния рекомендательных систем

Основу прогнозирующей модели составляет базовая статистическая модель, описывающая усреднённое влияние внедрения рекомендательной системы на ключевые бизнес-показатели [1]. Ожидаемые диапазоны изменения представлены в табл. 1.

Таблица 1. Диапазоны изменения бизнес-показателей.

Показатель	Диапазон
Коэффициент «кликабельности» (CTR)	[5 %; 60 %]
Принятие рекомендаций и конверсия (AC)	[3 %; 25 %]
Объём продаж и доход (SR)	[5 %; 35 %]
Структура продаж (SD)	[1 %; 65 %]
Поведение и вовлечённость пользователей (UEB)	[10 %; 50 %]

Для дальнейшего построения связи бизнес-показателей с метриками качества работы рекомендательной системы введём вспомогательные обозначения. Пусть Y — множество бизнес-показателей; Q — множество метрик качества ранжирования; S — множество поведенческих метрик. Все рассматриваемые метрики нормализованы и принимают значения в диапазоне $[0; 1]$.

Далее введём модель, позволяющую сделать прогноз изменения каждого бизнес-показателя чувствительным к качеству рекомендаций, сохраняя при этом согласованность с ранее сформированными статистическими диапазонами. Сначала проведем агрегацию метрик качества работы рекомендательной системы:

$$z_y^{(q)} = \sum_{q \in Q} w_q^y \cdot v_q, \quad z_y^{(s)} = \sum_{s \in S} w_s^y \cdot v_s.$$

Здесь $z_y^{(q)}$ — индекс качества ранжирования, $z_y^{(s)}$ — поведенческий индекс качества; $v_q, v_s \in [0; 1]$ — нормализованные значения метрик; $w_q^y, w_s^y \geq 0$ — веса, отражающие значимость соответствующих метрик относительно конкретного бизнес-показателя y . При этом сумма весов равняется единице:

$$\sum_{q \in Q} w_q^y = 1, \quad \sum_{s \in S} w_s^y = 1.$$

Такое определение весов гарантирует $z_y^{(q)}, z_y^{(s)} \in [0; 1]$.

Объединим полученные компоненты в единый индекс качества рекомендаций:

$$z_y = \alpha_y \cdot z_y^{(q)} + (1 - \alpha_y) \cdot z_y^{(s)},$$

где $\alpha_y \in [0; 1]$ — параметр, задающий относительный вклад качества ранжирования по сравнению с поведенческими характеристиками для бизнес-показателя y . По построению $z_y \in [0; 1]$.

Далее переведём z_y в долю реализованного эффекта с помощью нормированной функции экспоненциального насыщения:

$$g_y(z_y) = \frac{1 - \exp(-\lambda_y \cdot z_y^{\kappa_y})}{1 - \exp(-\lambda_y)},$$

где $\lambda_y > 0$ — параметр скорости насыщения, определяющий, насколько быстро кривая эффекта выходит на плато при росте z_y , а $\kappa_y > 0$ — параметр формы старта кривой, задающий пороговое значение качества рекомендаций, при котором эффект переходит в фазу быстрого роста. Нормирование в знаменателе обеспечивает $g_y(0) = 0$ и $g_y(1) = 1$, следовательно $g_y(z_y) \in [0; 1]$ при $z_y \in [0; 1]$.

Итоговая модель прогнозирования изменения бизнес-показателя y выглядит следующим образом:

$$\widehat{\Delta y} = B_y + (T_y - B_y) \cdot g_y(z_y),$$

где B_y и T_y — нижняя и верхняя границы ожидаемого эффекта, взятые из табл. 1. Ограничение $\widehat{\Delta y} \in [B_y; T_y]$ выполняется конструктивно.

Для валидации модели было проведено сопоставление её прогнозов с фактическими изменениями бизнес-показателей, зафиксированными по результатам А/В-тестов. В рамках эксперимента модели передавались значения метрик качества рекомендательного алгоритма до и после его модификации, после чего осуществлялась оценка ожидаемого изменения целевого показателя. Сравнение прогнозных значений с фактически наблюдаемыми изменениями представлено на рис. 1.

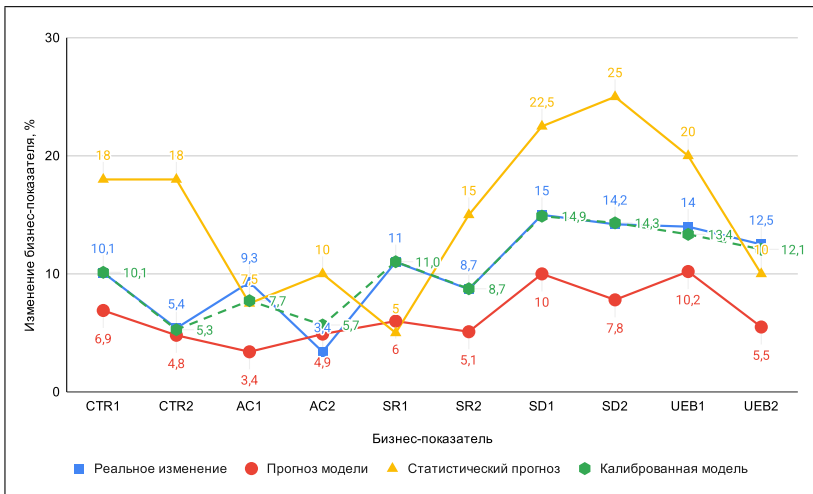


Рис. 1. Сравнение реальных изменений и прогнозов моделей.

Числовые результаты сравнения прогнозов с реальными изменениями бизнес-показателей приведены в табл. 2. Базовая статистическая модель демонстрирует наименьшую точность, систематически

Таблица 2. Результаты валидации модели.

