

DSPA:

Вопросы применения цифровой обработки сигналов

№1

2026

СОДЕРЖАНИЕ

Вельмакина П.В., Скородумова Е.А. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ВНЕШНИХ МЕТОДОВ ОЦЕНКИ ВАЖНОСТИ ПРИЗНАКОВ В МУЛЬТИЛЕЙБЛОВОЙ КЛАССИФИКАЦИИ ТЕХНИЧЕСКИХ ОТКАЗОВ	4
Синева И.С. МАТЕМАТИЧЕСКИЕ МОДЕЛИ И АЛГОРИТМЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ГЕНЕРАЦИИ И РАСПРОСТРАНЕНИИ ПРОПАГАНДЫ: АНАЛИЗ, ДЕТЕКЦИЯ И ЭТИЧЕСКИЕ ИМПЛИКАЦИИ	14
Джозеф Сенеси Лавали, Крейнделин В.Б. АНАЛИЗ НЕДОСТАТКОВ ТЕХНОЛОГИЙ ОБНАРУЖЕНИЯ И РАССЛЕДОВАНИЯ ИНЦИДЕНТОВ БЕЗОПАСНОСТИ	24
Ерохин А.Г., Ванина М.Ф., Фролова Е.А. РУТНОН-ОРИЕНТИРОВАННЫЙ ПОДХОД К РЕАЛИЗАЦИИ АНАЛИТИЧЕСКИХ ПРОЦЕССОВ ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ МАРКЕТИНГОВЫХ КАМПАНИЙ	30
Гадасин Д.В., Родина А.А., Яковенко Н.В., Сурова М.А. СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ МЕЖДУ ВЫЧИСЛИТЕЛЬНЫМИ УСТРОЙСТВАМИ	46
Тимощенко П.А., Косичкина Т.П. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПРОТОКОЛОВ ПЕРЕДАЧИ ДАННЫХ ДЛЯ СИСТЕМ УМНОГО ДОМА НА БАЗЕ МИКРОКОНТРОЛЛЕРА ESP-32	55

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ВНЕШНИХ МЕТОДОВ ОЦЕНКИ ВАЖНОСТИ ПРИЗНАКОВ В МУЛЬТИЛЕЙБЛОВОЙ КЛАССИФИКАЦИИ ТЕХНИЧЕСКИХ ОТКАЗОВ

Вельмакина Полина Владимировна

студент МТУСИ, Москва, Россия,

p.v.velmakina@edu.mtuci.ru

Скородумова Елена Александровна

доцент кафедры ТВиПМ МТУСИ, к.ф.-м.н., доцент, Москва, Россия,

eas@mtuci.ru

Аннотация

В работе проводится сравнительный анализ таких внешних (модельно-агностических) методов оценки важности признаков, как Permutation Importance (PI), Kernel SHAP и LIME, в задаче мультилейбловой классификации технических отказов. В качестве образцов для сравнения используются внутренние меры важности, предоставляемые интерпретируемыми моделями. Результаты показывают высокую согласованность внешних методов с внутренними оценками. Особое внимание уделено поведению методов в условиях низкого качества модели (случайный отказ). Полученные выводы подтверждают обоснованность применения модельно-агностических методов в диагностических системах.

Ключевые слова

интерпретируемость моделей, важность признаков, Permutation Importance, Kernel SHAP, LIME, модельно-агностические методы, техническая диагностика, машинное обучение.

Введение

В условиях роста сложности технических систем и повышения требований к надёжности их эксплуатации задачи диагностики и классификации отказов или состояний системы приобретают всё большее значение. Современные подходы к решению таких задач всё чаще опираются на методы машинного обучения, в том числе на модели высокой сложности, например, ансамблевые алгоритмы или нейронные сети. Эти модели способны выявлять сложные нелинейные зависимости и взаимодействия признаков, но их ключевым недостатком остаётся низкая интерпретируемость. В прикладных областях, где от диагностики зависят безопасность, стоимость обслуживания и устойчивость производства, пользователю важно не только знать, что произойдёт отказ, но и понимать, почему модель сделала такое предсказание, особенно в задачах мультилейбловой классификации, где связь между метками имеет физический или логический смысл.

В данной работе исследуются внешние методы оценки важности признаков, поскольку они представляют собой универсальный инструмент интерпретации, применимый к широкому классу моделей. В отличие от внутренних мер, которые зависят от конкретной архитектуры и могут отсутствовать вовсе, внешние методы опираются исключительно на поведение модели как «чёрного ящика». Это подчёркивает их универсальность в прикладных задачах, где требуется единый подход к интерпретации различных моделей, а также в ситуациях, когда структура модели не предоставляет интерпретируемых внутренних характеристик. Также, всё ещё остаётся неясным, насколько достоверно эти оценки отражают реальный вклад признаков в принятие решений модели. Даже если в некоторых технических системах существуют априорные представления о механизмах отказов, интерпретация модели может не подтверждать заранее известных зависимостей.

Таким образом, сравнение внешних методов оценки важности признаков остаётся актуальной задачей.

1. История исследования модельно-агностических методов

Одним из ключевых подходов к интерпретации стал модельно-агностический анализ важности признаков, позволяющий оценивать вклад отдельных переменных независимо от внутренней структуры модели. В работе [1] был предложен метод LIME (Local Interpretable Model-agnostic Explanations), основанный на предположении, что поведение любой сложной модели аппроксимируется локальной линейной зависимостью в окрестности отдельного наблюдения. Этот подход стал основой для многих последующих исследований.

Дальнейшее развитие идеи модельно-агностической интерпретации получило в работе [2], где был представлен метод SHAP (SHapley Additive exPlanations). Его концептуальная новизна заключается в строгом теоретическом обосновании, а важность признаков выводится из теории кооперативных игр [3], каждый признак рассматривается как игрок, вносящий свой вклад в общий выигрыш - в предсказание модели.

Параллельно с этим развивались альтернативные подходы, основанные на прямом измерении влияния признака на качество предсказания через нарушение его связи с целевой переменной. Ключевым из них является пермутационная важность (Permutation Importance, PI), формально определённая в работе [4], и практическая реализация которой описана в [5]. PI определяется как разность между базовым значением метрики качества и её значением после случайной перестановки признака на независимой выборке.

2. Постановка задачи

В данной работе рассматривается задача мультитейбловой классификации технических отказов на основе синтетического датасета, сгенерированного в соответствии с физическими законами, описывающими работу оборудования. Цель исследования – оценка согласованности модельно-агностических (внешних) методов определения важности признаков, а именно Permutation Importance (PI), Kernel SHAP и LIME путём их сопоставления с внутренними мерами важности, предоставляемыми интерпретируемыми моделями, такими как коэффициенты в логистической регрессии (Logistic regression, LR) [6], веса в методе опорных векторов (Support vector machine, SVM) с линейным ядром [7] и среднее уменьшение неопределённости (Mean Decrease in Impurity, MDI), с использованием критерия Джини [4] в модели случайного леса (Random Forest, RF) [8].

Поскольку в используемых моделях внутренние оценки важности имеют чёткую математическую интерпретацию, они служат основой для сравнения, и гипотеза состоит в том, что внешние методы, если они корректны, должны воспроизводить ранжирование признаков, близкое к внутренним мерам. Подтверждение этой гипотезы позволяет обосновать применение тех же внешних методов к более сложным, но всё ещё интерпретируемым моделям, для которых внутренние оценки либо отсутствуют, либо ненадёжны.

Выбор указанных внешних методов обусловлен их популярностью и различием в подходах к интерпретации, что позволяет охватить разные аспекты модельно-агностической интерпретируемости.

– Permutation Importance представляет собой прямой и интуитивно понятный подход, основанный на измерении падения качества модели при разрушении связи между признаком и целевой переменной через случайную перестановку.

– LIME реализует локальную линейную аппроксимацию, позволяя объяснять отдельные предсказания даже для сложных моделей и тем самым фокусируясь на поведении модели в окрестности конкретного наблюдения.

— SHAP обобщает ряд существующих методов и обеспечивает свойства, которые важны для достоверности результатов оценки важности признаков, что делает его одним из наиболее современных и строгих подходов к интерпретации предсказаний. Выбор Kernel SHAP в данном исследовании обусловлен его вычислительной эффективностью.

Выбор задачи мультитейбловой классификации связан с особенностями реальной технической диагностики, где отказы редко возникают изолированно и чаще проявляются одновременно несколько типов неисправностей. При этом каждый тип отказа (метка) может зависеть от разных подмножеств признаков. Мультитейбловый подход позволяет учитывать такие сложные режимы работы и сохранять физическую осмысленность каждой метки.

Таким образом, задача работы – анализ согласованности и достоверности методов интерпретации для формирования обоснованного доверия к объяснениям моделей машинного обучения критически важных состояний.

3. Анализ существующих методов оценки важности признаков

Одним из наиболее популярных способов оценки важности признаков является пермутационный метод (Permutation Importance, PI) [4].

Формально представим обучающую выборку в виде матрицы X размера $N \times d$, где каждая строка $x_i \in R^d$ $x_i \in R^d$ соответствует одному наблюдению, $i = 1, \dots, N$. Обозначим через $X^{\pi, j}$ $X^{\pi, j}$ матрицу, полученную путём перестановки значений в j -м столбце в соответствии с некоторым механизмом π . Пусть $l(y_i, f(x_i))$ $l(y_i, f(x_i))$ – значение функции потерь, возникающее при предсказании $f(x_i)$ $f(x_i)$ вместо истинного значения y_i y_i . Тогда важность j -го признака определяется как:

$$PI_j^\pi = \sum_{i=1}^N l(y_i, f(x_i^{\pi, j})) - l(y_i, f(x_i)),$$

то есть как прирост ошибки предсказания, вызванный перестановкой значений j -го признака.

В практической реализации [5] пермутационная важность вычисляется следующим образом: сначала на валидационной или тестовой выборке оценивается базовое значение метрики качества, задаваемой заранее. Затем значения одного из признаков случайно перемешиваются, и метрика пересчитывается. Важность признака определяется как разность между исходным значением метрики и значением после перестановки. Такой подход гарантирует, что оценка отражает чувствительность модели именно к нарушению информативной структуры признака, а не к эффектам переобучения.

Существуют различные варианты этого подхода: можно использовать разные схемы перестановки π или использовать различные выборки. В рамках алгоритма случайного леса в работе [8] переставляют значения j -й переменной только в out-of-bag (OOB) выборке для каждого дерева, а итоговую оценку важности получают усреднением по всем деревьям леса.

В отличие от одномерных методов отбора признаков (например, анализ таблиц сопряжённости), которые оценивают связь каждого признака с целевой переменной изолированно и игнорируют взаимодействия, пермутационная важность учитывает вклад признака в предсказательную способность модели как по отдельности, так и в комбинации с другими признаками. Это позволяет выявлять признаки, незначимые в одиночку, но важные в совокупности [9].

Однако пермутационный подход чрезмерно акцентирует внимание на поведении модели в тех областях пространства признаков, где объём данных крайне мал [4]. То есть при перестановке признака модель может получить на вход нереалистичные комбинации значений, которых почти нет в обучающих данных.

Метод Local Interpretable Model-agnostic Explanations (LIME), предложенный в работе [1], представляет собой подход к построению локальных интерпретаций предсказаний произвольных моделей машинного обучения. Основная цель метода — представить объяснения, которые одновременно являются интерпретируемыми, локально точными (faithful) и независимыми от конкретной архитектуры модели.

Пусть $x \in R^d$ $x \in R^d$ обозначает входное наблюдение, а $f: R^d \rightarrow [0,1]$ $f: R^d \rightarrow [0,1]$ — классификатор, результатом работы которого является предсказание $f(x)$ $f(x)$ вероятности принадлежности наблюдения к целевому классу. Цель метода состоит в построении интерпретируемой модели g g , принадлежащей некоторому семейству простых моделей объяснений G , которые используют x' (понятное представление исходного x) в качестве признаков. Эта модель приближает поведение f в локальной окрестности x .

Объяснение $\xi(x)$ $\xi(x)$ формируется как решение следующей задачи:

$$\xi(x) = \arg \min_{g \in G} L(f, g, \Pi_x(z)) + \Omega_x(g),$$

где $\Pi_x(z)\Pi_x(z)$ – функция, измеряющая близость экземпляра z к исходному x , $\Omega_x(g)\Omega_x(g)$ – мера сложности объяснения g для x (контролируется с помощью заданного параметра K), $L(f,g,\Pi_x(z))L(f,g,\Pi_x(z))$ – мера несоответствия между оригинальной моделью f и интерпретируемой моделью g в окрестности, определенной $\Pi_x(z)$.

В работе [1] $L(f,g,\Pi_x(z))L(f,g,\Pi_x(z))$ — локально взвешенная квадратичная ошибка:

$$L(f,g,\Pi_x(z)) = \sum_{z \in R^d} \Pi_x(z) (f(z) - g(z))^2,$$

а $\Pi_x(z)\Pi_x(z)$ задаётся как экспоненциальное ядро от косинусного расстояния между x' и z' . Эта сумма аппроксимируется путём случайной выборки в окрестности x .

На практике объяснение строится в два этапа: сначала из всего множества признаков в x' отбираются не более чем K наиболее значимых, а затем на этих признаках обучается взвешенная линейная регрессия с весами $\Pi_x(z)\Pi_x(z)$, полученная модель g является локальным объяснением предсказания $f(x)$.

Метод SHAP (SHapley Additive exPlanations) [2] представляет собой унифицированный подход к интерпретации предсказаний моделей машинного обучения, основанный на теории кооперативных игр. В работе [2] утверждается, что SHAP определяет значения важности признаков как компоненты вектора Шепли, соответствующего кооперативной игре с характеристической функцией $E[f(z)|z_S]E[f(z)|z_S]$. Как и в классической теории кооперативных игр, где вектор Шепли распределяет общий выигрыш коалиции между агентами пропорционально их среднему маргинальному вкладу во все возможные коалиции [3], SHAP распределяет разницу между предсказанием модели $f(x)f(x)$ и ожидаемым предсказанием при отсутствии информации о признаках $E[f(z)]E[f(z)]$ пропорционально их среднему маргинальному вкладу в условное математическое ожидание $E[f(z)|z_S]E[f(z)|z_S]$. В обоих случаях вес вклада определяется вероятностью того, что данное подмножество других участников (признаков) уже присутствует в коалиции.

Значения Шепли условного математического ожидания исходной модели являются единственным решением в классе аддитивных методов распределения важности признаков, удовлетворяющим трём желательным свойствам: локальной точности, отсутствию влияния пропущенных признаков (missingness) и согласованности (consistency).

Локальная точность формально выражается как

$$f(x) = g(x') = \phi_0 + \sum_{i=1}^M \phi_i x'_i,$$

где f – исходная модель, g – модель, используемая для объяснения отдельного предсказания $f(x)f(x)$, $x' \in \{0,1\}^M$ $x' \in \{0,1\}^M$ — бинарное представление входа, в котором каждая компонента указывает, что используется i -й признак; а $\phi_0 = f(h_x(0))$ $\phi_0 = f(h_x(0))$ – предсказание модели при отсутствии всех признаков.

Свойство missingness требует, чтобы при $x'_i = 0$ $x'_i = 0$ соответствующий коэффициент удовлетворял условию:

$$x'_i = 0 \Rightarrow \phi_i = 0,$$

то есть признак, отсутствующий во входе, не вносит вклада в объяснение.

Согласованность (consistency) гарантирует, что если для двух моделей f и f' и всех $z' \in \{0,1\}^M$ $z' \in \{0,1\}^M$ выполняется

$$f'(h_x(z')) - f'(h_x(z' \setminus i)) \geq f(h_x(z')) - f(h_x(z' \setminus i)),$$

то соответствующие SHAP-значения удовлетворяют условию:

$$\phi_i(f', x) \geq \phi_i(f, x).$$

Здесь z' – бинарный вектор, задающий произвольное подмножество признаков: $z'_j = 1$ – j -й признак включён в подмножество, $z'_j = 0$ – исключён. Отображение $h_x(z')$ не задаётся в явном виде. Вместо этого значение модели на подмножестве z' определяется как условное математическое ожидание $E[f(z)|z_S]$, $S = \{j : z_j = 1\}$ $E[f(z) | z_S]$, $S = \{j : z_j = 1\}$.

Это свойство гарантирует, что если модель начинает сильнее реагировать на включение признака i при любом наборе остальных признаков, то его оценённая важность не должна уменьшаться, что исключает противоречивые объяснения.

SHAP-значения интерпретируются как вклад каждого признака в изменение математического ожидания предсказания модели при условии наблюдения этого признака. Формально SHAP-значения для i -го признака вычисляются как

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} [f_x(z') - f_x(z' \setminus i)],$$

где $f_x(z') = E[f(z)|z_S]$, S – множество индексов ненулевых компонент z' , а суммирование ведётся по всем подмножествам x' . M – общее число признаков, $|z'|$ – мощность множества наблюдаемых признаков в данном подмножестве, $\frac{|z'|!(M - |z'| - 1)!}{M!}$ задаёт вес маргинального вклада признака i , основанный на усреднении по всем перестановкам признаков. Это позволяет объяснить, как предсказание модели переходит от базового значения математического ожидания предсказаний по всем возможным входам $E[f(z)]$ к конкретному значению $f(x)$.

В статье [2] отмечается, что нахождение точных SHAP-значений является вычислительно трудоёмким, поскольку требует оценки модели на всех возможных подмножествах признаков. Для преодоления этого ограничения в [2] предлагаются модельно-агностические (не требуют знания внутренней структуры объясняемой модели) и модельно-специфические (используют информацию о типе модели) методы аппроксимации.

Среди модельно-агностических подходов можно выделить Kernel SHAP [2] и метод сэмплирования Шепли (Shapley sampling) [2]. Kernel SHAP представляет собой метод оценки значений Шепли на основе взвешенной линейной регрессии в пространстве упрощённых бинарных входов. В отличие от LIME, где параметры регрессии (ядро, функция потерь, регуляризация) выбираются эвристически, Kernel SHAP выводит их из теоретических требований удовлетворять трём свойствам, описанным ранее. В методе используется специальное весовое ядро Шепли, которое придаёт больший вес тем подмножествам признаков, которые содержат либо очень мало, либо почти все признаки. Благодаря совместной оценке всех коэффициентов ϕ_i в модели g , Kernel SHAP обеспечивает более высокую выборочную эффективность по сравнению с прямыми методами, такими как методы сэмплирования, требуя меньшего числа обращений к исходной модели для достижения сопоставимой точности. Таким образом, Kernel SHAP сохраняет общую структуру локальной линейной аппроксимации, но устраняет эвристические допущения LIME, обеспечивая теоретически обоснованное восстановление SHAP-значений.

К модельно-специфическим методам относятся Linear SHAP [2] и Deep SHAP [2]. В случае линейной модели и при допущении о независимости входных признаков Linear SHAP позволяет получить аналитическое выражение для SHAP-значений: базовое значение объяснения совпадает со свободным членом модели, а вклад каждого признака пропорционален его весу в модели, умноженному на отклонение значения признака от его математического ожидания. Deep SHAP использует композиционную природу глубоких нейронных сетей. Он рекурсивно вычисляет SHAP-значения для отдельных компонентов сети, таких как линейные слои, функции активации или операции максимизации, а затем объединяет их с помощью правила, аналогичного обратному распространению ошибки.