Метрика	Статистика	Модель	Калиброванная модель
MAE	6,8	4,5	0,5
MAPE	84,1	39,9	9,7
RMSE	7,48	4,64	0,91
Корреляция	0,44	0,74	0,97
R ²	0,19	0,54	0,94

завышая ожидаемый эффект от замены алгоритма. Предложенная модель с аналитически заданными параметрами показывает заметно более высокую точность, корректно воспроизводя направление и относительный масштаб изменений, однако склонна к занижению прогноза в абсолютных значениях. Калиброванная модель, параметры которой были дообучены на валидационной выборке, обеспечивает наилучшее согласие с реальными данными.

Полученные результаты подтверждают, что предложенная модель адекватно описывает зависимость бизнес-показателей от качества рекомендательного алгоритма и превосходит базовый статистический подход даже без калибровки. При этом высокая точность калиброванной версии указывает на то, что при накоплении представительной обучающей выборки прогностические возможности сервиса могут быть дополнительно повышены.

3. Сервис прогнозирования влияния рекомендательных систем

Для получения значений метрик качества работы алгоритмов рекомендаций был выбран набор данных Instacart Market Basket Analysis. Сервис Instacart является агрегатором супермаркетов, вследствие чего его данные хорошо отражают специфику рекомендательных систем в сфере интернет-коммерции. В состав данных входят: календарная информация о заказе (день недели, час оформления, число дней с момента предыдущего заказа); поведенческая информация (порядок добавления товаров в корзину, признак перезаказа); товарная информация (принадлежность продуктов к категориям и

подкатегориям). Всего датасет содержит сведения о 206 209 уникальных пользователях, 49 689 товарных позициях и 3 421 084 заказах, включающих 33 819 108 добавлений товаров.

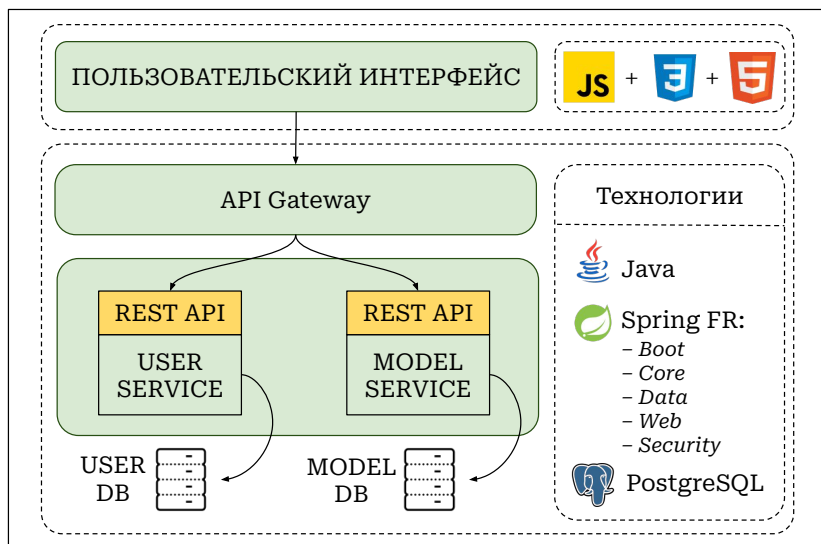


Рис. 2. Архитектура сервиса и используемые технологии.

Архитектура сервиса, представленная на рис. 2, построена по микросервисному принципу и включает пользовательский интерфейс, шлюз API и два специализированных backend-компонента, каждый из которых имеет собственное хранилище данных. Пользовательский интерфейс реализован с применением HTML, CSS и JavaScript, backend-компоненты — на языке Java с использованием фреймворка Spring. В качестве системы управления базами данных выбрана PostgreSQL. Сервис предоставляет два интерфейса взаимодействия — программный и пользовательский.

Программный интерфейс реализован в виде REST API и обеспечивает доступ ко всей функциональности сервиса. Через него выполняются операции, связанные с жизненным циклом учетной записи пользователя и управлением накопленными данными: регистрация и удаление профиля, доступ к результатам ранее выполненных запус-

ков, а также их перенос посредством импорта и экспорта в формате CSV. Помимо этого, программный интерфейс предоставляет доступ к основной прикладной логике сервиса: запуску тестовых сценариев для оценки рекомендательных алгоритмов, вычислению отдельных метрик качества, расчету прогнозных значений бизнес-показателей на основе модели, а также настройке параметров модели.

Пользовательский интерфейс, представленный на рис. 3, организован в виде информационной панели и позволяет одновременно сопоставлять несколько конфигураций рекомендательных алгоритмов. В нижней части экрана размещается раскрывающийся список добавленных конфигураций с детализацией значений метрик качества работы алгоритма, в центральной части — два графика, отображающие прогнозируемое изменение бизнес-показателей в процентах и значения метрик качества. Помимо визуализации, интерфейс предоставляет средства управления конфигурациями: их добавление, загрузку и сохранение через импорт и экспорт файлов, а также очистку текущей сессии.

На рис. 3 также представлено сравнение трёх алгоритмов, традиционно применяемых в сценариях «с этим товаром покупают» и «похожие товары»: коллаборативной фильтрации на основе сходства товаров (Item-based CF), поиска ассоциативных правил (Apriori) и контентной фильтрации (TF-IDF). Результаты показывают, что Apriori демонстрирует наилучший результат по коэффициенту кликабельности (CTR: +32,25 %) и принятию рекомендаций (AC: +5,34 %), Item-based CF лидирует по объёму продаж и дохода (SR: +12,44 %) и вовлечённости пользователей (UEB: +35,64 %), тогда как TF-IDF существенно превосходит остальные алгоритмы по влиянию на структуру продаж (SD: +16,85 %). Таким образом, предложенная модель позволяет не только определить алгоритм, демонстрирующий наилучшие результаты, но и установить, в каком именно аспекте бизнеса проявляются его преимущества.

Заключение

В ходе исследования разработана экономико-математическая модель, связывающая метрики качества рекомендательных алгоритмов с ключевыми бизнес-показателями интернет-коммерции, и



Рис. 3. Пользовательский интерфейс сервиса: сравнение различных конфигураций рекомендательных алгоритмов.

реализован программный сервис, обеспечивающий полный цикл тестирования РС — от расчёта метрик качества до прогнозирования экономического эффекта внедрения. Валидация подтвердила, что модель адекватно описывает зависимость бизнес-показателей от качества алгоритма и превосходит базовый статистический подход. Разработанная модель позволяет не только выявить лучший алгоритм, но и определить, в каких аспектах бизнеса проявляются его преимущества, что показано на примере использования сервиса.

Перспективы развития работы связаны с двумя основными направлениями. Первое — совершенствование самой модели за счёт накопления данных реальных А/В-тестов, расширения набора учитываемых метрик и уточнения их взаимосвязей с бизнес-показателями. Второе — расширение функционала сервиса для тестирования рекомендательных систем на устойчивость к их ключевым проблемам: холодному старту и разреженности данных, ограничениям производительности, нагрузке и внешним манипуляциям.

Список литературы

- [1] Biasio, A.D. Economic Recommender Systems — a Systematic Review / A.D. Biasio, N. Navarin, D. Jannach // Electronic Commerce Research and Applications. — 2024. — Vol.63. — P. 101352. — DOI: <https://doi.org/10.1016/j.elerap.2023.101352>.
- [2] Jadon, A. A Comprehensive Survey of Evaluation Techniques for Recommendation Systems / A. Jadon, A. Patil // Communications in Computer and Information Science. — 2024. — Vol. 2185. — P. 281–304. — DOI: https://doi.org/10.1007/978-3-031-71484-9_25.
- [3] Jannach, D. Measuring the Business Value of Recommender Systems / D. Jannach, M. Jugovac // ACM Transactions on Management Information Systems (TMIS). — 2019. — Vol. 10, № 4. — Art. 16. — DOI: <https://doi.org/10.1145/3370082>.

Библиографическая ссылка

Шварц, Н. А. Прогнозирование влияния рекомендательных систем на экономические показатели интернет-коммерции // Студенческая

конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 223–234.

Сведения об авторах

ШВАРЦ НИКИТА АЛЕКСАНДРОВИЧ
Студент магистратуры
Направление «Прикладная информатика»

УДК 519.767.6, 004.912

Создание гибридного метода классификации текстов на основе меры TF-IDF и рекуррентных нейронных сетей

Эквист В. Р.

Тверской государственный университет

Рекомендовано к публикации доцентом кафедры информатики

Б. Н. Карловым

Аннотация. Работа посвящена категориальной классификации русскоязычных текстов, в частности новостных статей. Предложен гибридный метод, сочетающий в себе отбор информативных слов с помощью TF-IDF с нейросетевой моделью BiLSTM. Для выявления влияния на качество результатов проведены эксперименты с различными значениями параметра n — количество информативных слов, которые подаются на вход нейронной сети. Разработанное решение может быть использовано для автоматической категоризации новых новостных статей.

КЛЮЧЕВЫЕ СЛОВА: классификация новостных статей, TF-IDF, рекуррентные нейронные сети, BiLSTM.

Введение

По мере того, как сеть Интернет распространялась и становилась более доступной, стремительно рос и объём информации, ежедневно публикуемой в этой среде, что привело к росту потребности в автоматизированной обработке этих данных. Одним из наиболее востребованных направлений в данной области является классификация новостных статей по категориям, лежащая в основе работы рекомендательных систем, агрегаторов новостных статей, сервисов мониторинга и аналитики информационного поля. Ручная разметка и распределение статей по категориям трудозатратны и плохо масштабируются, что приводит к росту интереса к методам автоматической классификации текстов новостных статей.

В данной статье предлагается гибридный метод классификации новостных статей, основанный на использовании статистической меры TF-IDF (Term Frequency – Inverse Document Frequency) в сочетании с рекуррентными нейронными сетями. Такой подход позволяет объединить преимущества двух методов: TF-IDF выделяет ключевые слова, а рекуррентные нейронные сети учитывают последовательную природу текста и контекстные зависимости между словами.

1. Лингвистическая и техническая нормализация текста

На первом этапе исходный текст новостной статьи приводится к единому виду, чтобы уменьшить влияние шума и разнообразия словоформ, характерного для русского языка. На данном этапе выполняются следующие операции.

- Приведение к нижнему регистру. Позволяет снизить размер словаря и повысить устойчивость модели.
- Удаление HTML-тегов. Такие элементы могут встречаться в «сырых» данных и засоряют токенизацию, не неся в себе никакой информации.
- Обработка чисел. Числа заменяются на специальный токен <NUM>, что позволяет сохранить сам факт наличия числа, не заставляя модель запоминать конкретные значения, которые редко бывают устойчивыми признаками категории.
- Токенизация. Из текста извлекаются слова и токен <NUM>, при этом игнорируются знаки препинания и символы, не попадающие под заданный шаблон.
- Удаление стоп слов. Убираются служебные слова, которые не помогают различать категории, но занимают место в словаре.
- Лемматизация. Каждое слово приводится к начальной форме (лемме). Для этого используется `ru morphology2` — библиотека для морфологического анализа русского языка. В результате снижается разреженность признаков и улучшается обобщающая способность модели.