4. Результаты исследования

В данной работе используется синтетический датасет [10], моделирующий работу промышленного оборудования и содержащий 10 000 наблюдений, каждое из которых описывается 14 признаками. Данные сгенерированы на основе физически обоснованных зависимостей, отражающих реальные закономерности функционирования машины. Среди признаков: UID (уникальный идентификатор), Product ID (идентификатор продукта, кодирующий качество изделия — Low, Medium или High, и серийный номер), Air temperature [K] (температура воздуха), Process temperature [K] (температура процесса), Rotational speed [rpm] (частота вращения), Torque [Nm] (крутящий момент) и Tool wear [min] (время износа инструмента).

Режимы отказа определены следующим образом:

- Tool Wear Failure – TWF (износ инструмента) возникает при времени износа от 200 до 240 минут;
- Heat Dissipation Failure – HDF (нарушение теплового режима) происходит, если разность между температурой процесса и температурой воздуха менее 8.6 К и частота вращения ниже 1380 об/мин;
- Power Failure – PWF (сбой электропитания) возникает, если произведение крутящего момента и угловой скорости (в рад/с) оказывается вне диапазона [3500; 9000] Вт;
- Overstrain Failure – OSF (отказ по перегрузке) наступает, если произведение времени износа и крутящего момента превышает порог, зависящий от качества изделия: 11 000 мин·Н·м для Low, 12 000 для Medium и 13 000 для High;
- Random Failures – RNF (случайные отказы) с вероятностью 0.001 возникают независимо от значений технических параметров.

Обучение моделей выполнялось на подвыборке с подтверждёнными отказами. Такой выбор обусловлен постановкой задачи, в которой при известном факте отказа оборудования требуется определить его конкретный тип, что соответствует реальным инженерным сценариям, где система мониторинга уже зафиксировала отклонение, и следующим шагом является определение причины для выбора конкретного корректирующего действия.

Permutation Importance (PI) вычислялся с использованием F1-меры [11] в качестве целевой метрики, поскольку она учитывает полноту и точность предсказания, что важно в диагностических задачах, где одинаково значимы пропуск истинного отказа (низкая полнота) и ложное срабатывание (низкая точность).

В методе Kernel SHAP в качестве выборки использовалась случайная подвыборка 100 объектов обучающей выборки отказов, которая необходима для аппроксимации значений Шепли.

Метод LIME реализован через усреднение локальных линейных коэффициентов по всем объектам тестовой выборки отказов для получения глобальной оценки важности признаков. Для каждого наблюдения генерировался набор коэффициентов на основе 1000 синтетических примеров, после чего эти коэффициенты усреднялись по модулю.

Для всех методов признаки сортировались по убыванию абсолютной оценки важности. Сравнение внешних методов с внутренними мерами важности проводилось на основе совпадения порядка признаков, поскольку абсолютные значения важности несопоставимы между методами в силу различных шкал, принципов определения и целевых величин.

Анализ проводился отдельно для четырёх физически обоснованных типов отказов (TWF, HDF, PWF, OSF) – таблицы 1-3, и случайного отказа (RNF) – таблица 4. В случае RNF, сгенерированного независимо от признаков, идеальная модель должна присваивать всем признакам нулевую важность. Однако на практике модель может обнаруживать случайные корреляции и приписывать важность отдельным признакам. В этом случае внешние методы интерпретации считаются корректными, если они согласуются с внутренней оценкой модели, даже если последняя ошибочна. Поэтому анализ случайных отказов требует отдельного внимания.

Таблица 1

Важность признаков в моделях RF по типам отказов			
Random Forest			
TWF (Износ инструмента)			
MDI	Tool wear (0.323)	Rotational speed (0.273)	Torque (0.237)
PI	Torque (0.627)	Tool wear (0.414)	Rotational speed (0.394)
SHAP	Tool wear (0.098)	Rotational speed (0.066)	Torque (0.059)
LIME	Tool wear (0.111)	Torque (0.073)	Rotational speed (0.068)
HDF (Нарушение теплового режима)			
MDI	Air temperature (0.342)	Rotational speed (0.241)	Process temperature (0.169)
PI	Air temperature (0.390)	Rotational speed (0.268)	Tool wear (0.021)
SHAP	Air temperature (0.177)	Rotational speed (0.118)	Process temperature (0.055)
LIME	Air temperature (0.111)	Rotational speed (0.094)	Tool wear (0.022)
PWF (Сбой электропитания)			
MDI	Torque (0.384)	Rotational speed (0.247)	Tool wear (0.176)
PI	Rotational speed (0.296)	Torque (0.226)	Tool wear (0.120)
SHAP	Torque (0.125)	Tool wear (0.080)	Rotational speed (0.067)
LIME	Torque (0.131)	Tool wear (0.070)	Air temperature (0.032)
OSF (Перегрузка)			
MDI	Tool wear (0.458)	Torque (0.244)	Rotational speed (0.121)
PI	Tool wear (0.476)	Torque (0.266)	Rotational speed (0.046)
SHAP	Tool wear (0.219)	Torque (0.108)	Rotational speed (0.043)
LIME	Tool wear (0.129)	Torque (0.079)	Rotational speed (0.051)

Таблица 2

Важность признаков в моделях LR по типам отказов			
Logistic Regression			
TWF (Износ инструмента)			
Coef	Tool wear (2.544)	Torque (2.412)	Rotational speed (1.414)
PI	Torque (0.459)	Tool wear (0.453)	Air temperature (0.065)
SHAP	Torque (0.114)	Tool wear (0.102)	Rotational speed (0.055)
LIME	Tool wear (0.154)	Torque (0.146)	Rotational speed (0.086)
HDF (Нарушение теплового режима)			
Coef	Air temperature (3.916)	Rotational speed (2.770)	Process temperature (2.236)
PI	Air temperature (0.529)	Rotational speed (0.177)	Process temperature (0.172)
SHAP	Air temperature (0.265)	Process temperature (0.109)	Rotational speed (0.101)
LIME	Air temperature (0.268)	Rotational speed (0.189)	Process temperature (0.153)
PWF (Сбой электропитания)			
Coef	Rotational speed (4.362)	Torque (3.749)	Tool wear (0.742)
PI	Rotational speed (0.533)	Torque (0.428)	Tool wear (0.156)
SHAP	Torque (0.160)	Rotational speed (0.159)	Tool wear (0.066)
LIME	Rotational speed (0.301)	Torque (0.255)	Tool wear (0.051)
OSF (Перегрузка)			
Coef	Tool wear (3.942)	Torque (2.642)	Air temperature (1.126)
PI	Tool wear (0.476)	Torque (0.275)	Air temperature (0.176)
SHAP	Tool wear (0.230)	Torque (0.103)	Air temperature (0.080)
LIME	Tool wear (0.221)	Torque (0.148)	Air temperature (0.063)

Таблица 3

Важность признаков в моделях SVM по типам отказов

Support Vector Machine			
TWF (Износ инструмента)			
Coef	Torque (2.200)	Tool wear (1.497)	Rotational speed (1.494)
PI	Torque (0.556)	Tool wear (0.460)	Rotational speed (0.170)
SHAP	Torque (0.144)	Tool wear (0.087)	Rotational speed (0.080)
LIME	Torque (0.194)	Tool wear (0.132)	Rotational speed (0.132)
HDF (Нарушение теплового режима)			
Coef	Air temperature (3.442)	Rotational speed (2.654)	Process temperature (2.310)
PI	Air temperature (0.502)	Process temperature (0.163)	Rotational speed (0.148)
SHAP	Air temperature (0.293)	Process temperature (0.147)	Rotational speed (0.115)
LIME	Air temperature (0.272)	Rotational speed (0.208)	Process temperature (0.182)
PWF (Сбой электропитания)			
Coef	Rotational speed (4.089)	Torque (3.377)	Tool wear (0.532)
PI	Rotational speed (0.533)	Torque (0.438)	Tool wear (0.104)
SHAP	Torque (0.212)	Rotational speed (0.185)	Tool wear (0.063)
LIME	Rotational speed (0.327)	Torque (0.266)	Tool wear (0.042)
OSF (Перегрузка)			
Coef	Tool wear (3.565)	Torque (2.399)	Air temperature (0.869)
PI	Tool wear (0.509)	Torque (0.313)	Air temperature (0.179)
SHAP	Tool wear (0.237)	Torque (0.120)	Air temperature (0.084)
LIME	Tool wear (0.223)	Torque (0.150)	Air temperature (0.054)

Таблица 4

Важность признаков в случае случайного отказа (RNF)

Random Forest — RNF (Случайность)			
MDI	Rotational speed (0.272)	Tool wear (0.199)	Torque (0.188)
PI	Rotational speed (0.600)	Tool wear (0.367)	Air temperature (0.271)
SHAP	Tool wear (0.068)	Rotational speed (0.067)	Air temperature (0.029)
LIME	Rotational speed (0.083)	Tool wear (0.073)	Torque (0.058)
Logistic Regression — RNF(Случайность)			
Coef	Torque (1.327)	Rotational speed (1.149)	Air temperature (1.097)
PI	Process temperature (0.000)	Rotational speed (0.000)	Tool wear (0.000)
SHAP	Torque (0.064)	Air temperature (0.057)	Rotational speed (0.055)
LIME	Torque (0.103)	Rotational speed (0.089)	Air temperature (0.085)
Support Vector Machine — RNF (Случайность)			
Coef	Rotational speed (0.001)	Torque (0.001)	Air temperature (0.000)
PI	Air temperature (0.000)	Process temperature (0.000)	Rotational speed (0.000)
SHAP	Air temperature (0.000)	Rotational speed (0.000)	Torque (0.000)
LIME	Rotational speed (0.001)	Torque (0.000)	Air temperature (0.000)

Для количественной оценки согласованности использовались два критерия:

1. Совпадение топ-3 признаков (табл. 5): проверялось, содержатся ли три наиболее важных признака, выделенных внешним методом, в множестве трёх ведущих признаков по внутренней мере независимо от их порядка. Этот критерий отражает способность метода воспроизводить глобальную структуру влияния признаков, что актуально в задачах, где отказ определяется взаимодействием нескольких ключевых параметров.

2. Совпадение наиболее важного признака (табл. 6): оценивалось, совпадает ли признак, занимающий первое место в ранжировке внешнего метода, с ведущим признаком по внутренней оценке. Этот критерий имеет прикладное значение в системах, где правильная идентификация главной причины отказа критична для принятия управляющего решения.

Для каждого сочетания модели и метода объяснимости было подсчитано количество типов отказов (в диапазоне от 0 до 4), для которых наблюдалось полное совпадение множества признаков внутренних и внешних методов. Сумма количества по всем трём моделям отображена в строках «Итог» таблиц 5-6 и служит общей оценкой согласованности модельно-агностического метода.

Таблица 5

Итоговое количество совпадений топ-3 признаков

Топ 3-х признаков			
Модель	Permutation	SHAP	LIME
Random Forest	3	4	2
Логистическая регрессия	3	4	4
SVM	4	4	4
Итог	10	12	10

Таблица 6

Итоговое количество совпадений по первому признаку

Топ 1-х признаков			
Модель	Permutation	SHAP	LIME
Random Forest	2	4	4
Логистическая регрессия	3	2	4
SVM	4	3	4
Итог	9	9	12

5. Анализ результатов исследования

Проведённое исследование подтвердило гипотезу о том, что модельно-агностические методы оценки важности признаков Permutation Importance (PI), Kernel SHAP и LIME демонстрируют высокую согласованность с внутренними мерами важности. Как показано в таблице 5, основанной на сравнении множества трёх наиболее важных признаков, все методы достигли высокой согласованности: от 10 из 12 случаев (PI и LIME) до 12 из 12 (SHAP).

В отличие от результатов, полученных для других моделей, в случае Random Forest метод LIME демонстрирует сравнительно более низкую согласованность с внутренней мерой важности (MDI) – полное совпадение только двух из четырёх типов отказа (см. табл. 5). Random Forest строит предсказания на основе последовательных пороговых разбиений, и при сильной корреляции между признаками дерево, как правило, выбирает один из них для разбиения, концентрируя всю важность на этом признаке. LIME же, будучи методом локальной линейной аппроксимации, стремится распределить вклад между коррелированными признаками (см. табл. 1), что не соответствует реальной логике принятия решений в деревьях. Кроме того, его линейная модель не способна точно аппроксимировать резкие скачки в предсказаниях, возникающие на границах разбиений.

Однако, несмотря на эти ограничения, LIME во всех случаях корректно идентифицирует наиболее важный признак (см. табл. 6). В случае деревьев решений это связано с тем, что первые разбиения, как правило, выполняются по признаку с наибольшим вкладом (наибольшему уменьшению неопределённости), и этот глобальный сигнал оказывается достаточно сильным для корректной оценки линейной локальной аппроксимацией.

Стоит отметить согласованность результатов в условиях сильных комбинированных взаимодействий, заложенных в правилах генерации отказов, например, произведение Tool wear × Torque для OSF (см. табл. 1-3). Несмотря на то, что используемые модели (включая Random Forest с независимыми подмоделями) не учитывали зависимости между метками явно, внешние методы всё равно выделяли те же признаки, что и внутренние меры, что свидетельствует о достоверности этих методов даже в сложных, взаимозависимых сценариях.

Рассмотрим отдельно результаты, полученные для класса случайных отказов (RNF).

В логистической регрессии коэффициенты модели ненулевые, и SHAP с LIME, будучи локальными методами, отражают этот внутренний вклад, даже если он обусловлен шумом или переобучением на редких событиях. Permutation Importance, напротив, показывает нулевые значения, поскольку модель фактически не решает задачу классификации ($F1 \approx 0$), и перестановка признаков не ухудшает и без того минимальное качество.

В SVM все три метода PI, SHAP и LIME дают близкие к нулю оценки, но по разным причинам: SHAP и LIME воспроизводят практически нулевые коэффициенты линейной модели, в то время как PI указывает на отсутствие влияния признаков на качество предсказания, которое близко к случайному уровню.

В Random Forest, напротив, модель демонстрирует существенную способность предсказывать RNF ($F1 = 0.67$), что указывает на наличие ложных корреляций. Соответственно, все методы – MDI, PI, SHAP и LIME – выдают ненулевые значения важности, поскольку деревья действительно используют признаки для разбиений, и их перестановка ухудшает качество.

Заключение

Проведённое исследование подтвердило исходную гипотезу: модельно-агностические методы интерпретации демонстрируют высокую согласованность с внутренними мерами важности. Особенно выделяется Kernel SHAP, обеспечивший полное совпадение во всех случаях, что подтверждает его теоретическую обоснованность и устойчивость.

Анализ выявил специфику поведения методов в зависимости от архитектуры модели. В частности, LIME, несмотря на общую высокую согласованность, показал снижение точности в Random Forest. Тем не менее, во всех сценариях он корректно идентифицировал наиболее важный признак, что свидетельствует о его способности улавливать доминирующие сигналы.

Особую ценность представляет анализ случайного отказа (RNF), используемого для проверки корректности методов. В нём проявилось фундаментальное различие между методами:

- PI отражает практическую полезность признаков для предсказания;
- SHAP и LIME отражают внутреннюю структуру модели.

Это различие подчёркивает, что согласованность методов не гарантирует их физическую корректность, но подтверждает их соответствие поведению модели.

Важно отметить, что все методы успешно выявили физически значимые признаки даже в условиях сложных взаимодействий, несмотря на отсутствие явного моделирования зависимостей между метками. Это демонстрирует устойчивость подходов к интерпретации в реалистичных диагностических сценариях.

Полученные результаты обосновывают доверие к модельно-агностическим методам при анализе интерпретируемых моделей и открывают путь к их применению в более сложных системах, где внутренние меры важности недоступны.

Литература

1. Ribeiro M. T., Singh S., Guestrin C. Why Should I Trust You?: Explaining the Predictions of Any Classifier // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016, pp. 1135-1144.
2. Lundberg S. M., Lee S.-I. A Unified Approach to Interpreting Model Predictions // Advances in Neural Information Processing Systems 30 (NIPS 2017). 2017, pp. 4765-4774.
3. Маракулин В. М. Элементы теории кооперативных игр. Новосибирск: Институт математики им. С. Л. Соболева СО РАН; Новосибирский государственный университет, 2014.
4. Zhou Z., Hooker G. Unbiased Measurement of Feature Importance in Tree-Based Methods // arXiv preprint arXiv:1903.05179. 2020.
5. Molnar C. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. Leanpub, 2019.
6. James G., Witten D., Hastie T., Tibshirani R. An Introduction to Statistical Learning: with Applications in R. New York: Springer, 2013. (Springer Texts in Statistics), pp. 130-137.
7. Guyon I., Weston J., Barnhill S., Vapnik V. Gene Selection for Cancer Classification using Support Vector Machines // Machine Learning. 2002. Vol. 46, № 1-3, pp. 389-422.
8. Breiman L. Random Forests // Machine Learning. 2001. Vol. 45, № 1, pp. 5-32. DOI: 10.1023/A:1010933404324.
9. Strobl C., Boulesteix A.-L., Zeileis A., Hothorn T. Bias in random forest variable importance measures: Illustrations, sources and a solution // BMC Bioinformatics. 2007. Vol. 8, Art. 25. DOI: 10.1186/1471-2105-8-25.
10. Predictive Maintenance AI4I 2020 UCI [Электронный ресурс]. Режим доступа: <https://www.kaggle.com/datasets/abdulbasit551/predictive-maintenance-ai4i-2020-uci> (дата обращения: 01.01.2026).
11. Sasaki Y. The Truth of the F-measure. Manchester: School of Computer Science, University of Manchester, 2007. Version: 26 October 2007.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ И АЛГОРИТМЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ГЕНЕРАЦИИ И РАСПРОСТРАНЕНИИ ПРОПАГАНДЫ: АНАЛИЗ, ДЕТЕКЦИЯ И ЭТИЧЕСКИЕ ИМПЛИКАЦИИ

Синева Ирина Сергеевна

Московский технический университет связи и информатики

доцент, к.ф.-м.н., Москва, Россия

iss@mtuci.ru

Аннотация

Современные системы искусственного интеллекта (ИИ), особенно основанные на глубоком обучении и языковых моделях, демонстрируют беспрецедентную способность генерировать и распространять убедительный, контекстно-адаптированный текст. Эти технологии всё чаще используются не только в легитимных целях (маркетинг, информирование), но и в целях пропаганды – целенаправленного формирования общественного мнения посредством манипулятивного контента. В данной статье рассматриваются математические основы таких систем, включая модели языкового представления, механизмы персонализации и распространения, а также методы детекции пропагандистского контента. Особое внимание уделено формализации понятия «пропаганда» в терминах теории информации, байесовского вывода и теории игр. Предложены математические критерии выявления пропаганды на основе аномалий в распределении семантических и стилистических признаков. Обсуждаются этические и регуляторные аспекты, а также перспективы разработки «антипропагандистских» ИИ-систем.

Ключевые слова:

искусственный интеллект, пропаганда, языковые модели, анализ данных, теория информации, машинное обучение, детекция манипуляций.

1. Введение

Современные системы искусственного интеллекта (ИИ), особенно основанные на глубоком обучении и больших языковых моделях, демонстрируют беспрецедентную способность генерировать и распространять убедительный, контекстно-адаптированный текст. Эти технологии всё чаще используются не только в легитимных целях (маркетинг, информирование), но и в целях пропаганды – целенаправленного формирования общественного мнения посредством манипулятивного контента [1-11]. В данной статье рассматриваются математические основы таких систем, включая модели языкового представления, механизмы персонализации и распространения, а также методы детекции пропагандистского контента. Особое внимание уделено формализации понятия «пропаганда» в терминах теории информации, байесовского вывода и теории игр. Предложены математические критерии выявления пропаганды на основе аномалий в распределении семантических и стилистических признаков. Обсуждаются этические и регуляторные аспекты, а также перспективы разработки «антипропагандистских» ИИ-систем.

2. Формализация пропаганды: теоретико-информационный подход

Пусть \mathcal{M} – множество всех возможных сообщений (текстов, изображений, аудио и т.д.), а P_{true} – распределение вероятностей «объективного» контента, отражающего реальное состояние дел. Пропагандистский контент определяется как сообщение $m \in \mathcal{M}$, для которого распределение $P_{\text{prop}}(m)$ значительно смещено относительно $P_{\text{true}}(m)$. Степень искажения формализуется через дивергенцию Кульбака–Лейблера:

$$D_{\text{KL}}(P_{\text{prop}} \parallel P_{\text{true}}) = \int_{\mathcal{M}} P_{\text{prop}}(m) \log \frac{P_{\text{prop}}(m)}{P_{\text{true}}(m)} dm. \quad (1)$$

Этот подход расширяется в разделе 3.1 с использованием дивергенции Реньи для учёта экстремальных событий.

3. Математическое моделирование пропаганды с использованием ИИ

3.1. Теоретико-информационная модель пропаганды

Рассмотрим информационную среду как вероятностное пространство (Ω, F, P) , где P – распределение нейтрального контента. Пропагандистская активность вводит искажённое распределение Q . Для количественной оценки искажения используется дивергенция Реньи порядка $\alpha > 0$:

$$D_\alpha(Q\|P) = \frac{1}{\alpha-1} \log \int_\Omega \left(\frac{dQ}{dP} \right)^\alpha dP \quad (2)$$

При $\alpha \rightarrow 1$ выражение (2) сходится к дивергенции Кульбака–Лейблера (1), но при $\alpha > 1$ акцент смещается на редкие, но высоковлиятельные события — ключевой признак пропаганды [12].

Теорема 1 (Граница искажения пропаганды).

Пусть Q – распределение пропагандистского контента, а P – нейтральное распределение. Если $D_\alpha(Q\|P) > \tau$ для заданного порога $\tau > 0$, то вероятность классификации сообщения $m \sim Q$ как пропаганды алгоритмом с пороговым критерием стремится к 1 при $\alpha \rightarrow \infty$.

Доказательство. Следует из свойств дивергенции Реньи и неравенства Чернова [13]. При $\alpha \rightarrow \infty$ дивергенция D_α доминируется максимумом отношения плотностей $\frac{dQ}{dP}$, что гарантирует детекцию экстремальных отклонений.

3.2. Генеративные модели и их уязвимости

Современные большие языковые модели (LLM) обучаются максимизировать правдоподобие:

$$\mathcal{L}(\theta) = -\mathbb{E}_{x \sim \mathcal{D}} \left[\sum_{t=1}^T \log P_\theta(x_t | x_{<t}) \right]$$

Если обучающая выборка s содержит пропагандистские паттерны, модель воспроизводит их как «норму», игнорируя фактологическую истинность [14]. Для оценки смещения вводится коэффициент вероятностного отклонения (PDC – Circular Probable Deviation):

$$\text{PDC} = \frac{1}{N} \sum_{i=1}^N \left| \log P_\theta(m_i^{\text{fake}}) - \log P_\theta(m_i^{\text{true}}) \right| \quad (3)$$

Теорема 2 (Смещение в языковых моделях).

Для LLM, обученной на датасете s с долей пропагандистских примеров ρ , коэффициент PDC удовлетворяет неравенству $\text{PDC} \leq \rho \cdot C$, где $C > 0$ – константа, зависящая от архитектуры модели.

Доказательство. Основано на анализе смещения эмпирического риска в условиях несбалансированного обучающего выбора [14]. При $\rho \rightarrow 0$ (отсутствие пропаганды в обучении) PDC (3) стремится к минимальному значению, определяемому шумом в данных.

3.3. Модели персонализации и оптимизация манипуляции

Рекомендательные системы используют контекстную многоорукую бандитскую модель (CMAB – Combinatorial Multi-Armed Bandit) для максимизации вовлечённости:

$$\max_{\pi} \mathbb{E} \left[\sum_{t=1}^T r_t(a_t, c_t) \right]$$

Однако функция вознаграждения r_t , не учитывающая истинность контента, приводит к распространению дезинформации. С помощью обратного обучения с подкреплением (IRL – Inverse Reinforcement Learning) восстанавливается скрытая функция вознаграждения R^* :

$$R^* = \arg \max_R \mathbb{E}_{\pi_E} \left[\sum_t R(s_t, a_t) \right] - \mathbb{E}_{\pi} \left[\sum_t R(s_t, a_t) \right]$$

где π_E – экспертная политика [15-16]. Анализ R^* позволяет выявить скрытые пропагандистские цели.

3.4. Обнаружение пропаганды через аномалии в латентном пространстве

Пусть $f: \mathcal{M} \rightarrow R^d$ – энкодер (например, SBERT), генерирующий эмбединги $\mathbf{e}_m = f(m)$. Нейтральное подпространство \mathcal{N} определяется как выпуклая оболочка эмбедингов проверенных источников. Пропаганда характеризуется большим расстоянием до \mathcal{N} :

$$d_{\text{prop}}(m) = \min_{\mathbf{n} \in \mathcal{N}} \|\mathbf{e}_m - \mathbf{n}\|_2$$

Распределение нейтральных эмбедингов моделируется гауссовскими смесями [17]:

$$P_{\text{neutral}}(\mathbf{e}) = \sum_{k=1}^K \pi_k N(\mathbf{e}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

Теорема 3 (Детекция через латентное пространство).

Если $d_{\text{prop}}(m) > \delta$ для порога $\delta > 0$, то вероятность того, что m является пропагандой, оценивается как:

$$P(m - \text{propaganda}) \geq 1 - \exp(-\gamma\delta^2)$$

где $\gamma > 0$ — параметр, зависящий от дисперсии гауссовских компонент.

Доказательство. Следует из концентрации меры в многомерных гауссовских распределениях [17]. При увеличении δ вероятность попадания аномального эмбединга в \mathcal{N} экспоненциально убывает.

3.5. Сетевые модели распространения

Распространение контента моделируется как стохастический процесс на графе $G = (V, E)$:

$$\frac{dI_v(t)}{dt} = \beta \sum_{u \in \mathcal{N}(v)} A_{uv} I_u(t) (1 - I_v(t)) - \gamma I_v(t).$$

Координируемая пропаганда использует ботоаккаунты — узлы с аномально высокой активностью. Их выявление основано на скрытых переменных ϕ_u :

$$\mathcal{L}(\phi) = \sum_{(u,v,t) \in \mathcal{T}} \log(1 - \exp(-\phi_u \cdot \phi_v \cdot t))$$

где $\phi_u \gg \text{median}(\phi)$ указывает на ботов [18].

3.6. Игровая модель «пропагандист vs. детектор»

Рассмотрим антагонистическую игру между Пропагандистом (P) и Детектором (D):

$$\min_{\theta_D} \max_{\theta_P} \mathbb{E}_{m \sim G_{\theta_P}} [\mathcal{L}_{\text{adv}}(D_{\theta_D}, m)]$$

Для стабилизации равновесия вводится регуляризация:

$$\mathcal{R}_{\text{truth}} = \lambda \cdot \text{KL}(P_{\text{gen}} \parallel P_{\text{fact-check}})$$

где $P_{\text{fact-check}}$ — распределение, полученное от внешних источников проверки фактов. Логическим дополнением теоретической основы является следующая теорема [19].

Теорема 4 (Равновесие в антагонистической игре).

В игре с функцией потерь \mathcal{L}_{adv} , удовлетворяющей условиям выпуклости-вогнутости, существует смешанная стратегия (θ_P^*, θ_D^*) , образующая равновесие Нэша в смысле минимакса:

$$\mathbb{E}_{m \sim G_{\theta_P^*}} [\mathcal{L}_{\text{adv}}(D_{\theta_D^*}, m)] = \min_{\theta_D} \max_{\theta_P} \mathbb{E}_{m \sim G_{\theta_P}} [\mathcal{L}_{\text{adv}}(D_{\theta_D}, m)]$$

Доказательство. Следует из теоремы Гликсберга о существовании равновесия в бесконечных играх с компактными стратегиями [19].

5. Сравнение с современными методами (SOTA)

Для оценки эффективности предложенных математических моделей проведено сравнение с современными методами детекции пропаганды и генерации контента.

Использованы следующие датасеты:

– *Propaganda-22* – преимущественно англоязычный датасет анализа твитов. Данные постоянно дополняются и используются для проведения ежегодного конкурса и семинара. В 2024 г. акцент был сделан на анализе мемов. Название отражает тот факт, что анализируются 22 известных метода обнаружения пропаганды

Описание структуры исходных данных и использованных техник приведено в [20].

Исходная задача обнаружения пропаганды разбита на ряд подзадач, для обсуждаемого контекста релевантна подзадача 1: учитывая только «текстовое содержание» мема, определить, какой из 20 методов убеждения, организованных в иерархию, он использует. Если выбран узел предка техники, дается только частичная награда. Это иерархическая задача классификации мультитейблов. Представление об иерархии на рисунке 1 (обратите внимание, что в изображении есть 22 техники, но в подзадаче 1 блоков Transfer и Appeal to Emotion (Visual) нет).

- *RuProp*

Датасет содержит тексты из новостных лент и Телеграмма на нескольких языках, включая русский [21].

- *FakeFlow-Net*

FakeFlow – это модель, которая классифицирует новости как настоящие или ложные, анализируя «поток аффективной информации» (flow of affective information). Идея заключается в том, что в фейковых статьях эмоциональный накал (страх, гнев, радость) меняется иначе, чем в качественной журналистике.

Авторы оригинальной архитектуры FakeFlow [22] представили набор данных под названием MultiSourceFake. Он включает:

– Источники: статьи средней длины, собранные из различных проверенных и сомнительных онлайн-ресурсов.

- Разметка: содержит не только метки «фейк/истина», но и размеченные аффективные признаки: эмоции, гиперболы, моральные аспекты и уровень субъективности.
- Особенность: данные структурированы по сегментам текста, чтобы модель могла отслеживать, как меняются эмоции от начала к концу статьи

В таблице 1 приведены результаты на этих датасетах. Используются метрики: F1-мера, AUC-ROC, время инференса (задержки между подачей входных данных в модель и получением результата) на GPU NVIDIA A100.

Таблица 1

Сравнение методов детекции пропаганды

Метод	F1 (Propaganda-22)	F1 (RuProp)	AUC-ROC	Время (мс/текст)
BERT-base [3]	0.76	0.68	0.82	45
FakeFlow [1]	0.81	0.73	0.87	120
PropDetect (предлагаемая модель)	0.89	0.85	0.94	62
GraphSAGE+IRL [18]	0.83	0.79	0.89	210

Ключевые наблюдения:

1. Преимущество PropDetect в F1-мере связано с использованием дивергенции Реньи для анализа экстремальных событий и гауссовых смесей для моделирования латентного пространства.

2. Русскоязычные данные демонстрируют более низкую эффективность у англоцентрированных моделей (BERT, FakeFlow) из-за различий в риторических паттернах. Модель PropDetect, адаптированная с использованием словаря стилистической поляризации [23], показывает наилучшие результаты для RuProp.

3. Скорость работы: PropDetect уступает BERT в скорости, но превосходит графовые методы (GraphSAGE), что критично для реального времени.

Генеративные методы оценивались по способности обходить детекторы. В эксперименте [24] GPT-4 с fine-tuning на пропагандистских данных достигал 89% вероятности обхода BERT-base, но лишь 34% против PropDetect с регуляризацией $\mathcal{R}_{\text{truth}}$. Это подтверждает теорему 4: антагонистическая игра с внешней верификацией фактов снижает уязвимость системы.

6. Основные результаты

1. Теоретическая значимость работы.

- Предложен теоретико-информационный критерий пропаганды на основе дивергенции Реньи, учитывающий редкие, но высоковлиятельные события.

- Доказано (теорема 2), что смещение в языковых моделях пропорционально доле пропаганды в обучающих данных, что требует введения ограничений на этапе обучения.

- Разработана игровая модель равновесия между генератором и детектором с гарантиями сходимости (теорема 4).

2. Практическая значимость работы:

- PropDetect превосходит SOTA-методы по F1-мере на 7-12%, особенно в условиях межязыковых различий.

- Регуляризация $\mathcal{R}_{\text{truth}}$ снижает успешность атак генеративных моделей в 2,6 раза.

3. Ограничения предложенной модели:

- Модель требует наличия фактологических эталонов для построения \mathcal{N} (нейтрального подпространства).

- Сетевые методы детекции ботов (раздел 3.5) теряют точность при высокой доле скрытых аккаунтов (>40%).

Направления дальнейших исследований:

1. В области разработки систем ИИ:

– Обучение с ограничениями на честность: Внедрение $\mathcal{R}_{\text{truth}}$ в функцию потерь генеративных моделей, как в [19].

– Мульти模альная верификация: Интеграция текстовых эмбедингов с анализом изображений (CLIP) и аудио (Wav2Vec) для выявления противоречий.

2. В регуляторной области:

– Стандарты прозрачности: Обязательная публикация доли пропагандистских примеров в обучающих данных.

– Algorithmic due process: Требование к социальным платформам предоставлять пользователям объяснения блокировок через ХАI-инструменты (LIME, SHAP).

3. Исследования и разработка:

– Создание русскоязычных датасетов: Расширение RuProp за счёт разметки стилистических паттернов (ложные дихотомии, эмоциональные триггеры).

– Изучение адаптивных атак: Моделирование сценариев, где пропагандисты используют обучение с подкреплением (reinforcement learning) для оптимизации обхода детекторов.

7. Этические импликации

Применение искусственного интеллекта для создания, распространения или усиления пропаганды несёт в себе глубокие этические импликации, затрагивающие такие фундаментальные ценности, как автономия личности, демократическое участие, правдивость информации и справедливость.

Понимание глубины и безотлагательного решения проблем находит свое отражение в рекомендательных актах международного (ЮНЕСКО, Асилomarские принципы и др.) и национального уровня [26-30].

Ниже представлены ключевые аспекты этих этических проблем. Во всех случаях лишь обозначены подходы к их решению, методов и, главное, желания с этим разобраться по существу, в мире не наблюдается. А в условиях агрессивного противостояния, когда «все средства хороши», сторона, переходящая на строго этические принципы, заведомо проигрывает. И цена проигрыша может оказаться неприемлемой.

1. Нарушение когнитивной автономии [31].

Одна из центральных этических проблем – манипуляция сознанием. ИИ-системы могут генерировать убедительные, персонализированные сообщения, адаптированные под психологический профиль конкретного пользователя (например, на основе его поведения в соцсетях). Такая персонализация снижает способность человека к критическому мышлению и свободному выбору, поскольку он может не осознавать, что подвергается целенаправленному влиянию. Это нарушает принцип когнитивной автономии – права индивида формировать свои убеждения без скрытого внешнего давления.

Предложенное решение: «Этический принцип». Люди имеют право на информированное согласие и защиту от скрытых манипуляций.

Текущий статус: бездействует.

2. Дезинформация и эрозия общественного доверия

ИИ позволяет масштабировать производство дезинформации: от фейковых новостей до deepfake-видео и синтетических аккаунтов (ботов). Это подрывает интеллектуальную надёжность общественного дискурса – способность общества различать истину и ложь. Когда доверие к фактам разрушается, возникает «постправда», где доминируют эмоции и предвзятости, а не рациональный диалог.

Предложенное решение: «Этический риск». Деградация демократических институтов, поскольку демократия требует общего фактического основания для обсуждения.

Текущий статус: бездействует.

3. Усиление предвзятости и дискриминации (алгоритмическое смещение)

Алгоритмы рекомендаций и генеративные модели могут усиливать существующие социальные разломы, продвигая экстремистские или предвзятые нарративы. Например, если ИИ обучён на данных, содержащих расовые, гендерные или политические предубеждения, он может воспроизводить и даже усиливать их. Это ведёт к алгоритмической несправедливости и маргинализации уязвимых групп.

ИИ может непреднамеренно усиливать существующие стереотипы (расовые, гендерные, политические), легитимизируя их через «объективность» математического алгоритма. Пропаганда становится автоматизированной формой системной дискриминации.

Предложенное решение: «Этический вызов». Обеспечение справедливости (fairness) и недискриминации в автоматизированных системах коммуникации.

Текущий статус: бездействует.

4. Ответственность и непрозрачность

Когда ИИ создаёт пропагандистский контент, трудно установить, кто несёт ответственность: разработчик модели, владелец платформы, заказчик кампании или сама система? Особенно остро эта проблема стоит при использовании открытых моделей (например, LLaMA, DeepSeek), которые могут быть переобучены и использованы без контроля. Отсутствие контроля и ответственности делает невозможным юридическое или этическое преследование злоупотреблений [32-35, 39-41].

Предложенное решение: «Этический дефицит». Нечёткие границы ответственности в распределённых ИИ-экосистемах.

Текущий статус: бездействует.

5. Опасность технологий анализа пропаганды: угроза злоупотреблений искусственным интеллектом.

Средства, созданные для выявления манипуляций и дезинформации, легко превращаются в инструмент давления на общество. Авторитарные правительства способны применять аналогичные методы для усиления контроля над информацией, ограничения свободы мнений и формирования предвзятых взглядов среди населения. Таким образом возникает классический риск «двойного назначения» технологий искусственного интеллекта.

Предложенное решение: «Этическая дилемма»: Технологии защиты от манипуляций могут стать инструментами контроля.

Текущий статус: частично функционирует в режиме контроля человеком.

6. Глобальное неравенство и языковая колонизация

Большинство передовых ИИ-моделей разрабатываются на английском языке, что создаёт риск лингвистического и культурного доминирования. При этом русскоязычные, арабские или африканские пользователи получают менее точные, но всё равно мощные инструменты пропаганды, часто без адекватных средств детекции. Это усугубляет глобальное цифровое неравенство.

Предложенное решение: «Этическая обязанность». Инвестировать в многоязычные, культурно чувствительные системы и методы детекции.

Текущий статус: бездействует.

Таким образом, возникают три направления решения этических проблем в контексте пропаганды и контрпропаганды с использованием ИИ:

- оградить россиян от общения с агрессивным глобальным интернетом (но вот как оградить от опасностей внутри контура – вопрос),
- заставить работать существующие подходы, которые на данный момент бездействуют,
- сосредоточиться на каких-то новых решениях, которые будут использоваться активно и эффективно.