Этот этап уменьшает размер словаря, снижает шум в данных и делает признаки более устойчивыми. Без лемматизации и нормализации у модели ухудшается качество категориальной классификации из-за того, что модель распределяет статистику по множеству словоформ.

2. Статистическая мера TF-IDF и отбор токенов

После предобработки каждый документ предоставляет собой последовательность лемм. Далее будем применять TF-IDF для оценки информативности слов в конкретной статье.

TF-IDF (Term Frequency – Inverse Document Frequency) – это вес слова t в документе d , который растёт, если слово часто встречается в документе, и уменьшается, если слово встречается во многих документах коллекции.

В данной работе для вычисления меры используется стандартный сценарий `TfidfVectorizer` из библиотеки `scikit-learn` [2]. В этом случае веса считаются так:

- 1) tf – частота слова t в документе d . В нашем случае оно определяется так:

$$tf(t, d) = w_{t,d},$$

где $w_{t,d}$ число вхождений слова t в документ d ;

- 2) idf со сглаживанием:

$$idf(t) = \ln \left(\frac{1 + u}{1 + df(t)} \right) + 1,$$

где u – число документов в корпусе, $df(t)$ – число документов, где встречается слово t ;

- 3) $tfidf$ до нормировки:

$$tfidf(t, d) = tf(t, d) \cdot idf(t);$$

- 4) итоговый вектор \vec{u}_d делится на L2-норму:

$$\vec{v}_d = \frac{\vec{u}_d}{\|\vec{u}_d\|_2} = \frac{\vec{u}_d}{\sqrt{u_1^2 + u_2^2 + \dots + u_m^2}}.$$

TF-IDF применяется не как окончательный признак для модели, а как фильтр: каждый документ получает свой TF-IDF вектор, отбираются n слов с максимальными весами, в исходной последовательности остаются только выбранные раннее слова. Такой отбор уменьшает число токенов, что попадают в нейросеть, и концентрирует модель на словах, наиболее характерных для данной статьи. Это позволяет снизить шум и ускорить обучение, улучшить качество классификации, отбросив лишние слова, но при малых значениях n результат может ухудшаться из-за потери части контекста.

После этапа фильтрации каждый документ представляет собой укороченный список лемм, который необходимо перевести в числовую последовательность для подачи в нейросеть. Для этого строится словарь токенов, где каждому слову присваивается индекс, полученные последовательности выравниваются, формируя входную матрицу.

3. Архитектура нейронной сети

Для классификации новостных статей используется последовательная модель, основанная на двунаправленной рекуррентной сети [1, с. 129]. Данная архитектура была выбрана с учётом того, что необходимо учитывать контекстные зависимости текста и эффективно обрабатывать последовательности переменной длины. Модель состоит из следующих слоёв.

- **Embedding.** Этот слой преобразует числовые идентификаторы слов в плотные векторы и содержит матрицу весов, которая обучается таким образом, чтобы слова с похожим смыслом располагались близко в векторном пространстве.
- **Двунаправленный LSTM (BiLSTM).** Слой LSTM (Long Short-Term Memory) запоминает важную информацию в длинных последовательностях, избегая проблем затухания градиентов. Двунаправленность означает, что модель обрабатывает текст дважды, в прямом и обратном направлениях, объединяя контекстную информацию с двух сторон.
- **Dropout.** Слой регуляризации, случайно отключающий часть нейронов для предотвращения переобучения.

- Dense. Структура заканчивается двумя полносвязными слоями. Первый слой с функцией активации ReLU служит для преобразования признаков из LSTM в более абстрактное и нелинейное представление. Последний слой имеет количество нейронов, равное количеству категорий, и использует функцию активации Softmax для преобразования выхода сети в распределение вероятностей по категориям.

Данная архитектура обеспечивает баланс между выразительной способностью модели, учётом контекста и устойчивостью к переобучению, что делает её эффективным решением для задачи многоклассовой классификации новостей.

4. Результаты экспериментов

Экспериментальная часть работы была проведена на датасете новостных статей интернет-газеты Gazeta.ru [3]. Для оценки качества моделей использовались стандартные метрики многоклассовой классификации. Помимо точности (accuracy), также использовались такие метрики, как precision, recall, f1-score и их усреднённые версии: macro и weighted average. В дальнейшем в результатах будут указываться метрики accuracy, macro F1-score (mF1) и weighted F1-score (wF1). В табл. 1 представлены результаты обучения моделей с различным параметром n .

Параметр n определяет количество слов, отбираемых для каждого документа на основе TF-IDF. При малых значениях n в модель попадает ограниченный набор весомых токенов, что снижает шум и уменьшает размер входа, но может привести к потере контекста и ухудшению определения близких категорий. При увеличении параметра n сохраняется больше контекстной информации, что улучшает качество классификации, но вместе с этим возрастает длина входной последовательности и вычислительная сложность.

В ходе обучения моделей оценивалось влияние значения n на итоговые метрики. Увеличение n для некоторых наборов данных приводит к увеличению метрик, тогда как для других — к уменьшению, что может быть связано с ростом шума или переобучением.

Проведём проверку работы моделей на новостных статьях, что не присутствуют в исходном датасете. По результатам этого тестирования можно будет сделать вывод о практической применимости

Таблица 1. Результаты обучения моделей.

	$n = 10$	$n = 15$	$n = 20$
Accuracy	0,8279	0,8435	0,8452
Время	0,47 ч	0,53 ч	0,51 ч
mF1	0,7613	0,7834	0,7687
wF1	0,8244	0,8407	0,84

модели и о том, какие типы ошибок характерны, например, смешение близких по теме категорий. Далее кратко опишем текст статей и к каким категориям они относятся, категории перенесены такими, какими они были на сайтах интернет-газет.