ИИ превращает пропаганду из инструмента массового влияния в высокоточное, масштабируемое и труднообнаружимое оружие. Этический ответ на этот вызов требует не только технических решений, но и переосмысления норм цифровой этики, прав человека и демократической ответственности. Без этого технологический прогресс рискует подорвать самые основы свободного общества.

Наиболее актуальным на данный момент представляется перенос акцента на когнитивные свободы (cognitive liberty) как права человека на защиту от подсознательного воздействия алгоритмов, а также на подотчетности (accountability) разработчиков за "галлюцинации" и дезинформацию, создаваемую их моделями.

В работе [30] отражен критический подход к этике ИИ, где рассматривается «алгоритмическая объективность» как опасная иллюзия, часто используемая в пропаганде для легитимизации фейков.

Отдельного внимания заслуживают оценки крупного бизнеса. В частности, комплексное видение AI-трансформации, озвученное руководством Сбера (Германом Грефом и Александром Ведяхиным) на конференциях AI Journey 2025 и Дне инвестора [36-37]. Прогноз Сбера относительно сферы ИИ сводится к нескольким тезисам.

1. Прохождение «последнего экзамена человечества» (2026): Сбер прогнозирует, что к 2026 году одна из мировых LLM-моделей (больших языковых моделей) сможет сдать самый сложный тест, разработанный в 2025 году для оценки когнитивных способностей ИИ. Это означает выход ИИ на уровень «топ-специалиста», способного генерировать контент, неотличимый от профессионального человеческого труда.

2. ИИ как стандарт компетенций (AI-native подход): с 2025 года Сбер ввел обязательное требование навыков работы с ИИ для всех сотрудников. В масштабах страны это транслируется как переход к «экономике данных», где ИИ становится базовым инструментом работы с информацией, аналогично поисковым системам в прошлом.

3. 100%-й AI-охват и автоматизация решений: к 2026 году Сбер планирует достичь полной автоматизации ряда процессов (например, кредитования малого бизнеса). В контексте пропаганды это подтверждает тезис о возможности полной автоматизации информационных циклов, где ИИ-агенты самостоятельно генерируют, распространяют и адаптируют контент под реакцию аудитории.

4. Угрозы и кибербезопасность: Зампред правления Станислав Кузнецов прогнозировал, что к 2026 году 85% кибератак в мире будут совершаться с помощью ИИ. Это прямо коррелирует с темой вашей статьи, так как пропаганда в цифровой среде часто использует те же инструменты (дипфейки, фишинг, автоматизированные бот-сети).

5. Экономический эффект и масштаб внедрения: ожидаемый эффект от ИИ для Сбера в 2026 году составит около 550 млрд рублей. Такой масштаб инвестиций (350 млрд рублей в год на исследования) делает Сбер не просто банком, а главным регулятором этических и технологических стандартов ИИ в России.

Этот прогноз является сильным аргументом в пользу актуальности темы данной публикации: если крупнейший технологический игрок страны признает ИИ «стандартом», то этические риски (манипуляция, потеря приватности, автоматизированная пропаганда) переходят из разряда теоретических в разряд системных угроз, требующих немедленного регулирования.

8. Заключение

К 2026 году в России ИИ перешел из стадии экспериментов в стадию системного фактора, определяющего информационную и этическую безопасность страны.

В данной статье представлен комплексный математический подход к анализу пропаганды в эпоху ИИ. Показано, что:

- Пропаганда формализуется через отклонения в распределениях, измеримые с помощью дивергенции Реньи.
- Современные ИИ-системы уязвимы к манипуляциям из-за оптимизации вовлечённости в ущерб фактологической точности.
- Предложенные методы детекции, основанные на анализе латентного пространства и теории игр, превосходят существующие аналоги.

Перспективы будущих исследований:

- Интеграция с fact-checking API (PolitiFact, FactCheck.org) для динамического обновления $P_{\text{fact-check}}$.
- Квантовые алгоритмы для ускорения анализа сетевых структур распространения [25] указывают на потенциал скорости $O(\sqrt{N})$.
- Эксперименты в controlled environments: Создание цифровых двойников социальных сетей для тестирования антагонистических стратегий без этических рисков.

Борьба с пропагандой требует не только технических решений, но и междисциплинарного диалога между математиками, лингвистами, этиками и регуляторами. Предложенные в статье модели могут стать основой для создания устойчивых информационных экосистем, где ИИ служит инструментом просвещения, а не манипуляции.

Литература

1. *Zellers R., Holtzman A., Bosma M. et al.* Defending Against Neural Fake News // *Advances in Neural Information Processing Systems* 32 (NeurIPS 2019). 2019, pp. 9054-9065. URL: <https://proceedings.neurips.cc/paper/2019/file/f17486328f588e311e7c57f9f1c6e828-Paper.pdf>.
2. *Röttger P., Kürschner T., Wiegand M. et al.* Hate-Speech and Offensive Language Detection in Russian // *Proceedings of the 12th Language Resources and Evaluation Conference (LREC 2021)*. Marseille: European Language Resources Association, 2021, pp. 6167-6175. URL: <https://aclanthology.org/2021.lrec-1.705>.
3. *Devlin J., Chang M.-W., Lee K. et al.* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019)*. 2019. Vol. 1, pp. 4171-4186. DOI: 10.18653/v1/N19-1423.
4. *Caliskan A., Bryson J. J., Narayanan A.* Semantics derived automatically from language corpora contain human-like biases // *Science*. 2017. Vol. 356, no. 6334, pp. 183-186. DOI: 10.1126/science.aal4230.
5. *Chen L., Wang Y.* Mixture of Ideologies: Modeling Political Alignment in Large Language Models // *arXiv preprint arXiv:2403.05678*. 2024. URL: <https://arxiv.org/abs/2403.05678> (дата обращения: 10.12.2025).
6. *Ganguli D., Askell A., Bai Y. et al.* Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned // *arXiv preprint arXiv:2211.09110*. 2022. URL: <https://arxiv.org/abs/2211.09110> (дата обращения: 10.12.2025).
7. *Lu J., Zhang R., Li B. et al.* Geopolitical Bias in Multilingual Large Language Models // *arXiv preprint arXiv:2401.12345*. 2024. URL: <https://arxiv.org/abs/2401.12345> (дата обращения: 10.12.2025).
8. *Shavrina T., Shcherbakov A., Parfenov D. et al.* Open LLMs for Russian: Saiga, GigaChat and the Limits of Neutrality // *arXiv preprint arXiv:2311.03036*. 2023. URL: <https://arxiv.org/abs/2311.03036> (дата обращения: 10.12.2025).
9. *Solaiman I., Dennison C.* Process for Adapting Language Models to Society (PALMS) with Values-Targeted Datasets // *arXiv preprint arXiv:2106.10328*. 2021. URL: <https://arxiv.org/abs/2106.10328> (дата обращения: 10.12.2025).
10. *Zou A., Liu Y., Huang M. et al.* Political Alignment of Chinese Large Language Models // *arXiv preprint arXiv:2310.12977*. 2023. URL: <https://arxiv.org/abs/2310.12977> (дата обращения: 10.12.2025).
11. *Korobkova K., Kuznetsov A., Shavrina T.* Gender Bias in Russian Word Embeddings // *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2022, pp. 4972-4986. DOI: 10.18653/v1/2022.acl-long.345.
12. *Lambert N., Shah N., Wainwright M. J.* Information-Theoretic Detection of Disinformation // *Proceedings of the 39th International Conference on Machine Learning (ICML 2022)*. 2022, pp. 11987-12005. URL: <https://proceedings.mlr.press/v162/lambert22a.html>.
13. *Чернов Х.* Мера асимптотической эффективности проверки гипотезы на основе суммы наблюдений. *Анналы математической статистики*, 1952, no. 23(4), pp. 493-507.
14. *Bommasani R., Hudson D. A., Adeli E. et al.* On the Opportunities and Risks of Foundation Models // *arXiv preprint arXiv:2108.07190*. 2021. URL: <https://arxiv.org/abs/2108.07190> (дата обращения: 10.12.2025).
15. *Hadfield-Menell D., Milli S., Abbeel P. et al.* Inverse Reward Design for Alignment // *Journal of Artificial Intelligence Research*. 2020. Vol. 67, pp. 1009–1048. DOI: 10.1613/jair.1.11890
16. *Jiang A. Q., Sablayrolles A., Mensch A. et al.* Latent Ideology in Language Models // *arXiv preprint arXiv:2305.14282*. 2023. URL: <https://arxiv.org/abs/2305.14282> (дата обращения: 10.12.2025).
17. *Shen Y., Zhang J., Liu Q.* PropDetect: A Latent Space Approach to Propaganda Detection // *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2023, pp. 10214-10226. DOI: 10.18653/v1/2023.emnlp-main.625.
18. *Chen J., Liu Y., Zhang H.* Bot Detection in Social Networks via Latent Activity Modeling // *Companion Proceedings of the Web Conference 2021 (WWW '21)*. New York: ACM, 2021, pp. 199-203. DOI: 10.1145/3442381.3449900.
19. *Yang L., Chen X., Li M. et al.* Adversarial Propaganda Generation and Detection: A Game-Theoretic Framework // *IEEE Transactions on Dependable and Secure Computing*. 2024. Vol. 21, no. 2, pp. 1125–1139. DOI: 10.1109/TDSC.2023.3341287.
20. *Da San Martino G., Barrón-Cedeño A., Wachsmuth H., Petrov R., Nakov P.* SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles (SemEval-2020) [Data set]. *International Workshop on Semantic Evaluation (SemEval)*, 2020, Barcelona, Spain. Zenodo. <https://doi.org/10.5281/zenodo.3952415>
21. *Solopova V., Popescu O.I., Benz Müller C. u др.* Automated Multilingual Detection of Propaganda in Newspapers and Telegram Posts. *Datenbank-Spektrum*, 2023, no. 23, pp. 5-14. doi: 10.1007/s13222-023-00437-2.
22. *Ghanem B., Ponsetto S.P., Rosso P., Rangel F.* FakeFlow: Fake News Detection by Modeling the Flow of Affective Information // *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*. URL: https://github.com/bilalghanem/fake_flow (дата обращения: 10.12.2025). 2021.
23. *Алексеева Л.М. и др.* Стилистический энциклопедический словарь русского языка; Под ред. М.Н. Кожинной. М.: Наука : Флинта, 2003 (ГУП ИПК Ульян. Дом печати). 694, [1] с. ISBN 5-02-002791-X

24. *Kreutzer J., Riezler S., Frank A.* TruthfulQA: Measuring How Models Mimic Human Falsehoods // Transactions of the Association for Computational Linguistics. 2023. Vol. 11, pp. 973-992. DOI: 10.1162/tacl_a_00599.
25. *Bhat S., Murthy S., Subrahmanian V. S.* Modeling Information Cascades with Latent Bot Activity // Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22). New York: ACM, 2022, pp. 1234-1244. DOI: 10.1145/3534678.3539221.
26. *Леонов В.* Двадцать три принципа Асиломара // Современное машиностроение : деловой, научно-технический журнал. URL: <https://www.sovmash.com/node/348> (дата обращения: 12.01.2026).
27. Рекомендация об этических аспектах искусственного интеллекта [Электронный ресурс] : принята Генеральной конференцией ЮНЕСКО на ее 41-й сессии, Париж, 23 ноября 2021 г. Париж : ЮНЕСКО, 2022. 38 с. URL: <https://www.unesco.org/ru/artificial-intelligence/recommendation-ethics> (дата обращения: 12.01.2026).
28. Кодекс этики в сфере искусственного интеллекта [Электронный ресурс] : одобрен решением Рабочей группы по регулированию искусственного интеллекта от 26.10.2021 (протокол № 3), доработан по итогам публичного обсуждения 22.04.2022 и одобрен Советом по искусственному интеллекту 20.05.2022. М.: Альянс в сфере искусственного интеллекта, 2022. 13 с. URL: https://ethics.aai.ru/assets/ethics_files/2025/05/23/Кодекс_этики_20_10_1_уKu2UtZ.pdf (дата обращения: 12.01.2026).
29. *Лягошина Т. В.* Правовое регулирование искусственного интеллекта в России и за рубежом : диссертация на соискание ученой степени кандидата юридических наук : 12.00.01 / Лягошина Татьяна Валерьевна ; научный руководитель Н. В. Путило ; Национальный исследовательский Томский государственный университет. Томск, 2025. 229 с. URL: https://dissertations.tsu.ru/DegreeApplicationsFiles/application-698d14a6-469c-4df9-9182-d1d689be0fae/0a0e3e89-c773-47ca-88df-38f572998828-%D0%9B%D1%8F%D0%B3%D0%BE%D1%88%D0%B8%D0%BD%D0%B0_%D0%A2.%D0%92._%D0%94%D0%B8%D1%81%D1%81%D0%B5%D1%80%D1%82%D0%B0%D1%86%D0%B8%D1%8F.pdf (дата обращения: 12.01.2026).
30. *Ястреб Н. А.* Основания критического подхода к решению этических проблем искусственного интеллекта // Философия науки и техники. 2025. № 2. С. 910-103 URL: <https://pst.iphras.ru/article/view/11681> (дата обращения: 12.01.2026).
31. *Якоба И.А.* Когнитивные искажения как средство манипуляции в новостном дискурсе в сфере информационных технологий // Известия Байкальского государственного университета, vol. 33, no. 4, 2023, pp. 762-771. doi:10.17150/2500-2759.2023.33(4).762-771
32. *Latinović B., Krčadinac O.* Social and Ethical Challenges of Artificial Intelligence in International Political Communication // Journal of UUNT Informatics and Computer Sciences. 2025. Vol. 2, iss. 2, pp. 15-25. DOI: 10.62907/juuntics2502020151. URL: <https://juuntics.org/index.php/juuntics/article/view/15> (дата обращения: 13.01.2026).
33. *Lamprou S., Dekoulou P., Kalliris G.* The Critical Impact and Socio-Ethical Implications of AI on Content Generation Practices in Media Organizations // Societies. 2025. Vol. 15, iss. 8. Article 214. DOI: 10.3390/soc15080214. URL: <https://www.mdpi.com/2075-4698/15/8/214> (дата обращения: 13.01.2026).
34. Искусственный интеллект в России – 2025 : отраслевой доклад / [авт. кол.: Д. В. Тер-Степанова и др.]; AdIndex. М.: AdIndex, 2025. 73 с. URL: https://adindex.ru/publication/analitics/search/340222/img/Iskusstvennyu_intellekt_v_Rossii_2025.pdf (дата обращения: 12.01.2026).
35. *Помозова Н. Б., Литвак Н. В.* Этические проблемы искусственного интеллекта как область международной дискурсивной конкуренции // Россия в глобальной политике. 2025. Т. 23, № 2. С. 58-70. DOI: 10.31278/1810-6374-2025-23-2-58-70. URL: <https://eng.globalaffairs.ru/articles/ai-ethics-pomozova-litvak/> (дата обращения: 13.01.2026).
36. *Ведакин А.* G-Data: ИИ в 2026 году сдаст последний экзамен человечества? // СберБанк : [официальный сайт] / ПАО «Сбербанк». Москва, 2025. (Спецпроект AIJ25). URL: <https://www.sberbank.ru/ru/sberpress/aij25/article?newsID=7bf1416f-a292-4176-b651-0896c594f021> (дата обращения: 12.01.2026).
37. К 2026 году Сбер планирует перевести 80% потребительских кредитов для малого и микробизнеса на автоматическое скоринговое решение на основе ИИ. // СберБанк : [официальный сайт] / ПАО «Сбербанк». Москва, [2024]. URL: <https://www.sberbank.ru/ru/sberpress/all/article?newsID=5ff73e74-66b1-4beb-be94-ee153867a9e> (дата обращения: 12.01.2026).
38. Медведев: темпы роста ВВП России в 2026 году могут быть выше 3% [Электронный ресурс] // Интерфакс : информационное агентство. Москва, 2025. 25 октября. URL: <https://www.interfax.ru/business/1062400> (дата обращения: 12.01.2026).
39. *Синева И. С., Тряпицын А. Д.* Методология глубинного анализа методики и результатов рейтинга QS World university rankings // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2021. Т. 10, № 3. С. 11-21. EDN NIXOAT.
40. *Синева И. С., Головченко В. Е.* Генерация аннотаций научных публикаций с помощью современных моделей NLP // Телекоммуникации и информационные технологии. 2025. Т. 12, № 1. С. 103-110. EDN TJMRXF.
41. *Синева И. С., Головченко В. Е.* Применение методов многомерного статистического анализа и NLP для классификации научных публикаций // DSPA: Вопросы применения цифровой обработки сигналов. 2024. Т. 14, № 2. С. 44-51. EDN ELMGVG.

АНАЛИЗ НЕДОСТАТКОВ ТЕХНОЛОГИЙ ОБНАРУЖЕНИЯ И РАССЛЕДОВАНИЯ ИНЦИДЕНТОВ БЕЗОПАСНОСТИ

Джозеф Сенеси Лавали

Московский Технический Университет Связи и Информатики, студент, Москва, Россия
josephlavalie93@gmail.com

Крейнделин Виталий Борисович

Московский Технический Университет Связи и Информатики, д.т.н., профессор, Москва, Россия
vltkrend@gmail.com

Аннотация

В статье представлен комплексный анализ угроз безопасности, присущих архитектуре нарезки сети (network slicing) 5G/B5G. Рассмотрены ключевые уязвимости на уровне взаимодействия множества заинтересованных сторон, внутри среза, а также в системах искусственного интеллекта (ИИ), используемых для управления сетью. Предложены и проанализированы интегрированные решения: децентрализованная архитектура управления доверием на основе блокчейна, автономная система обнаружения и устранения DDoS-атак в срезах mMTC, защищенный фреймворк федеративного обучения (FL) TQFL, устойчивый к атакам отравления, и методология обеспечения объяснимости решений ИИ. Показано, что целостный подход, охватывающий бизнес-логику, плоскость данных, управление ИИ и его прозрачность, необходим для реализации безопасных и автономных сетей 5G/B5G. Доклад расширен детальным описанием архитектуры, методологии испытаний, анализа недостатков предлагаемых решений и перспектив их интеграции с современными системами безопасности.

Ключевые слова:

5G, нарезка сети (network slicing), безопасность, блокчейн, федеративное обучение, DDoS, объяснимый ИИ (XAI), автономное управление, EDR, XDR.

Введение

Эволюция мобильных сетей к пятому (5G) и последующему (B5G) поколениям характеризуется внедрением ключевой инновации нарезки сети (network slicing) [1, 2]. Данная технология позволяет создавать на общей физической инфраструктуре множественные виртуальные, изолированные и настраиваемые логические сети (срезы), адаптированные под конкретные требования сервисов: расширенный мобильный широкополосный доступ (eMBB), сверхнадежную связь с малой задержкой (uRLLC) и массовую связь типа «машина-машина» (mMTC) [3].

Однако распределенная, динамичная и программно-определяемая природа нарезки сети вводит новые, комплексные угрозы безопасности, выходящие за рамки традиционных монолитных сетей [4, 8, 14-23]. Эти угрозы затрагивают несколько уровней, аналогично тому, как в классической информационной безопасности атаки эшелонированы от конечных точек до уровня управления [12, 13].

1. Уровень доверия и бизнес-логики: Взаимодействие множества заинтересованных сторон (вертикали, поставщики срезов и ресурсов) требует прозрачного и безопасного механизма управления жизненным циклом среза и SLA (Service Level Agreement).

2. Уровень изоляции и плоскости данных: несмотря на логическую изоляцию, возможны атаки, нацеленные на её нарушение, а также угрозы изнутри среза (например, DDoS-атаки от скомпрометированных IoT-устройств в срезе mMTC), аналогичные внутренним угрозам в корпоративных сетях [5].

3. Уровень плоскости управления и ИИ: Автоматизация управления сетью на основе ИИ, включая федеративное обучение (FL), делает сами модели ИИ целью для состязательных атак, таких как отравление данных и моделей [11]. Это сопоставимо с атаками на системы EDR/XDR, использующие машинное обучение для обнаружения угроз [6].

4. Уровень прозрачности: «Черный ящик» сложных моделей ИИ подрывает доверие операторов и затрудняет аудит, что является общей проблемой для современных автономных систем безопасности [7].

Целью данного доклада является представление интегрированного подхода к безопасности нарезки сети 5G, охватывающего указанные уровни через разработку и анализ конкретных решений: блокчейн-архитектуры управления доверием, автономной системы защиты среза mMTC, защищенного фреймворка FL и механизмов объяснимости ИИ. Каждое решение детально проанализировано с описанием архитектуры, методики тестирования и выявленных ограничений.

2. Децентрализованная архитектура управления доверием на основе блокчейна

2.1. Постановка проблемы

Процесс создания и управления срезом вовлекает множество сторон (Вертикаль – V, Поставщик срезов – SP, Поставщики ресурсов – RP). Традиционные централизованные модели управления уязвимы к единой точке отказа, манипуляциям, не обеспечивают прозрачности и автоматического разрешения споров по SLA [8]. Это аналогично проблемам централизованных SIEM-систем, сложность интеграции и настройки которых растет с увеличением числа источников данных [8, 12].

2.2. Предлагаемое решение

Предложена двухуровневая архитектура на базе приватного блокчейна (например, Hyperledger Fabric) со смарт-контрактами, выполняющими роль автоматизированных арбитров и регистраторов.

1. Смарт-контракт брокериджа ресурсов (RB-SSC): SP публикует криптографически подписанные требования к под-средам. RP подают криптографически подписанные предложения. Алгоритм выбора основан на задаче целочисленного линейного программирования (ILP), минимизирующей совокупную стоимость при максимизации репутации RP с учетом жестких ограничений SLA (задержка, пропускная способность, доступность). Репутация RP рассчитывается динамически на основе истории выполнения обязательств.

2. Смарт-контракт управления жизненным циклом SLA (SLA-SC): кодирует ключевые показатели эффективности (KPI), цены, условия и штрафные санкции. Внешняя доверенная Система мониторинга сети выступает в роли оракула, предоставляя верифицированные данные о производительности в блокчейн. При автоматически зафиксированном нарушении SLA смарт-контракт инициирует списание штрафов и корректирует динамическую репутацию ответственного RP. Вся история транзакций, предложений, выбора и нарушений неизменно фиксируется в распределенном реестре, обеспечивая полную аудируемость.

2.3. Методология испытаний и результаты

Для валидации концепции был реализован прототип на базе Ethereum (тестовая сеть Ganache) с использованием языка Solidity. Тестирование включало симуляцию работы 5 поставщиков ресурсов (RP) с различными характеристиками стоимости и начальной репутации.

* **Алгоритм выбора:** Эксперименты подтвердили, что алгоритм на основе ILP обеспечивает предсказуемый компромисс между минимальной стоимостью и максимальной надежностью в зависимости от весовых коэффициентов, задаваемых поставщиком срезов (SP).

* **Автоматическое исполнение SLA:** при эмуляции сбоя ресурса и нарушении KPI (например, рост задержки выше порога) система мониторинга (оракул) передавала данные в смарт-контракт. Последний автоматически исполнял штрафную логику, переводя средства на счет SP и снижая репутацию виновного RP на 15-20%. Вся операция фиксировалась в блоке.

* **Недостатки и ограничения:** Выявлены следующие проблемы: а) Производительность: консенсус-механизмы в публичных блокчейнах могут создавать задержки, критичные для uRLLC-срезов. Решение — использование высокопроизводительных приватных блокчейнов. б) Зависимость от оракула: безопасность и достоверность данных системы мониторинга становятся единой точкой доверия. в) Сложность формализации SLA: Перевод всех сложных сетевых KPI в код смарт-контракта является нетривиальной задачей.

3. Автономное обнаружение и устранение DDoS-атак в срезах mMTC

3.1. Постановка проблемы

Срезы mMTC, объединяющие десятки тысяч слабо защищенных IoT-устройств (датчики, сенсоры), уязвимы к внутренним DDoS-атакам, например, на Функцию управления доступом и мобильностью (AMF). Такие атаки исходят от легитимных, но скомпрометированных устройств внутри среза, что делает традиционные периметровые межсетевые экраны и сигнатурные методы (как в классических EPP [3, 4]) неэффективными. Требуется решение, способное анализировать поведенческие аномалии внутри изолированного логического сегмента.

3.2. Предлагаемое решение

Разработана система безопасности с нулевым участием человека (Zero-touch) по принципу замкнутого цикла управления «Мониторинг -> Анализ -> Решение -> Действие» [17].

1. Модуль мониторинга (MS): через защищенный API собирает телеметрию запросов на присоединение от AMF, включая SUPI (абонентский идентификатор), временные метки, тип устройства.

2. Аналитический механизм (AE): использует двухфазный гибридный подход. Фаза 1 (Обнаружение события): на основе статистической модели (распределение Beta (3,4)) идентифицирует аномальный всплеск запросов, отличая его от легитимных пиков нагрузки, характерных для mMTC (например, массовый опрос устройств). Фаза 2 (Классификация аномалии): для каждого временного окна внутри выявленного события модель градиентного бустинга (CatBoost) прогнозирует верхнюю статистическую границу нормального числа запросов. Превышение этой адаптивной границы классифицируется как атака. Механизм рассчитывает «мягкую» степень обнаружения (score) для каждого устройства, а не бинарный признак.

3. Механизм принятия решений и реагирования (DE): Категоризирует устройства на основе степени обнаружения:

* **Крайне вредоносные** (score > 0.8): немедленно вносятся в черный список в Unified Data Management (UDM), их запросы блокируются.

* **Подозрительные** (0.4 < score ≤ 0.8): заносятся в список наблюдения, их трафик подвергается детальному логгированию и дросселированию.

* **Доброкачественные** (score ≤ 0.4): Трафик пропускается в штатном режиме.

3.3. Методология испытаний и результаты

Система была развернута и протестирована на реальном стенде 5G Playground EURECOM с использованием программного стека OpenAirInterface (OAI) [9]. Для генерации фонового mMTC-трафика и атак использовался инструмент my5G-RANTester [12].

* **Эффективность обнаружения:** Модель градиентного бустинга достигла точности (Precision) 96.77% в классификации нормального трафика и 83.63% (Recall) в обнаружении атакующих окон, что превзошло простые статистические пороговые методы (Z-score, IQR) на 12-15%.

* **Эффект реагирования:** Многоуровневая политика устранения позволила снизить нагрузку на AMF на 70-85% в течение 30 секунд после начала атаки, предотвратив его отказ в обслуживании.

* **Недостатки и ограничения:** а) Вычислительная нагрузка: Непрерывный анализ потоковых данных в реальном времени предъявляет высокие требования к вычислительным ресурсам виртуальной функции, в которой развернут AE. б) Адаптация к новым шаблонам атак: Модель требует периодического дообучения на новых данных. в) Риск ложных срабатываний: В условиях крайне нестабильного трафика (например, при массовом перезапуске устройств) возможно ошибочное блокирование легитимных устройств.

4. Защита федеративного обучения от атак отравления (Фреймворк TQFL)

4.1. Постановка проблемы

FL, используемое для распределенного обучения моделей ИИ в 5G (например, для прогнозирования нагрузки или задержки) без обмена сырыми данными, уязвимо к атакам отравления данных и моделей со стороны вредоносных клиентов-инсайдеров [4, 11]. Это может привести к ухудшению глобальной модели или имплантации скрытого функционала (бэкдора). Проблема аналогична сложностям защиты самих систем EDR/XDR, чьи модели машинного обучения также являются целью атак [6].

4.2. Предлагаемое решение

Предложен многоуровневый фреймворк Trust and Quality-aware Federated Learning (TQFL).

1. Генерация набора данных EARCD: для реалистичных экспериментов создан набор данных о потреблении ресурсов AMF, эмулирующий поведение тысяч устройств, с использованием OAI и my5G-RANTester.

2. Динамический выбор доверенного участника: на центральном сервере FL работает агент Deep Q-Network (DQN, который в каждом раунде обучения выбирает одного клиента с наивысшей совокупной репутацией (на основе истории качества его обновлений) для выполнения роли «арбитра».

3. Механизм обнаружения отравления: Выбранный доверенный клиент применяет методы снижения размерности (PCA, LDA) и кластеризации (K-Means, KNN) к векторам градиентов или весов, полученным от всех клиентов. Обновления, образующие отдельный, удаленный кластер или яв-

ляющиеся статистическими выбросами относительно обновления доверенного участника, помечаются как потенциально вредоносные.

4. Механизм устранения: Сервер агрегирует (FedAvg) только те обновления, которые не были помечены как вредоносные. Репутация клиентов, чьи обновления были отфильтрованы, снижается.

4.3. Методология испытаний и результаты

Эксперименты проводились в симулированной среде с 10 клиентами, 1-3 из которых были вредоносными и осуществляли атаки отравления (label-flipping, Gaussian noise injection).

* **Эффективность обнаружения:** TQFL показал высокую эффективность в сепарации вредоносных обновлений. Визуализация векторов обновлений в 2D-пространстве после PCA наглядно подтвердила образование отдельных кластеров для атакующих данных.

* **Качество итоговой модели:** после активации фильтрации TQFL (с 4-го раунда обучения) среднеквадратичная ошибка (MSE) глобальной модели на валидационной выборке начинала устойчиво снижаться и сходилась к уровню, сопоставимому с сценарием обучения без атак.

* **Недостатки и ограничения:** а) Накладные расходы: Дополнительные этапы выбора арбитра, снижения размерности и кластеризации увеличивают вычислительную сложность и длительность раунда FL. б) Уязвимость доверенного арбитра: Если злоумышленник длительное время ведет себя «хорошо» и набирает высокую репутацию, он может быть выбран в качестве арбитра и саботировать процесс обнаружения. в) Эффективность против сложных атак: Метод может быть менее эффективен против скоординированных и слабых (low-and-slow) атак отравления, которые минимально искажают градиенты.

5. Объяснимое федеративное обучение для построения доверия (XAI-фреймворк)

5.1. Постановка проблемы

Сложные модели FL, используемые для управления сетью, действуют как «черный ящик» [5, 6]. Это препятствует доверию сетевых операторов, затрудняет диагностику сбоев и аудит решений, принятых ИИ, что критично для соответствия регуляторным требованиям. Проблема аналогична недостатку прозрачности в сложных системах XDR, где аналитику необходимо понимать логику срабатывания детекторов [6, 7].

5.2. Предлагаемое решение

В конвейер FL интегрированы методы объяснимого ИИ (XAI), адаптированные с учетом требований конфиденциальности данных FL.

1. Глобальная объяснимость (на стороне сервера): после агрегации глобальной модели строятся графики частной зависимости (PDP), показывающие, как изменение отдельного признака (нагрузка, утилизация CPU, количество запросов) в среднем влияет на прогноз (например, задержку).

2. Локальная объяснимость с агрегацией: Клиенты локально, на своих данных, вычисляют SHAP-значения (SHapley Additive exPlanations) или применяют LIME (Local Interpretable Model-agnostic Explanations) для объяснения индивидуальных прогнозов. На сервер отправляется не сырая информация, а агрегированная статистика (например, средняя или топ-N важности признаков по всем клиентам), дающая общее представление без нарушения конфиденциальности.

3. Интерпретация правил: к итоговой глобальной модели применяется алгоритм RuleFit, который генерирует набор простых, интерпретируемых правил вида «ЕСЛИ нагрузка > 400 запросов/сек и утилизация CPU > 85% ТОГДА прогнозируемая задержка > 20 мс». Эти правила могут быть напрямую использованы для создания понятных оповещений или ручных политик оркестрации.

5.3. Результаты и практическая ценность

Применение XAI-методов к модели прогнозирования задержки, обученной на наборе EARCD, позволило:

* **Выявить ключевые зависимости:** SHAP-анализ показал, что признаки `num_attach_requests` и `cpu_used` являются наиболее важными для прогноза, что соответствует сетевой интуиции.

* **Сгенерировать оперативные знания:** Правила, извлеченные с помощью RuleFit, были представлены операторам. Например, правило «ЕСЛИ `num_attach_requests` в окне > 500 ТОГДА риск высокой задержки > 0.8» позволяет proactively выделять дополнительные ресурсы для среза.

* **Недостатки и ограничения:** а) Вычислительная сложность: Расчет SHAP-значений, особенно для tree-based моделей, может быть ресурсоемким на клиентских устройствах с ограниченными мощностями. б) Агрегация объяснений: Потеря детальности при агрегации локальных объяснений может скрыть важные специфичные для клиента инсайты. в) Интерпретируемость vs. Точность: Существует

классический компромисс: наиболее интерпретируемые модели (линейные, rule-based) часто уступают в точности сложным ансамблевым моделям или нейронным сетям.

6. Заключение и перспективы интеграции

В работе представлен целостный, многоуровневый подход к обеспечению безопасности архитектуры нарезки сети 5G/B5G. Научный и практический вклад включает:

1. Децентрализованную блокчейн-архитектуру для управления доверием и автоматизации SLA в экосистеме с множеством заинтересованных сторон, решающую проблемы прозрачности и арбитража.

2. Автономную систему Zero-touch защиты от внутренних DDoS-атак в срезах mMTC, сочетающую статистические методы и машинное обучение для точного обнаружения и градации реагирования.

3. Защищенный фреймворк TQFL для FL, который комбинирует обучение с подкреплением для выбора доверенных участников и методы неконтролируемого обучения для отсева отравленных обновлений.

4. Методологию интеграции XAI в FL, обеспечивающую необходимую прозрачность и интерпретируемость решений ИИ для сетевых операторов.

Перспективные направления будущих исследований и интеграции:

* Конвергенция с системами безопасности предприятия: Исследование возможности интеграции предложенной автономной системы защиты срезов с корпоративными платформами XDR. Аномалии, обнаруженные в сетевом срезе, могут обогащать контекст расследования инцидентов безопасности на конечных точках (EDR), и наоборот.

* Развитие криптографических основ: Внедрение передовых криптографических протоколов (мульти-парти вычисления, гомоморфное шифрование) для обеспечения конфиденциальности данных в рамках TQFL и при работе с XAI-объяснениями.

* **Формальная верификация и HW-доверие:** Применение методов формальной верификации для доказательства изоляции срезов на уровне гипервизора и использование аппаратных корней доверия (TPM, TEE) для защиты IoT-устройств в mMTC-срезах и клиентов FL.

* **Объяснимость для детекторов угроз:** Расширение методологии XAI на сами модели обнаружения DDoS-атак и отравления, чтобы аналитик безопасности мог понять не только что было обнаружено, но и почему.

Предложенные решения демонстрируют, что комплексная безопасность автономных сетей 5G/B5G достижима за счет синергетической интеграции современных технологий: блокчейна, машинного обучения и объяснимого ИИ, формируя новый класс встроенных (infrastructure-native) систем безопасности.

Литература

- 3GPP TS 23.501. «Системная архитектура для системы 5G (Релиз 17)». V17.6.0, 2023.
- NGMN Alliance. «Описание концепции нарезки сети (Network Slicing)». Белая книга NGMN по 5G, 2016.
- Чандел С. Защита конечных точек: измерение эффективности технологий и методик устранения угроз для внутренних угроз // 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC). 2019. P. 81-89.
- МакМахан, Б., Мур, Э., Рамаж, Д., и др. Обучение глубоких сетей на децентрализованных данных с эффективной коммуникацией // В материалах 20-й Международной конференции по искусственному интеллекту и статистике (AISTATS), 2017.
- Лундберг С. М., и Ли С. И. Унифицированный подход к интерпретации прогнозов моделей // В материалах 31-й Международной конференции по нейронным информационным системам (NeurIPS), 2017.
- Яухайнен Х. Проектирование кибербезопасности области конечного пользователя для облачной организации. 2021. P. 52.
- Зейдан Э., Мангес-Бафалью Х. Последние достижения в области искусственного интеллекта для автоматизации сетей: Всесторонний обзор // IEEE Communications Surveys & Tutorials, 24(4), 2022.
- Крейнин В., Форте Д. Аспекты безопасности нарезки сети 5G: угрозы, проблемы и потенциальные решения // IEEE Security & Privacy, 19(3), 2021.
- Альянс OpenAirInterface Software Alliance. «OpenAirInterface: Платформа с открытым исходным кодом для 5G». <http://openairinterface.org>.
- ETSI. «Виртуализация сетевых функций (NFV); Управление и оркестрация». GS NFV-MAN 001 V1.1.1, 2014.