- 1) Статья про штрафы и рост цен для автомобилистов. Категория — автомобили.
- 2) Статья про выход игровой консоли и её характеристики. Категория — технологии.
- 3) Статья про выставку о семье Бенуа. Категория — афиша Plus (культура).
- 4) Статья про запуск горячей линии для предпринимателей по вопросам налоговой реформы. Категория — финансы.
- 5) Статья про фотографии времён Великой Отечественной, их описание и исторический контекст. Категория — наука и техника.

В табл. 2 приведены результаты практического тестирования моделей.

Эксперименты показали, что параметр n существенно влияет на качество классификации модели и требует подбора под конкретный датасет. В нашем случае значение $n = 15$ обеспечивает достаточное сохранение контекста без избыточного роста шума.

Заключение

В работе разработан метод многокатегориальной классификации текстов на основе предобработки текста, отбора информативных слов

Таблица 2. Результаты практического тестирования.

$n \backslash \text{№}$	1	2	3	4	5
10	Авто 0,6574	Технологии 0,9576	Культура 0,9922	Бизнес 0,8979	Культура 0,2713
	Бизнес 0,2362	Бизнес 0,0322	Общество 0,0041	Финансы 0,0583	Общество 0,2433
	Общество 0,0799	Политика 0,0042	Спорт 0,0014	Общество 0,0316	Политика 0,1930
15	Авто 0,8810	Технологии 0,9765	Культура 0,9519	Бизнес 0,7951	Наука 0,7974
	Бизнес 0,0804	Бизнес 0,0154	Общество 0,0297	Финансы 0,1205	Армия 0,0572
	Общество 0,0142	Наука 0,0042	Спорт 0,0050	Общество 0,0606	Общество 0,0462
20	Авто 0,9549	Технологии 0,9729	Культура 0,9006	Бизнес 0,7938	Общество 0,3143
	Бизнес 0,0182	Бизнес 0,0159	Спорт 0,0325	Финансы 0,0937	Культура 0,1970
	Общество 0,0156	Политика 0,0036	Технологии 0,0160	Общество 0,0746	Политика 0,1512

с помощью статистической меры TF-IDF и модели BiLSTM. Метод реализован по следующей схеме: очистка текста, токенизация, удаление стоп-слов, лемматизация, выделение наиболее информативных токенов по TF-IDF, преобразование текстов в последовательности равной длины и обучение модели нейронной сети.

Эксперименты показали, что качество классификации существенно зависит от параметра n , определяющего количество подаваемых на вход нейросети слов: при увеличении числа отобранных слов меняется баланс между сохранением контекста и добавлением шума, что отражается на итоговых метриках. Проверка работы на текстах, не входящих в исходный датасет, показала работоспособность модели на новых текстах и наличие близких по темам категорий, на которых происходят ошибки. Полученные результаты показывают, что предложенный метод может использоваться для практической классификации новостных статей, например, при работе агрегатора новостей или автоматической категоризации новых статей.

Список литературы

- [1] Brownlee, J. Long Short-Term Memory Networks With Python / J. Brownlee. — Vermont : Machine Learning Mastery, 2017. — 229 p. — URL: <https://machinelearningmastery.com/lstms-with-python/>.
- [2] Feature extraction // Scikit-learn : [сайт]. — URL: https://scikit-learn.org/stable/modules/feature_extraction.html (дата обращения: 05.05.2025). — Загл. с титул. экрана.
- [3] IlyaGusev/gazeta // Hugging Face : [сайт]. — URL: <https://huggingface.co/datasets/IlyaGusev/gazeta> (дата обращения: 17.12.2025). — Загл. с титул. экрана.

Библиографическая ссылка

Эквист, В. Р. Создание гибридного метода классификации текстов на основе меры TF-IDF и рекуррентных нейронных сетей // Студенческая конференция факультета ПМИК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 235–242.

Сведения об авторах

Эквист Вячеслав Романович
Студент магистратуры
Направление «Прикладная информатика»

УДК 004.852, 519.237.5, 631.559

Построение нейросетевых моделей для прогнозирования урожайности

Яковлев А. М.

Тверской государственный университет
Рекомендовано к публикации доцентом кафедры
математической статистики и системного анализа С. В. Архиповым

Аннотация. В статье рассматривается задача построения нейросетевых моделей для прогнозирования урожайности. Описываются принципы построения нейронных сетей и их теоретические основы. Приводится краткий обзор языка R, его возможностей и пакетов, используемых для решения поставленной задачи. Представленные результаты работы программы на данных из базы исследования. **Ключевые слова:** R, neuralnet, урожайность, прогнозирование, нейронные сети, линейная регрессия.

Введение

Сельское хозяйство играет многогранную роль в обеспечении стабильности и устойчивого развития общества, а изменение природно-климатических характеристик отражается на производстве сельскохозяйственной продукции. В связи с этим особая роль отводится прогнозам урожайности различных культур, которые позволяют принимать обоснованные управленческие решения, минимизировать потери и повышать экономическую эффективность земледелия. Используемые в оперативной практике методы прогнозов с течением времени «устаревают» и перестают удовлетворять современным требованиям к качеству, возможностям и заблаговременности прогнозов. Для поддержания должного уровня прогнозов необходимо разрабатывать новые методы и технологии прогнозирования урожайности.

Изучение нейросетевых моделей сейчас активно набирает обороты во всех сферах, например, экономика, автоматизация в производстве, робототехнике, медицине и т. д. Они способны обрабатывать огромные объёмы информации, выявлять закономерности и делать предсказания. Использование нейронных сетей открывает новые

возможности и позволяет рассчитывать на результаты высокой точности.