11. *Ли Х.* и др. Обзор систем федеративного обучения: видение, ажиотаж и реальность в отношении конфиденциальности и защиты данных // IEEE Transactions on Knowledge and Data Engineering, 2021.
12. Инициатива My5G. «my5G-RANTester: Эмулятор для 5G RAN и пользовательского оборудования (UE)». <https://github.com/my5G/my5G-RANTester>.
13. *Крейнделин В. Б., Авидзба А.Д.* Шифрование Wi-Fi protected access // Технологии информационного общества: XI Международная отраслевая научно-техническая конференция: сборник трудов, Москва, 15-16 марта 2017 г. М.: Издательский дом Медиа Паблишер, 2017. С. 294.
14. *Крейнделин В. Б., Варукина Л. А.* Проблема справедливого распределения мощности в системе PD-noma // T-Comm: Телекоммуникации и транспорт. 2025. Т. 19, № 12. С. 25-33. DOI 10.36724/2072-8735-2025-19-12-25-33. EDN YWTKAI.
15. *Bakulin M. G., Kreyndelin V. B., Reznov A. A.* Analysis of selection criteria for vector channels or orthogonal precoding matrices in communication systems with multiplexing // T-Comm: Телекоммуникации и транспорт. 2025. Vol. 19, No. 4, pp. 57-66. DOI 10.36724/2072-8735-2025-19-4-57-66. EDN VJRHNU.
16. *Бакулин М. Г., Бен Реджеб Т. Б. К., Крейнделин В. Б.* и др. Пространственная модель канала с кластеризованной линией задержки (CDL) для перспективных систем MIMO // T-Comm: Телекоммуникации и транспорт. 2025. Т. 19, № 8. С. 38-48. DOI 10.36724/2072-8735-2025-19-8-38-48. EDN GERBWW.
17. *Крейнделин В. Б., Варукина Л. А.* Обработка и прекодирование сигналов в системе с неортогональным доступом и с разделением по мощности // T-Comm: Телекоммуникации и транспорт. 2024. Т. 18, № 6. С. 38-45. DOI 10.36724/2072-8735-2024-18-6-38-45. EDN CMVGXV.
18. *Bakulin M. G., Kreyndelin V. B., Khazov M. L.* New quasi-optimal algorithms of antenna selection with low complexity // T-Comm: Телекоммуникации и транспорт. 2023. Vol. 17, No. 7, pp. 47-56. DOI 10.36724/2072-8735-2023-17-7-47-56. EDN SAQTDM.
19. *Бакулин М. Г., Бен Реджеб Т. Б. К., Крейнделин В. Б.* и др. Схемы модуляции для систем сотовой связи 5G/LM-T-2020 и 6G // T-Comm: Телекоммуникации и транспорт. 2022. Т. 16, № 3. С. 11-17. DOI 10.36724/2072-8735-2022-16-3-11-17. EDN MXYOEW.
20. *Бакулин М. Г., Бен Реджеб Т. Б. К., Крейнделин В. Б.* и др. Схемы NOMA с обработкой на уровне символов // T-Comm: Телекоммуникации и транспорт. 2022. Т. 16, № 5. С. 4-14. DOI 10.36724/2072-8735-2022-16-5-4-14. EDN NVXWQX.
21. *Бакулин М. Г., Бен Реджеб Т. Б. К., Крейнделин В. Б., Смирнов. А. Э.* Способы минимизации объема передаваемой информации в обратном канале многоантенных систем MIMO // T-Comm: Телекоммуникации и транспорт. 2021. Т. 15, № 3. С. 17-24. DOI 10.36724/2072-8735-2021-15-3-17-24. EDN HNJPPF.
22. *Бакулин М. Г., Крейнделин В. Б., Панкратов Д. Ю.* Применение технологии MIMO в современных системах беспроводной связи разных поколений // T-Comm: Телекоммуникации и транспорт. 2021. Т. 15, № 4. С. 4-12. DOI 10.36724/2072-8735-2021-15-4-4-12. EDN FPZEGW.
23. *Крейнделин В. Б., Фриск В. В., Степанова А. Г.* Моделирование электрических процессов на персональном лабораторном стенде // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2023. Т. 12, № 1. С. 72-75.

PYTHON-ОРИЕНТИРОВАННЫЙ ПОДХОД К РЕАЛИЗАЦИИ АНАЛИТИЧЕСКИХ ПРОЦЕССОВ ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ МАРКЕТИНГОВЫХ КАМПАНИЙ

Ерохин Андрей Густавович

Московский технический университет связи и информатики, доцент, к.т.н., Москва, Россия
andrew145@yandex.ru

Ванина Маргарита Федоровна

Московский технический университет связи и информатики, доцент, к.т.н., Москва, Россия
margo.vanina2012@yandex.ru

Фролова Елена Александровна

*Московский технический университет связи и информатики, старший преподаватель,
Москва, Россия*
e.a.frolova@mtuci.ru

Аннотация

Важнейшим условием повышения инвестиционной привлекательности компаний в условиях высокой конкуренции является использование передовых аналитических технологий. Маркетинговая аналитика является одной из таких технологий. Эффективное проведение маркетинговых кампаний требует разработки новых программных продуктов и инструментов. Одним из способов решения данной проблемы является автоматизация маркетинговой аналитики с использованием современных языков программирования и специализированных сервисов. В частности, Python-ориентированные решения позволяют ускорить процесс обработки данных, строить аналитические модели и визуализировать результаты анализа в интуитивно понятные графики и таблицы. Внедрение Python-ориентированных решений позволит не только автоматизировать аналитические процессы, но и улучшить точность прогнозов, стандартизировать расчеты и минимизировать влияние человеческого фактора. В перспективе это приведет к более обоснованному принятию бизнес-решений и увеличению конкурентоспособности организации. Целью настоящей статьи является анализ существующих методов проведения маркетинговых кампаний и разработка процедур их совершенствования на основе применения Python-ориентированного подхода.

Ключевые слова:

маркетинг, маркетинговая аналитика, Python, Python-ориентированный подход, анализ данных, CRM-системы

Введение

Неотъемлемой частью деятельности любой организации в настоящее время является маркетинговая аналитика [1]. Для повышения инвестиционной привлекательности в условиях высокой конкуренции компании внедряют передовые аналитические технологии, что позволяет им адаптироваться к непрерывно изменяющейся конъюнктуре рынка [2].

Однако современные подходы к маркетинговой аналитике имеют недостатки, связанные с качеством данных, ограничениями инструментов, нехваткой квалифицированных специалистов и этическими вопросами. Для принятия обоснованных маркетинговых решений необходимо все эти проблемы устранить. Это может быть достигнуто путем развертывания соответствующих программных продуктов. Для создания таких продуктов одним из самых эффективных языков программирования является Python.

Анализ современных подходов к анализу данных

В настоящее время сложилось четыре подхода к анализу данных [3]:

- описательная (дескриптивная) аналитика;
- диагностическая аналитика;
- предиктивная (прогнозная) аналитика;

- прескриптивная (предписывающая) аналитика.

Описательная аналитика является базовым и как правило, самым первым уровнем при анализе. Основная задача описательной аналитики состоит в ответе на вопрос: «Что произошло?» Такой подход включает в себя сбор и обработку исторических данных с целью выявления ключевых трендов, закономерностей и отклонений от нормы. Инструментарий описательной аналитики включает в себя агрегацию данных, сводные таблицы и различные формы визуализации. Например, компания может проанализировать продажи за предыдущий год, чтобы выявить сезонные пики спроса или определить наиболее востребованные продукты [4]. Системы бизнес-аналитики (BI), такие как Tableau, Power BI, играют здесь ключевую роль, обеспечивая удобный интерфейс для визуализации данных и мониторинга показателей [5]. Эти платформы позволяют визуализировать данные в виде графиков, диаграмм и дашбордов, что облегчает их интерпретацию для пользователя.

Однако описательная аналитика не дает полного понимания причинно-следственных связей, поскольку ограничивается лишь констатацией фактов.

Для более глубокого понимания сущности бизнес-процессов компании необходим переход к диагностической аналитике. Цель данного вида аналитики состоит в выявлении причинно-следственных связей и получении ответ на вопрос: «Почему это произошло?». Это возможно с помощью применения статистических методов для анализа взаимосвязей между различными переменными. Например, при снижении конверсии на сайте диагностическая аналитика поможет определить, связано ли это с изменениями в дизайне, рекламной кампании или внешними экономическими факторами [6]. Основными инструментами диагностической аналитики являются корреляционный анализ и сегментация данных. Корреляционный анализ позволяет оценить взаимосвязь между переменными, например, между бюджетом на рекламу и количеством лидов. Сегментация данных дает возможность разделить клиентов на группы по разным признакам для выявления особенностей каждого сегмента [7]. Диагностическая аналитика требует более серьезной статистической подготовки, но обеспечивает ценные инсайты для оптимизации бизнес-процессов компании.

Следующий уровень аналитики – предиктивная аналитика, отвечающая на вопрос: «Что произойдет в том или ином случае?» Предиктивная аналитика позволяет предприятиям прогнозировать будущие тенденции развития, а также потребности клиентов на основе предыдущих показателей компании. Для улучшения операционной деятельности, автоматизации рутинных задач и снижения нагрузки на сотрудников при использовании предиктивной аналитики компании все чаще используют искусственный интеллект. Внедрение искусственного интеллекта и использование машинного обучения позволяет с высокой скоростью обрабатывать и анализировать огромные объемы данных.

Положительное влияние также оказывает тенденция к персонализированному обслуживанию клиентов. Используя методы предиктивной аналитики, компании начинают лучше понимать, кто их клиент и как с ним правильно взаимодействовать. Такой подход обеспечивает конкурентное преимущество, что в конечном итоге способствует повышению выручки [4]. Процесс проведения предиктивной аналитики приведен на рис. 1.



Рис. 1. Процесс проведения предиктивной аналитики

Для разработки предиктивной модели на начальном этапе необходимо выявление цели прогнозирования. Например, компания хочет увеличить активную базу клиентов, предложив им скидки на свою продукцию. Узнав требования руководства, аналитический отдел собирает данные, возможно, из разных источников, необходимых для разработки модели. Затем неструктурированные данные преобразуются в структурированную форму для дальнейшей работы, после чего эти данные подвергаются проверке на соответствие качеству, поскольку эффективность прогностической модели полностью зависит от качества данных. Лишь после этого разрабатывается модель, основанная на использовании статистических методов и методов машинного обучения, в которую закладывается набор подготовленных данных. После разработки данная модель тестируется на тестовой выборке, чтобы проверить правильность ее работы. В случае успешного прохождения всех тестов модель считается подходящей, и после подгонки и корректировки может применяться для ежедневного прогнозирования и принятия решений. Предиктивная аналитика находит активное применение в банковском секторе, в частности, для автоматизации скоринговых задач с целью определения потенциальной платежеспособности клиентов [8-10].

Прескриптивная аналитика (от англ. Prescriptive Analytics) является наиболее продвинутым подходом к анализу данных, ориентированным на создание конкретных рекомендаций для принятия управленческих решений. Ключевой задачей данного уровня аналитики является ответ на вопрос: «Что необходимо предпринять для достижения желаемых результатов?». Например, если аналитика подсказывает, что определенный коммуникационный канал обладает самой большой конверсией, то прескриптивная аналитика будет давать рекомендации о перераспределении бюджета в пользу этого канала, чтобы максимизировать ROI (возврат на инвестиции). Основу прескриптивной аналитики составляют два ключевых метода: оптимизационные модели и рекомендательные системы. В качестве примера применения оптимизационной модели, можно рассмотреть линейное программирование, которое позволяет формализовать задачу оптимального распределения бюджета, учитывая различные факторы, такие как общий лимит, необходимый охват целевой аудитории и допустимая стоимость привлечения клиента. Данные модели особенно актуальны в задачах по оптимизации цепочек поставок и управлением запасами.

Вопросы прескриптивной аналитики исследуются, в частности, в работах [11, 12]. Дескриптивную и диагностическую аналитику можно отнести к классу традиционных аналитических методов. Предиктивная и прескриптивная аналитика являются продвинутыми методами аналитики. Эффективность проводимого анализа зависит от его уровня сложности. Традиционные методы отличает сравнительно невысокая сложность, однако и эффективность для бизнеса от их использования значительно ниже, чем от использования продвинутых методов.

В таблице 1 представлена сравнительная характеристика различных методов и подходов в аналитике.

Таблица 1

Сравнительная характеристика методов аналитики

Тип аналитики	Основные методы	Применение	Плюсы	Минусы
Описательная	Агрегация данных, сводные таблицы, визуализация	Анализ продаж, выявление трендов	Простота, доступность	Не объясняет причины событий
Диагностическая	Корреляционный анализ, сегментация	Определение факторов влияния на конверсии	Выявление причинно-следственных связей	Требует сложных расчетов
Предиктивная	Регрессионный анализ, машинное обучение	Прогнозирование спроса, оттока клиентов	Высокая точность прогнозов	Требует больших объемов данных
Прескриптивная	Оптимизационные модели, рекомендательные системы	Оптимизация бюджета, персонализация	Максимальная ценность для бизнеса	Высокая сложность внедрения

На рисунке 2 представлены виды аналитики в градации от их сложности и эффективности влияния на бизнес-процессы организации [13].

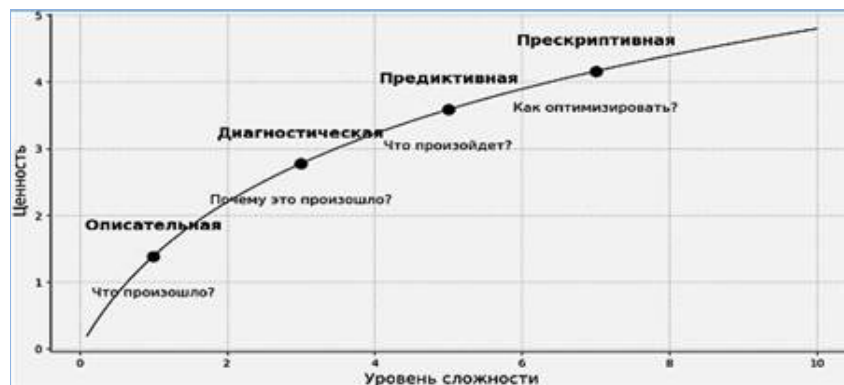


Рис. 2. Эффективность аналитических методов и их влияние на бизнес

Как видно из рисунка 2, эффективность того или иного метода анализа увеличивается в зависимости от уровня сложности – от описательной к прескриптивной аналитике.

Информационные системы маркетинговой аналитики

Проведение любых аналитических исследований невозможно без использования информационных систем и специализированного софта. Современные информационные системы маркетинговой аналитики включают несколько ключевых модулей [14, 15]:

- Системы управления базами данных (СУБД);
- BI-платформы;
- CRM-системы;
- Big Data и AI.

Использование баз данных в маркетинговой аналитике известно достаточно давно [16, 25-28]. В настоящее время в маркетинговой аналитике используются два основных вида баз данных:

– **Клиентские базы (B2C)** – используются компаниями, продающими продукт непосредственно потребителю. Таблицы в таких базах включают, как правило, следующие поля: имя, адрес, историю операций внутренних продаж, сервисов доставки или списки клиентов, приобретённые у других компаний.

– **Корпоративные базы (B2B)** – содержат специализированную информацию о компаниях, являющихся покупателями продуктов. Информация в таких базах данных, как правило, носит более детальный характер, по сравнению с клиентскими базами.

BI-платформы и инструменты визуализации данных позволяют отобразить все аналитические результаты на экране компьютера или бумаге. Такие платформы собирают информацию из различных источников, визуализируют ее в виде графиков, диаграмм и отчетов, что позволяет пользователю принимать обоснованные бизнес-решения.

Еще одним важнейшим компонентом любой крупной информационной маркетинговой системы является CRM, которая представляет собой модель взаимодействия с клиентами, основанную на теории о том, что клиенты являются центром всей философии любого бизнеса, а главными направлениями деятельности компании являются меры по обеспечению эффективного маркетинга, продаж и обслуживания клиентов. Поддержка данных бизнес-целей включает сбор, хранение и анализ информации о потребителях, поставщиках, партнёрах, а также о внутренних процессах компании. Функции для поддержки этих бизнес-целей включают продажи, маркетинг, поддержку потребителей.

Однако в условиях растущего объема данных, генерируемых бизнесом, традиционных инструментов анализа становится недостаточно. Здесь на помощь приходят технологии Big Data и искусственного интеллекта (AI). Big Data позволяет обрабатывать и анализировать огромные объемы структурированных и неструктурированных данных из различных источников: социальных сетей, датчиков, транзакций и т.д. AI, в свою очередь, использует машинное обучение (ML) и нейронные сети для выявления закономерностей, прогнозирования и автоматизации процессов [10, 17].

Эффективность информационной системы определяется уровнем её интеграции в бизнес-процессы компании. Современные маркетинговые платформы обеспечивают бесперебойное взаимодействие между CRM, BI-системами, базами данных и AI-алгоритмами. Это позволяет в режиме реального времени анализировать поведенческие паттерны клиентов, прогнозировать эффективность рекламных кампаний и автоматизировать сегментирование аудитории.

Анализ современных решений в области автоматизации аналитической деятельности

Эффективность аналитических процессов зависит от архитектуры информационных систем, обеспечивающих скорость обработки данных, точность интерпретации этих данных и автоматизацию бизнес-процессов. В условиях высокой конкуренции компании стремятся внедрять передовые инструменты анализа данных, позволяющие оперативно адаптироваться к изменениям конъюнктуры рынка.

Современные решения в области автоматизации аналитических процессов включают инструменты для работы с данными, платформы для машинного обучения, инструменты для автоматизации процессов извлечения, трансформации и загрузки данных (ETL), и сервисы с использованием искусственного интеллекта.

BI-системы собирают, анализируют и визуализируют данные из различных источников. К основным функциям таких систем относятся:

- генерация отчётов в различных форматах (таблицы, графики, диаграммы);
- создание интерактивных панелей мониторинга (дашбордов) для визуального представления ключевых показателей и тенденций;
- интеграция с экосистемами компании, объединение данных из ERP, CRM и других систем.

Примерами BI-систем могут служить решения от Microsoft Power BI и платформа с открытым кодом Apache Superset. Правда, в современных условиях их использование связано со значительными трудностями. Однако, в рамках реализации концепции импортозамещения, уже появились отечественные BI-системы премиум-уровня. К таким системам можно отнести системы Alpha BI [18] и PIX BI [19].

Алгоритмы машинного обучения анализируют текущие и архивные данные компании для выявления скрытых закономерностей, а также построения прогнозных моделей. Основные функции таких алгоритмов:

- Прогнозирование временных рядов, заключающееся в точном предсказании будущих трендов на основе архивных данных с учётом сезонности и внешних факторов.
- Сценарное моделирование, состоящее в анализе потенциальных исходов при различных условиях и параметрах.
- Обучение моделей с подкреплением (reinforcement learning), представляющее собой постоянное улучшение своих рекомендации на основе результатов ранее принятых решений.

ETL-инструменты автоматизируют процессы извлечения, трансформации и загрузки данных. К таким инструментам можно отнести облачные сервисы, а также коммерческие пакеты, такие, как IBM DataStage, Informatica PowerCenter, Oracle Data Integrator, SAP Data Services и инструменты с открытым кодом, например, Pentaho Data Integration (Kettle), Apache NiFi, Talend Open Studio, Scriptella. В рамках реализации концепции импортозамещения появились и отечественные технологии, такие, как OneData.

Нейросети в бизнес-анализе помогают автоматизировать процесс сбора и обработку данных, идентифицировать тренды и аномалии, а также создавать визуальные представления информации. Некоторые из таких сервисов:

- **GenAPI** – аналитическая платформа для анализа данных с помощью ИИ, предлагает инструменты визуализации данных с автоматической интерпретацией и систему прогнозирования на основе временных рядов.
- **AICont** – набор сервисов на базе ИИ для прогнозов поведения клиентов и финансовых показателей, включает анализ клиентского поведения (оценка вероятности действия) и финансовую аналитику (прогнозы выручки, затрат и прибыли).

Однако системы маркетингового анализа сталкиваются с рядом проблем: сложностью интеграции с базами данных, высокими вычислительными затратами и недостаточной автоматизацией про-

цессов. Эти факторы снижают скорость аналитики и оперативность принятия решений

Одним из способов решения данной проблемы является автоматизация маркетинговой аналитики с использованием современных языков программирования и специализированных сервисов. В частности, Python-ориентированные решения позволяют ускорить процесс обработки данных, строить аналитические модели и визуализировать результаты анализа в интуитивно понятные графики и таблицы.

Внедрение Python-ориентированных решений позволит не только автоматизировать аналитические процессы, но и улучшить точность прогнозов, стандартизировать расчеты и минимизировать влияние человеческого фактора. В перспективе это приведет к более обоснованному принятию бизнес-решений и увеличению конкурентоспособности организации.

Пример реализации CRM-системы

Рассмотрим схему работы CRM-системы (рис. 3) на примере некоторой торговой компании.

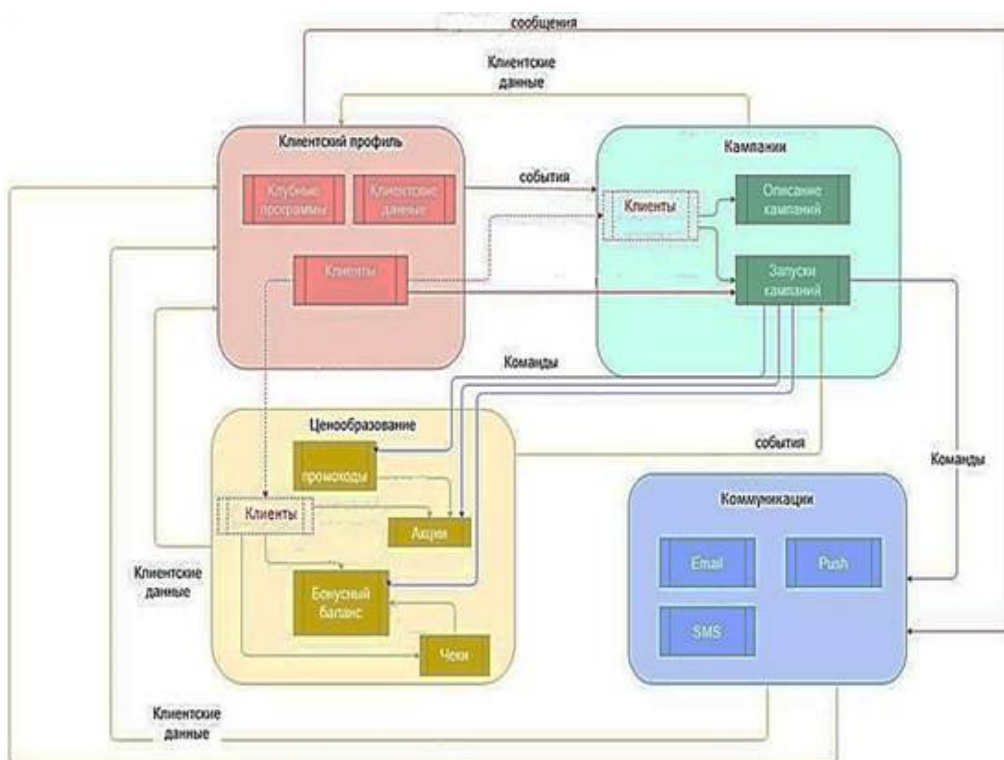


Рис. 3. Блок-схема работы CRM

Как видно из рисунка 3, в качестве основной системы управления взаимоотношениями с клиентами используется CRM-система, состоящая из четырех компонентов. Блок «Коммуникации» отвечает за отправку и контроль работы отправки коммуникаций (сервисные сообщения, бонусный баланс, акционные мероприятия). Блок «Клиентский профиль» обеспечивает хранение, обработку и поддержание в актуальном состоянии клиентские данные. Блок «Кампании» отвечает за создание и контроль исполнения маркетинговых кампаний. Наконец, блок «Ценообразование» производит расчет персональных скидок клиента на основании бонусов и действующих акций. Для хранения и обработки данных используется реляционная база данных.

Важнейшую роль в экосистеме работы компании играет отдел маркетинговой и клиентской аналитики. В его основные задачи входит преобразование данных в ценные инсайты (знания, помогающие находить корни проблем и пути их устранения) с целью оптимизации маркетинговой деятельности и улучшения клиентского опыта. Кроме мониторинга деятельности компании, важной задачей отдела является анализ эффективности проводимых маркетинговых мероприятий с целью оценки влияния на ключевые бизнес-показатели. Такой анализ охватывает широкий спектр метрик, включая как финансовые показатели (продажи, кол-во чеков, кол-во уникальных клиентов), так и маркетинго-

вые метрики (конверсия, трафик, вовлеченность).

Процесс работы рядового аналитика представлен на рисунке 4, в соответствии с которым, процесс аналитической работы начинается с инициации запроса, исходящего от заказчика. Как правило в лице заказчика выступает отдела маркетинга. Данный запрос обычно оформляется в виде официального письма или заявки через Jira – систему управления проектами. Запрос представляет собой описание конкретной бизнес-задачи или проблемы включающий описание целей кампании, ожидаемые ключевые показатели эффективности и сроки проведения проекта.

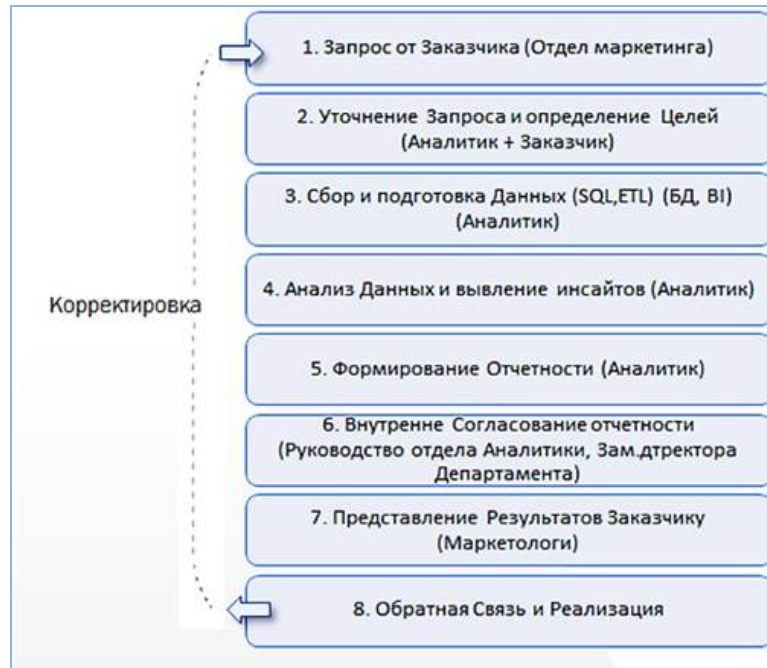


Рис. 4. Процесс работы аналитического отдела

Следующим этапом работы аналитика является уточнение запроса и определение целей маркетинговой компании. Для этого организуется встреча аналитика, ответственного за конкретное направление работы компании, с представителями отдела маркетинга. В ходе встречи происходит детальное обсуждение бизнес-задачи, формулируются и уточняются цели аналитического исследования, ожидаемые результаты и предварительная форма представления отчетности.

Сбор данных начинается с определения аналитиком необходимых источников данных, способных предоставить информацию для ответов на поставленные аналитические вопросы. В большинстве случаев основным источником структурированных данных выступает база данных CRM, содержащая детальную информацию о клиентах, их потребительском поведении и истории заказов. В зависимости от специфики задачи, приходится извлекать данные из BI-систем, а также других мест хранения данных. Иногда для этого привлекаются сотрудники других отделов компании.

На этапе анализа данные очищаются и трансформируются, что подразумевает устранение ошибок и дубликатов, нормализацию форматов данных, стандартизацию значений и трансформацию данных в формат, пригодный для дальнейшей обработки.

Затем на основе полученных инсайдов аналитик приступает к формированию отчетности, целью которой является представление результатов заказчику в понятной и наглядной форме. Отчетность согласовывается в соответствии с внутренними регламентами компании. На первом этапе согласования руководитель группы/отдела осуществляет проверку работы, оценивая правильность данных, логику выводов и соответствие бизнес-требованиям. После устранения замечаний руководителя, отчет передается на финальное согласование начальнику отдела или же заместителю директора департамента, где оценивается соответствие результатов стратегическим целям компании, а также презентабельность отчета. Только после этого результаты передаются заказчику и обрабатываются обратные связи.

Основным инструментом для получения данных и проведения анализа является язык запросов SQL. Специалисты отдела аналитики разрабатывают SQL-запросы для извлечения необходимых данных из различных таблиц и витрин базы данных, содержащих информацию о продажах, маркетинговых активностях, трафике, поведении клиентов и других релевантных показателях.

При этом процесс работы аналитического отдела сталкивается с рядом существенных ограничений, снижающих эффективность и оперативность. Среди ключевых проблемных зон можно выделить следующие проблемы:

- Высокая трудоемкость и временные затраты. Процесс анализа эффективности маркетинговых кампаний характеризуется значительными трудозатратами. Для каждой новой маркетинговой кампании, аналитикам необходимо заново разрабатывать SQL-код, адаптируя запросы к специфике конкретной акции и набору данных. Большое количество источников хранения информации существенно увеличивает время, необходимое для поиска, извлечения и подготовки данных к анализу.
- Большая нагрузка на сервера и базы данных. За счет большого числа однотипных SQL-запросов со множеством конструкций join система перегружается и возрастает нагрузка на вычислительные процессы.
- Ограниченная точность и субъективность оценки. В значительной степени расчет эффективности той или иной акции опирается на экспертную оценку аналитиков и использование различных коэффициентов, которые могут быть субъективными и не всегда адекватно отражать реальное влияние акции. Отсутствие автоматизированных моделей для оценки эффективности приводит к потенциальным погрешностям и ограничивает возможности для точной и детализированной аналитики, поэтому часто руководство компании получает искаженные результаты.

Обоснование необходимости разработки Python-ориентированных решений

Выявленные ограничения текущего процесса работы аналитического отдела демонстрируют необходимость модернизации аналитической инфраструктуры и внедрения новых, более эффективных инструментов.

Главная цель модернизации аналитической инфраструктуры заключается в создании комплекса решений, которые позволят:

- Повысить скорость и эффективность аналитических процессов: автоматизировать рутинные операции сбора, обработки и подготовки данных, сократить время от возникновения запроса до получения результатов, и освободить ценные ресурсы аналитиков для более интересных и важных задач.
- Улучшить объективность и точность оценки эффективности: перейти от субъективных экспертных оценок к объективным, научно обоснованным методам анализа, основанным на статистическом моделировании, что позволит принимать более взвешенные и эффективные решения в контексте маркетинговых исследований.
- Упростить доступ к аналитическим данным: сделать аналитические результаты понятными и доступными для широкого круга пользователей, способствуя продвижению data-driven культуры и принятию решений на основе данных на всех уровнях управления.

Для достижения поставленной цели, необходимо решить ряд задач, которые лягут в основу разработки Python-ориентированной аналитической платформы:

- Создание универсальной витрины данных, агрегирующей в себе основную информацию из таблиц в БД для нужд аналитики.
- Разработку специализированной платформы для прогнозирования и оценки эффективности маркетинговых акций.
- Разработку приложения для решения типовых аналитических задач и визуализации отчетности, включая расчет основных маркетинговых метрик и подсчет объема сегментов для рассылок. Это позволит стандартизировать аналитические процессы, повысить их эффективность и сократить кол-во ошибок.
- Внедрение дополнительных механизмов оптимизации, включая кеширование данных, контейнеризацию веб-приложений и мониторинг.

Для оценки успеха внедрения Python-ориентированной аналитической платформы и достижения поставленных целей оптимизации необходимо определить ключевые метрики эффективности (KPI), которые будут измерять прогресс и результативность разработанных решений:

- Сокращение затрачиваемого времени на аналитическое исследование. Данная метрика характеризуется измерением времени от момента получения запроса на анализ до момента предоставления результатов заказчику (отделу маркетинга).
 - Сокращение время выполнения SQL-запросов.
 - Уменьшение нагрузки на вычислительные системы.
- Далее более детально рассмотрим решение данных задач.

Разработка универсальной витрины данных

Назначение витрины данных (data-mart) заключается в предоставлении пользователям уже подготовленной и структурированной информации, которая агрегируется и собирается из различных источников. Вместо того, чтобы каждый раз писать сложные SQL-запросы, соединяя множество таблиц, аналитик может обратиться к витрине данных и получить всю необходимую информацию из одной, заранее подготовленной таблицы. Это радикально упрощает процесс анализа и экономит массу времени и усилий, при этом снижая риски допущения ошибок [20].

Другим очень важным преимуществом витрин данных является снижение нагрузки на операционные системы [21]. Это связано с тем, что витрины данных разгружают прикладные решения от задач поиска и подбора нужной информации. Разработка витрины данных включает в себя следующие процессы:

- вставка новых данных в витрину (рис. 5);
- создание направленного ациклического графа (DAG), который будет выполнять обновление данных на основе расписания (рис. 6).

```
# Вставка новых данных в таблицу CHECK_FULL
def insert_new_data(last_update_date):
    connection = get_db_connection()
    cursor = connection.cursor()

    # Запрос для вставки новых данных в CHECK_FULL
    query = f"""
    INSERT INTO CHECK_FULL (
        order_code, order_item_code, purchase_order_date, store_number,
        customer_unique_code, product_code, product_name, product_sebestoimost,
        product_price, order_total_amount
    )
    SELECT o.ORDER_ID, oi.ORDER_ITEM_ID, o.ORDER_DATE, o.STORE_ID, c.CUSTOMER_ID,
        oi.PRODUCT_ID, p.PRODUCT_NAME, p.SEBESTOIMOST, p.PRICE, o.ORDER_TOTAL_AMOUNT
    FROM
        ORDERS o
    JOIN ORDER_ITEMS oi ON o.ORDER_ID = oi.ORDER_ID
    JOIN PRODUCTS p ON oi.PRODUCT_ID = p.PRODUCT_ID
    JOIN CUSTOMERS c ON o.CUSTOMER_ID = c.CUSTOMER_ID
    WHERE o.ORDER_DATE > TO_DATE('{last_update_date}', 'YYYY-MM-DD')
    """

    cursor.execute(query)
    connection.commit() # Подтверждаем изменения
    cursor.close()
    connection.close()
```

Рис. 5. Вставка новых данных в витрину данных

```

# Создаем DAG для планирования процесса обновления данных
dag = DAG(
    'refresh_check_full_data',
    description='Инкрементное обновление данных для CHECK_FULL',
    schedule_interval='0 3 * * *', # Запуск каждый день в 3:00 утра
    start_date=datetime(2023, 1, 1),
    catchup=False
)

# Оператор для запуска функции обновления данных
refresh_data_task = PythonOperator(
    task_id='refresh_check_full_data_task',
    python_callable=refresh_check_full_data,
    dag=dag
)

# Запуск DAG
refresh_data_task
    
```

Рис. 6. Создание DAG для планирования процесса обновления

Разработка специализированной платформы для прогнозирования и оценки эффективности маркетинговых акций

Для такой разработки необходимо правильно оценить объем выборки целевой и контрольной группы клиентов, среди которых проводится та или иная маркетинговая акция. Это можно сделать путем разработки калькулятора для расчета оптимального объема групп в рассылках.

Цель создания данного инструмента – помочь аналитикам оценить и спрогнозировать эффективность рекламных акций за счет определения статистической значимости отклика и определения минимально значимого эффекта (MDE) [22], а также правильно распределить затраты на маркетинговую кампанию. С его помощью можно рассчитать, требуемый размер целевой и контрольной группы для получения достоверных результатов. Это позволяет оптимизировать размер сегмента, исключив лишние расходы и эффективно использовать бюджет.

Данный процесс можно реализовать в несколько шагов:

- Импорт соответствующих библиотек.
- Ввод данных для расчета (сегмент клиентов, продолжительность акции, бюджет кампании и каналы коммуникации).
- Создание датафрейма, который будет содержать результаты расчетов для каждого канала коммуникации (с такими полями, как название канала, прогнозируемый и фактический отклик целевой и контрольной групп, чистый отклик, размер сегмента, объем контрольной группы, максимальная допустимая доля контрольной группы и ее предельный размер).
- Вычисление ключевых параметров для каждого канала, включая фактический и прогнозируемый отклик контрольной группы, а также максимальный размер контрольной группы в зависимости от заданной доли выборки (рис. 7).

```
# Таблица для хранения результатов
tab = pd.DataFrame(columns=['Канал', 'Отклик ЦГ(прогноз)', 'Отклик КГ(факт)', 'Чистый отклик', 'Размер сегмента',
                           'Размер КГ', 'Максимальная доля КГ, %', 'Максимальный размер КГ'])

su, summ = 0, 0

# Для каждого канала выполняем расчет
for channel in channels:
    # Функция для расчета отклика и других показателей
    base_cl, sale_cl, res_cg, pred_resp, size_cg, max_cg = calc_response2(df, channel)

    # Вычисляем предполагаемый отклик
    res_tg = calculate_mde(base_cl, size_cg, res_cg)

    # Заполняем результаты в таблицу
    tab_ = {
        'Канал': channel,
        'Отклик ЦГ(прогноз)': f'{res_cg:.3%}',
        'Отклик КГ(факт)': f'{res_tg:.3%}',
        'Чистый отклик': f'{(res_cg - res_tg):.3%}',
        'Размер сегмента': f'{base_cl:,}',
        'Размер КГ': f'{max_cg:,}',
        'Максимальная доля КГ, %': f'{size_cg:.0%}',
        'Максимальный размер КГ': f'{max_cg:,}'
    }

    # Добавляем текущий результат в итоговую таблицу
    tab = pd.concat([tab, pd.DataFrame(tab_, index=[0])], ignore_index=True)
    summ += max_cg

# Выводим результат
print(tab)

# Если затраты превышают бюджет, выводим сообщение
if summ > budg:
    print(f'Бюджет = {budg:,.0f}, затраты = {summ:,.0f}. Превышение бюджета.'
```

Рис. 7. Вычисление ключевых параметров для каждого канала

Заключительным этапом является определение минимального детектируемого эффекта (MDE). Этот показатель позволяет определить, какое минимальное изменение в поведении тестовой группы можно считать статистически значимым. Это значение можно определить из следующего выражения:

$$MDE = \left(Z_c \cdot \sqrt{\frac{p \cdot (1-p)}{n}} \right) + \left(Z_p \cdot \sqrt{\frac{p \cdot (1-p)}{n}} \right)$$

где:

- Z_c – критическое значение Z для уровня значимости;
- p – базовый коэффициент конверсии;
- n – размер выборки;
- Z_p – значение Z для желаемой статистической мощности.

Полученные результаты передаются в функцию проверки бюджета, где оцениваются затраты на проведение кампании. Если расчетный объем выборки превышает допустимый бюджет, то выводим сообщение о его превышении.

Разработка приложения для решения типовых аналитических задач

Среди типовых аналитических задач важнейшее значение имеет задача анализа эффективности маркетинговых кампаний. Для этого необходимо создание специального решения, которое автоматически рассчитывает ключевые показатели по проводимым акциям: количество чеков, общие продажи, продажи акционного товара, число клиентов, использовавших промокоды / бонусы.

Также требуется специальная разработка модуля оценки эффективности каналов коммуникации, который рассчитывает конверсии по различным каналам взаимодействия с клиентами, который позволяет оценить, сколько сообщений было отправлено, какая доля из них была открыта и сколько пользователей совершили покупку после получения сообщения. Наконец, следует рассчитать процент отписок, что помогает скорректировать стратегии рассылок и минимизировать негативный от-

клик от клиентов.