Целью данного исследования является реализация программы построения нейросетевых моделей для прогнозирования урожайности. Объектом исследования выступают данные за определённый год с полей содержащие различные факторы, влияющие на урожайность. Предмет исследования — нейросетевые модели и их способность к предсказыванию переменной.

1. Постановка задачи

Задача прогнозирования урожайности заключается в предсказании её значения с помощью независимых переменных. Иными словами, это задача регрессионного анализа [3].

Регрессионный анализ изучает связь между зависимой переменной и одной (или несколькими) независимыми переменными. Подобная связь представляется в виде математической модели.

Зависимая (объясняемая, результирующая, эндогенная) переменная (признак) y — отражает результат деятельности системы. Воздействие других переменных и факторов напрямую влияет на её значение, поэтому результирующая переменная является случайной величиной.

Объясняющие (независимые, предикторные, экзогенные) переменные $x = (x_1, x_2, \dots, x_k)$ — переменные (признаки), которые описывают условия системы и в существенной мере определяют значения объясняемой переменной. В нашей модели были использованы случайные независимые переменные.

В рамках данного исследования мы будем придерживаться модели множественной линейной регрессии. Она имеет следующий вид:

$$y = a + b_1x_1 + b_2x_2 + \dots + b_kx_k + \varepsilon,$$

где

- a — константа,
- b_1, b_2, \dots, b_k — коэффициенты (параметры) модели,
- k — количество факторов модели,

- ε — случайная ошибка.

Необходимо найти такие параметры модели, с помощью которых можно получить минимальный средний квадрат ошибки модели. Чтобы достичь этой цели, применим нейронные сети.

2. Введение в нейронные сети

Термин «нейронные сети» относится к системе нейронов, как органических, так и искусственных по своему происхождению. В смысле искусственного интеллекта нейронные сети представляют собой набор алгоритмов, которые предназначены для идентификации закономерностей подобно тому, как это делает человеческий мозг [1]. Они интерпретируют сенсорные данные с помощью машинного восприятия, маркировки или кластеризации исходных данных. Распознавание осуществляется в числовом формате, который хранится в векторах. В эти вектора могут быть преобразованы реальные данные: изображения, звук, текст или временные ряды. Нейронную сеть можно представить как систему, состоящую из множества тесно взаимосвязанных узлов, называемых «нейронами», которые организованы в слои и обрабатывают информацию, используя динамические реакции состояния на внешние входные данные.

Для решения задач регрессионного анализа применяется тип искусственной нейронной сети под названием многослойный перцептрон.

Многослойный перцептрон — популярный тип нейронной сети прямого распространения, которая состоит из нескольких слоёв нейронов: входного, скрытого (или скрытых) и выходного. Входной слой принимает данные, которые последовательно обрабатываются через скрытые слои. Скрытые слои выполняют нелинейные преобразования входных данных, что позволяет сети решать сложные задачи, недоступные для линейных моделей.

Процесс обучения многослойного перцептрона основан на методе обратного распространения ошибки. Метод обратного распространения — это метод обучения нейронных сетей с учителем, где основная идея заключается в минимизации функции ошибки путём корректировки весов сети в направлении, противоположном градиенту этой функции. Без этого метода глубокие нейронные сети оставались

бы теоретическим конструктом без практической ценности. Метод обратного распространения ошибки работает циклически:

- сигнал проходит через все слои, формируется предсказание;
- рассчитывается ошибка на выходе, сравнивается с целевым значением;
- ошибка распространяется обратно для корректировки весов группировки;

Этот механизм позволяет сети постепенно улучшать предсказания.

3. Описание задачи

Рассматривается следующая модель:

$$Y = a + b_1 \cdot Height + b_2 \cdot Radiation + b_3 \cdot Curvature + \\ + b_4 \cdot Steepness + b_5 \cdot pH + b_6 \cdot P + b_7 \cdot K + b_8 \cdot Humus + \varepsilon,$$

где

- a — константа,
- b_1, b_2, \dots, b_8 — коэффициенты (параметры) модели,
- $Height, Curvature, Steepness$ — значения высоты, кривизны и крутизны участка соответственно,
- $Radiation$ — значение солнечной радиации,
- pH — кислотность почвы,
- $P, K, Humus$ — содержания фосфора, калия и гумуса в почве,
- Y — урожайность,
- ε — случайная ошибка.

Существенную роль в решении поставленной задачи играет подход кластеризации. Кластеризация — это метод машинного обучения без учителя, задача которого — разбить множество объектов на группы (кластеры) так, чтобы объекты внутри одной группы были максимально похожи друг на друга, а объекты из разных групп — максимально отличались.

В работе используются кластеры, полученные в результате разбиения данных долговременного (1998–2024 гг.) мониторинга урожайности многолетних трав (сено) в пределах полигона, расположенного к востоку от г. Твери [2]. Все года были разделены на четыре периода, различающихся по степени нагрева и пропитывания влагой территории.

- 1) Кластер 1. В него вошли года с максимальной урожайностью.
- 2) Кластер 2. Условия летнего прогрева территории в года, которые попали в эту группу, практически не отличаются от первого, но соотношение тепла и влаги, а также суммы осадков, заметно выше.
- 3) Кластер 3. Годы, относящиеся к третьему кластеру, характеризуются прохладным летом и сопоставимо влажными весной и осенью, что способствует значительному снижению урожайности трав и увеличению её пространственной пестроты.
- 4) Кластер 4. Объединил сухие и жаркие года (среднее количество жарких дней в году при этом травы в эти временные промежутки производили значительную биомассу).

Главная идея задачи в том, чтобы вычислить параметры модели для каждого из кластеров на основе данных за 1998–2024 годы и использовать эти модели для прогнозирования урожайности входных данных, для которых первым делом определяется принадлежность к одному из обозначенных выше кластеров. Чтобы отнести данные к тому или иному кластеру мы будем использовать многомерный критерий Т-квадрат Хотеллинга. Многомерный критерий Т-квадрат Хотеллинга используется для определения, значительно ли отличаются друг от друга средние значения двух групп по нескольким зависимым переменным. Он расширяет концепцию t-критерия Стьюдента на более высокие измерения.

Статистика теста вычисляется с использованием средних значений двух групп, объединённой матрицы ковариации и размеров выборки. Формула для статистики Т-квадрат имеет вид

$$T^2 = \frac{n_1 n_2}{n_1 + n_2} (\bar{X}_1 - \bar{X}_2)' S_p^{(-1)} (\bar{X}_1 - \bar{X}_2),$$

где

- \bar{X}_1 и \bar{X}_2 — вектора средних значений выборок X_1 и X_2 ,
- n_1 и n_2 — размеры выборок,
- S_p — объединённая матрица ковариации.

Для достоверных результатов теста необходимо соблюдение нескольких предположений:

- данные должны быть многомерно нормально распределены;
- выборки должны быть независимы друг от друга;
- ковариационные матрицы сравниваемых групп должны быть равны.

После того, как была проведена оценка отличия средних каждого кластера от значения средних входных данных, определяется наименьшее значение из них. Таким образом и устанавливается принадлежность к конкретному кластеру. Следующим шагом является использование соответствующей модели для расчёта прогноза урожайности. Точность предсказания оценивается с помощью среднеквадратичной ошибки, известной также как RMSE. Формула вычисления среднеквадратичной ошибки выглядит так:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2},$$

где y_i — реальные значения, а \hat{y}_i — предсказанные.

4. Выбор и обоснование языков программирования и технологий

Для решения поставленной задачи работе была реализована программа, написанная на языке R в среде разработки RStudio с использованием пакетов `neuralnet`, `NeuralNetTools`, `caTools`, `MASS`, `Hotelling` и `tools`.

R это язык программирования с открытым исходным кодом, специализирующийся на статистических вычислениях, анализе данных и их визуализации. Он свободно распространяемый, широко используется аналитиками, исследователями и аналитиками данных благодаря огромному множеству готовых пакетов и мощным графическим возможностям. В частности, в работе были использованы следующие технологии:

- `neuralnet` — пакет для создания и обучения искусственных нейронных сетей, в частности, многослойных перцептронов. Используется для решения задач регрессии и классификации путём моделирования сложных, нелинейных зависимостей в данных.
- `NeuralNetTools` — это пакет в языке программирования R, предназначенный для визуализации и анализа нейронных сетей. Содержит инструменты для построения графиков, количественной оценки важности переменных, проведения анализа чувствительности и получение простого списка весовых коэффициентов модели.
- `caTools` — пакет, предназначенный для упрощения различных задач анализа данных. Предоставляет набор инструментов для статистических расчетов, обработки данных и визуализации, активно используется в сфере машинного обучения и анализе временных рядов.
- `MASS` — этот пакет включает в себя множество полезных функций и примеров данных, в том числе функции для оценки линейных моделей с помощью обобщённого метода наименьших квадратов (GLS), подгонки линейных моделей с отрицательным биномиальным распределением, робастной подгонки

линейных моделей и неметрического многомерного шкалирования Крускала.

- **Hotelling** — пакет, предоставляющий удобные функции для проведения многомерного статистического анализа. В частности, для выполнения двухвыборочного Т-квадрат теста Хотеллинга, который проверяет различие между векторами средних двух групп по нескольким зависимым переменным одновременно.
- **tools** — пакет, содержащий основные утилиты для разработки, администрирования и документирования пакетов R. Включает в себя функции для проверки целостности пакетов, создания документации (файлов Rd) и работы с метаданными пакетов.

5. Описание алгоритма программы

Входные/исходные данные

- Таблица (-ы) формата CSV с значениями описанных ранее параметров модели за конкретный год.
- Таблицы кластеров формата CSV с значениями тех же параметров.

Выходной результат

- Таблица, в которой приводятся результаты расчёты трёх видов модели для каждого из кластеров с отражением параметров и ошибки.
- Значения прогноза урожайности.
- Среднеквадратичная ошибка.
- Распределение нейронов по слоям.
- Значения статистик двухвыборочного Т-квадрат теста Хотеллинга.

Алгоритм программы

- 1) Считываются таблица с данными и таблицы кластеров. Таблиц с данными может быть несколько.
- 2) Для каждой таблицы кластеров используется нормализация min-max, которая преобразует данные в общий диапазон, таким образом устраняя эффект масштабирования всех переменных.
- 3) Данные каждого из кластеров разделяются на обучающую выборку и тестовую выборку. Обучающая выборка используется для определения взаимосвязи между зависимыми и независимыми переменными, в то время как тестовая выборка анализирует производительность модели. Мы используем 70 % набора данных в качестве обучающего набора. Распределение данных по обучающему и тестовому наборам выполняется с использованием случайной выборки, которая генерируется с помощью функции языка R.
- 4) Выбирается не только лучший алгоритм вычисления нейронной сети, но и подбирается оптимальное количество нейронов на одном, двух или трёх слоях, которые обеспечивают наименьшую RMSE.
- 5) Вычисляются значения статистики двухвыборочного Т-квадрат теста Хотеллинга для определения принадлежности данных к тому или иному кластеру.
- 6) Определяется наименьшее значение из тех, что были получены на предыдущем шаге.
- 7) Вычисляется прогноз урожайности с использованием нейронной сети, полученной на шаге 4 и соответствующей кластеру, средние значения которого минимально отличаются от средних значений входных данных.
- 8) Результаты работы выводятся в консоль.

6. Результаты испытаний

Взглянем на таблицы для кластеров. В них отражается поведение модели при различном числе скрытых слоёв и нейронов на них.

neurons		RMSE	Algorithm
1	6	0.611267693907259	sag
2	4, 5	0.563265289312648	slr
3	7, 6, 5	0.536961077975029	sag

neurons		RMSE	Algorithm
1	8	0.566164704202423	rprop-
2	8, 9	0.541721514935711	rprop+
3	7, 4, 4	0.530722498399975	rprop+

neurons		RMSE	Algorithm
1	8	0.332703699169177	slr
2	8, 9	0.364621479137158	rprop-
3	9, 4, 2	0.365572732157736	sag

neurons		RMSE	Algorithm
1	8	0.54150603510732	sag
2	6, 2	0.49180907281421	slr
3	6, 4, 3	0.49608998893091	slr

Рис. 1. Таблицы для кластеров 1, 2, 3 и 4 соответственно.

В качестве примера взяты результаты работы программы на данных с поля номер 1 за 2022 год.

Рассчитанные значения статистик двухвыборочного Т-квадрат теста Хотеллинга следующие:

$$1,012982 \cdot 10^{-7}; 2,927100 \cdot 10^{-7}; 8,780685 \cdot 10^{-8}; 7,655867 \cdot 10^{-8}.$$

$7,655867 \cdot 10^{-8}$ является наименьшим, следовательно исходные данные можно отнести к кластеру 4, в который попали сухие и жаркие года. Вторая модель в четвёртой таблице на рис. 1 соответствует этому кластеру. Был сделан прогноз:

26,03996; 25,26021; 24,18627; 26,42412; 26,41805; 30,54236;
 25,69549; 41,22857; 46,68086; 48,25304; 50,37938; 13,86855;
 18,55075; 12,98441; 12,76177; 18,4424; 13,08492; 12,50383;
 12,6179; 14,91427; 16,04348; 23,67351; 27,22612; 29,18092;
 28,92055; 29,64653; 31,26201; 29,76253; 30,71053; 29,5229;
 26,57687; 29,24043; 29,70402; 29,94514; 32,64909; 30,20666.

На рис. 2 далее можно увидеть: входной слой, два скрытых слоя и выходной слой. B_1, \dots, B_3 — это узлы смещения (bias), олицетво-

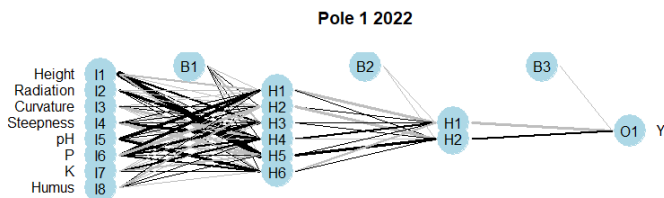


Рис. 2. График нейронной сети для данных поля номер 1 за 2022 год.

ряющие константу в уравнении регрессии. $H1, \dots, H6$ — нейроны на скрытых (hidden) слоях. На первом скрытом слое 6 — нейронов, на втором — 2.

Значение средней квадратичной ошибки равно 6,105544, а лучшим алгоритмом для построения нейронной сети оказался slr (модифицированная глобально конвергентная версия метода обратного распространения ошибки).

Заклучение

Рассмотренный в статье подход по прогнозированию урожайности с использованием кластеров и двухвыборочного Т-квадрат теста Хотеллинга был реализован с помощью средств языка R. Данный подход позволяет существенно экономить время при поступлении на вход больших (-ой) таблиц (-ы) данных за счёт вычисления всего лишь четырёх оптимальных моделей и дальнейшего выбора подходящей модели.

Список литературы

[1] Аггарвал, Ч. Нейронные сети и глубокое обучение : учебный курс : Пер. с англ. — СПб. : ООО «Диалектика», 2020. — 752 с. —

URL: http://www.physics.gov.az/book_N/neyronnyie_seti_C_Aggarval.pdf.

- [2] Иванов, Д. А. Сравнение математических алгоритмов при исследовании продукционного процесса многолетних трав // Д. А. Иванов, С. В. Архипов, К. С. Курпас // Агрофизика. — 2025. — № 4. — С. 50–59. — DOI: <https://doi.org/10.25695/AGRPH.2025.04.08>.
- [3] Максимова, Т. Г. Эконометрика : учебно-методическое пособие / Т. Г. Максимова, И. Н. Попова. — СПб. : Университет ИТМО, 2018. — 70 с. — URL: https://books.ifmo.ru/book/2210/ekonometrika:_uchebno-metodicheskoe_posobie.htm.

Библиографическая ссылка

Яковлев, А. М. Построение нейросетевых моделей для прогнозирования урожайности // Студенческая конференция факультета ПМиК. Сборник трудов. — Тверь : ТвГУ, 2026. — С. 243–254.

Сведения об авторах

ЯКОВЛЕВ АНТОН МАКСИМОВИЧ

Студент магистратуры

Направление «Прикладная математика и информатика»

Научное издание

Студенческая конференция
факультета ПМиК.
Сборник трудов

Тверь
13-30 апреля 2026 г.

Под редакцией С. М. Дудакова и Б. Н. Карлова

Подписано к использованию 26.05.2026. Формат 60x84 1/16*.

Усл. печ. л. 14,82. Заказ № 93.

Печать по требованию.

Издательство Тверского государственного университета.

Адрес: 170100, г. Тверь, Студенческий пер. 12, корпус Б.

Тел. (4822) 35-60-63.