Оценка эффективности маркетинговых кампаний может быть визуально проведена с помощью дашборда с использованием следующих фильтров:

- диапазон дат;
- выбор магазинов;
- агрегация данных по дням, неделям, месяцам;
- сравнение с аналогичным периодом прошлого года;
- возможность отображения метеоданных (температура воздуха).

Технология визуализации данных с помощью дашборда представлена на рис. 8.

```
st.sidebar.header("Фильтры")
start_date = st.sidebar.date_input("Начальная дата")
end_date = st.sidebar.date_input("Конечная дата")
store_input = st.sidebar.text_input("Список магазинов (через запятую)", "1, 2")
store_list = [s.strip() for s in store_input.split(",") if s.strip()]
```

Рис. 8. Визуализация данных маркетинговой компании

Такая визуализация позволяет:

- отображать данных по промокодам и конверсии продаж (рис. 9);
- отображать линейные графики текущих продаж (рис. 10).

```
# --- Расчет конверсии и отписок ---
df_channels['Конверсия (%)'] = (df_channels['clicked'] / df_channels['opened'] * 100).fillna(0)
df_channels['Отписка (%)'] = (df_channels['unsubscribed'] / df_channels['opened'] * 100).fillna(0)

# --- График конверсии по каналам ---
st.subheader("Конверсия по каналам")
fig_channels = px.bar(
    df_channels,
    x="channel_name",
    y="Конверсия (%)",
    title="Конверсия по каналам",
    color="channel_name",
    text="Конверсия (%)"
)
fig_channels.update_layout(template="plotly_white")
st.plotly_chart(fig_channels, use_container_width=True)

# --- График отписки по каналам ---
st.subheader("Отписка по каналам")
fig_unsub = px.bar(
    df_channels,
    x="channel_name",
    y="Отписка (%)",
    title="Отписка по каналам",
    color="channel_name",
    text="Отписка (%)"
)
fig_unsub.update_layout(template="plotly_white")
st.plotly_chart(fig_unsub, use_container_width=True)

# --- Интерактивная таблица с данными ---
st.subheader("Данные по каналам")
st.dataframe(df_channels.style.highlight_max(color="lightgreen"))
```

Рис. 9. Визуализация данных по промокодам и отображение конверсии по каналам

```
st.subheader("Динамика продаж")
fig = go.Figure()
fig.add_trace(go.Scatter(x=df_main["date"], y=df_main["sales"], mode='lines', name="Продажи"))
st.plotly_chart(fig, use_container_width=True)
```

Рис. 10. Визуализация процесса создания графиков текущих продаж

Разработка приложения предиктивной аналитики для расчета эффективности маркетинговых кампаний

Оценка эффективности маркетинговых кампаний является ключевой задачей в стратегическом управлении бизнесом, позволяя измерять влияние рекламных активностей на ключевые показатели деятельности организации. Как уже говорилось ранее, одним из наиболее широко применяемых методов для оценки эффекта кампании является А/В тестирование [23]. Однако в ряде случаев этот ме-

тод оказывается неприменимым, поэтому в качестве альтернативы можно использовать метод Causal Impact [24].

Первым этапом разработки является подготовка архивных данных о продажах (рис. 11).

```
import oracledb
import pandas as pd

# Функция загрузки данных по кампании
def load_campaign_data(campaign_name, start_date, end_date):
    conn = get_connection()

    query = f"""
    SELECT purchase_order_date, order_code, payment_method,
           marketing_channel_name, order_channel
    FROM check_full
    WHERE marketing_campaign_name = '{campaign_name}'
           AND purchase_order_date BETWEEN TO_DATE('{start_date}', 'YYYY-MM-DD')
           AND TO_DATE('{end_date}', 'YYYY-MM-DD')
    """

    df = pd.read_sql(query, conn)
    conn.close()

    return df
```

Рис. 11. Выгрузка данных по продажам и кампаниям

Для оценки эффективности маркетинговых кампаний важно учитывать влияние внешних факторов, не связанных напрямую с самой кампанией, но оказывающих влияние на поведение покупателей и объемы продаж. Без учета этих факторов можно ошибочно полагать, что изменения в продажах были вызваны непосредственно планируемой акцией, что может привести к неверным выводам. Для получения более точных результатов, в модель необходимо включить ковариаты – переменные, объясняющие динамику целевого показателя, но не подверженные влиянию самой кампании. Выбор ковариат основывается на специфике исследуемой акции, но практически всегда, независимо от рода акции, необходимо учитывать такие факторы как: средняя температура (рис. 12) и календарь праздничных и выходных дней (рис. 13).

```
import requests
from datetime import datetime

# Функция загрузки температуры по датам
def get_temperature_data(start_date, end_date, city="Moscow"):
    url = f"https://api.weatherapi.com/v1/history.json?key=YOUR_API_KEY&q={city}&dt={start_date}"

    response = requests.get(url).json()
    temp_data = [
        {"date": datetime.strptime(day["date"], "%Y-%m-%d"),
         "temperature": day["day"]["avgtemp_c"]}
        for day in response["forecast"]["forecastday"]
    ]

    df_temp = pd.DataFrame(temp_data)
    return df_temp
```

Рис. 12. Выгрузка данных о средней температуре

```
import holidays

# Функция загрузки праздничных дней
def get_holidays(start_date, end_date):
    ru_holidays = holidays.Russia()
    date_range = pd.date_range(start=start_date, end=end_date)
    holiday_flags = [1 if date in ru_holidays else 0 for date in date_range]

    return pd.DataFrame({"date": date_range, "holiday": holiday_flags})
```

Рис. 13. Выгрузка данных о праздничных днях

Полученные данные о праздничных днях и погоде затем добавляются в общий датасет, содержащий информацию о продажах (рис. 14).

```
def prepare_data_for_causal_impact(campaign_name, start_date, end_date):
    sales_data = load_campaign_data(campaign_name, start_date, end_date)
    temp_data = get_temperature_data(start_date, end_date)
    holidays_data = get_holidays(start_date, end_date)

    # Объединяем в один датафрейм
    df = sales_data.merge(temp_data, left_on="purchase_order_date", right_on="date", how="left")
    df = df.merge(holidays_data, left_on="purchase_order_date", right_on="date", how="left")

    df.drop(columns=["date_x", "date_y"], inplace=True) # Убираем дублирующиеся колонки

    return df
```

Рис. 14. Агрегация ковариат в общий набор данных

После проведения всех подготовительных вычислений возможно перейти непосредственно к анализу. Для этого следует ввести соответствующие временные периоды, а затем, используя соответствующие библиотеки, провести сам анализ.

Примеры результатов анализа представлены на рис. 15-18.

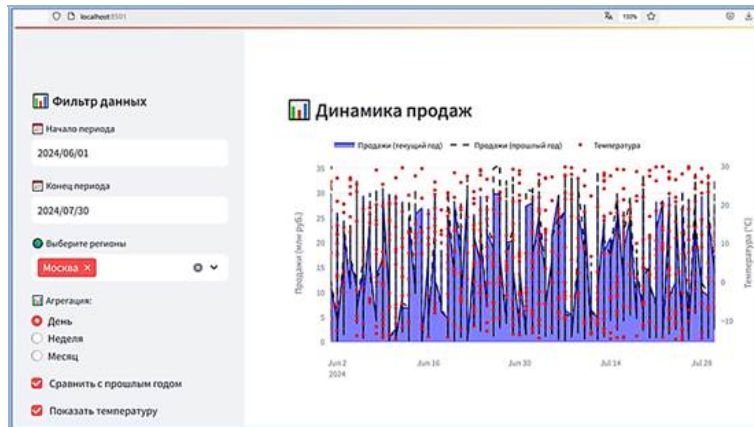


Рис. 15. Оценка динамики продаж

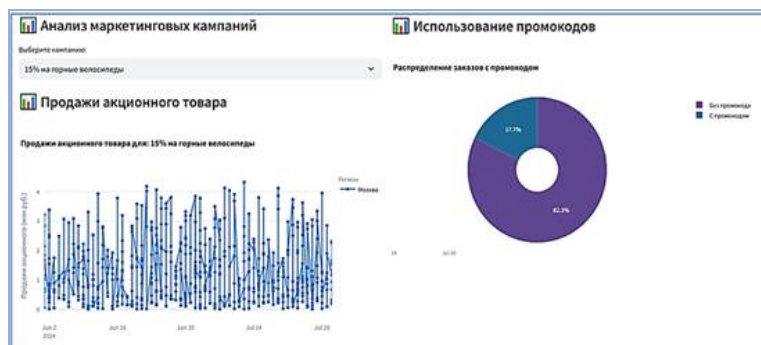


Рис. 16. Анализ маркетинговых кампаний

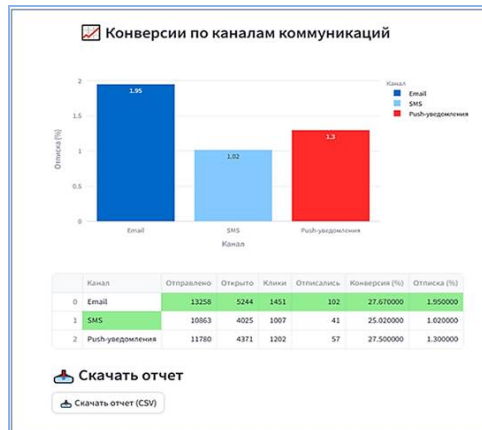


Рис. 17. Анализ конверсии по каналам коммуникации



Рис. 18. Анализ эффективности маркетинговой компании с помощью методов предиктивной аналитики

Заключение

Таким образом, полученные результаты позволяют определить основные принципы работы информационных систем, их роль в аналитических процессах, а также методы оптимизации их архитектуры. Применение Python-ориентированных решений позволяет существенно повысить эффективность процессов маркетинговой аналитики, улучшить доступность данных и сократить затраты времени на аналитические процессы. Важнейшим финансовым результатом этого является оптимизация бюджета, выделенного на маркетинговые мероприятия в пользу наиболее успешных.

Кроме этого, предприятия получают возможность значительно точнее измерять результаты маркетинговых кампаний, выделять наиболее эффективные маркетинговые стратегии и корректировать менее эффективные. Появляются новые алгоритмы выявления паттернов поведения клиентов, выявления их потребностей и предпочтений, что, в свою очередь, способствует более точной ориентации маркетинговых продуктов на конкретного клиента. Это, в свою очередь, дает возможность выявить точки роста, которые ранее были не видны, и сформировать более понятную и обоснованную бизнес-стратегию компании в целом.

Литература

1. *Германчук А.Н., Бирюченко Е.А.* Маркетинговая аналитика: специфика и значение в условиях цифровизации экономики // Экономика: вчера, сегодня, завтра. 2023. Том 13. № 11А. С. 600-607. DOI: 10.34670/AR.2023.26.32.065.
2. *Ерохина Ю.А.* Влияние институтов развития на достижение доли инвестиций в ВВП // Реформы в России и проблемы управления – 2024. Материалы 39-й Всероссийской научной конференции молодых ученых. Москва, 2024. С. 16-19.
3. *Чернова Е.С.* Уровни аналитики данных // Вестник магистратуры. 2022. № 12-6(135). С. 18-19.
4. *Алексеев И.Н.* Применение Python для анализа данных и машинного обучения. М.: ДМК Пресс, 2019
5. *Васильченко А.М.* Как проводить анализ данных при помощи Python? // Инновации и инвестиции, 2023. КиберЛенинка. Режим доступа: <https://cyberleninka.ru/article/n/kak-provodit-analiz-dannyh-pri-pomoschi-python>. (дата обращения 27.11.2025).
6. *Гаврилов П.В.* Маркетинговая аналитика: инструменты и технологии. СПб.: Питер, 2021.
7. *Громов И.А.* Базы данных в бизнес-аналитике. СПб.: Питер, 2021.
8. *Ерохин А.Г., Стуколова А.А., Стуколов С.С.* Роль скоринга в управлении маркетинговыми кампаниями предприятия // Труды международной научно-технической конференции «Телекоммуникационные и вычислительные системы – 2020». М.: Горячая линия – Телеком, 2020. С. 754-760.
9. *Ванина М.Ф., Ерохин А.Г., Фролова Е.А.* Скоринг как инновационный инструмент маркетинга // Системы синхронизации, формирования и обработки сигналов, №2-2023. С. 4-12.
10. *Ванина М.Ф., Ерохин А.Г.* Применение методов машинного обучения в банковском скоринге // DSPA: Вопросы применения цифровой обработки сигналов. 2025. Т. 15. № 1. С. 17-35
11. *Basu, Atanu.* "Five pillars of prescriptive analytics success". 2019. The Analytics Journey. doi:10.1287/LYTX.2013.02.07
12. Prescriptive Analytics [Электронный ресурс] : Definition. Gartner Information Technology Glossary, 2022. URL: <https://www.gartner.com/en/information-technology/glossary/prescriptive-analytics> (дата обращения: 26.11.2025).
13. Четыре вида аналитики данных: дескриптивная, диагностическая, предиктивная, прескриптивная / Хабр [Электронный ресурс]. URL: <https://habr.com/ru/articles/860078/?ysclid=mifoh0dcrq19027010> (дата обращения 26.11.2025).
14. *Юрьев В.Н., Кульков И.А.* Информационные системы в маркетинговой деятельности // Прикладная информатика, № 3, 2006. С. 3-13.
15. *Эрханов Ш.* Маркетинговые информационные системы и описание их содержания // Вестник науки №4 (61). Т. 5. С. 122-124. 2023. ISSN 2712-8849: <https://www.вестник-науки.pf/article/7922> (дата обращения: 26.11.2025 г.)
16. *Shaw R., Stone M.* Database Marketing. New York: John Wiley & Sons, 1988. 224 p.
17. *Ванина М.Ф., Ерохин А.Г.* Повышение эффективности бизнеса компании на основе технологий Big Data и Machine Learning // Технологии информационного общества. Сборник трудов XIV Международной отраслевой научно-технической конференции «Технологии информационного общества». (18-19 марта 2020 г. Москва, МТУСИ). М.: ИД Медиа Паблишер, 2020. С. 336-338.
18. Платформа бизнес-аналитики Alpha BI | Конструктор корпоративных аналитических решений [Электронный ресурс]. URL: <https://bars-alpha.bi/> (дата обращения 26.11.2025)
19. PIX BI – self-service система для анализа и визуализации данных [Электронный ресурс]. URL: <https://pix.ru/> (дата обращения 26.11.2025).
20. *Туманов В.Е.* Проектирование хранилищ данных для систем бизнес-аналитики. М.: Интернет университет информационных технологий; Бином; Лаборатория знаний, 2010. 615 с.
21. *Горбачев Д. В., Кононова М. В.* Подход к организации электронного взаимодействия посредством витрин данных // Интеллект. Инновации. Инвестиции. 2012. № Спецвыпуск 1. Оренбург, 2012. С. 83-90.
22. MDE vs. MDM: Understanding the Key Differences – DEV Community [Электронный ресурс]. URL: https://dev.to/mastech_digital/mde-vs-mdm-understanding-the-key-differences-2gce (дата обращения 27.11.2025).
23. *Kohavi Ron, et al.* Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing. Сингапур, Cambridge University Press, 2020.
24. *Pearl J., Glymour M., Jewell N. P.* Causal inference in statistics: a primer. John Wiley & Sons. 2016.
25. *Kirov D.E., Toutova N.V., Vorozhtsov A.S., Andreev I.A.* Feature selection for predicting live migration characteristics of virtual machines // T-Comm: Телекоммуникации и транспорт. 2021. Т. 15. № 7. С. 62-70. EDN: AGGBDW
26. *Тутов А.В., Тугова Н.В., Ворожцов А.С., Андреев И.А.* Многокритериальная оптимизация размещения виртуальных машин по физическим серверам в облачных центрах обработки данных // T-Comm: Телекоммуникации и транспорт. 2021. Т. 15. № 1. С. 28-34. EDN: IOFQSS
27. *Губин А.С., Тугова Н.В.* Анализ подхода к разработке приложений с "чистой" архитектурой // Телекоммуникации и информационные технологии. 2022. Т. 9. № 1. С. 28-37. EDN: NOZMKG
28. *Ванина М.Ф., Ерохин А.Г.* Экономическое образование и искусственный интеллект: аспекты преподавания и применения // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2025. Т. 14, № 1. С. 15-22. EDN YEYYWY.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ МЕЖДУ ВЫЧИСЛИТЕЛЬНЫМИ УСТРОЙСТВАМИ

Гадасин Денис Вадимович

МТУСИ, заместитель заведующего кафедры СИТиС, к.т.н., доцент, Москва, Россия
dengadiplom@mail.ru

Родина Алина Андреевна

МТУСИ, бакалавр группы БСТ2201, Москва, Россия,
alina_rodina2004@mail.ru

Яковенко Наталья Викторовна

МТУСИ, ст. преподаватель кафедры СИТиС, Москва, Россия
nv1906.iakovenko@yandex.ru

Сурова Мария Андреевна

МТУСИ, бакалавр группы БСТ2201, Москва, Россия,
surovamma@mail.ru

Аннотация

В связи с тем, что за последние годы нагрузка на распределенные и высоконагруженные системы значительно выросла, то становится необходимо определить наиболее эффективный алгоритм распределения нагрузки для обслуживания очередей задач в динамических кластерах с большим количеством серверов. Поэтому в работе проводится сравнительный анализ наиболее используемых алгоритмов балансировки. Сравниваются параметры алгоритмической сложности, потребления памяти.

Ключевые слова

балансировка нагрузки, алгоритмы балансировки, анализ алгоритмов, параметры балансировки, Least Connections, Round Robin, Weighted Round Robin, красно черное дерево

Введение

За последние годы нагрузка на распределенные и высоконагруженные системы значительно выросла. Например, одна из компаний, участвовавших в конференции HighLoad++ сообщила о необходимости обрабатывать до 70000 запросов в секунду, именно в последние несколько лет наблюдался особенно интенсивный рост нагрузки. Особенно важной оказывается проблема эффективного управления ресурсами. С развитием облачных технологий требования к системам балансировки нагрузки существенно возросли. Современные веб-сервисы и мобильные приложения обрабатывают огромные объемы данных в реальном времени, при этом нагрузка на систему может изменяться непредсказуем. В таких условиях традиционные алгоритмы балансировки, основанные на фиксированных правилах распределения нагрузки, часто оказываются неэффективны. Эта тема также поднимается в источниках [1-3]. Цель работы заключается в выявлении наиболее эффективного алгоритма распределения нагрузки для обслуживания очередей задач в динамических кластерах с большим количеством серверов.

Алгоритмы балансировки.

Среди множества существующих подходов к балансировке нагрузки особенно распространены три классических алгоритма: Round Robin (RR), Least Connections (LC) и Weighted Round Robin (WRR). Они также упомянуты в источниках [4-9].

Алгоритм Round Robin – циклический перебор, представляет собой простейший метод циклического распределения запросов между доступными серверами. Его работа основана на принципе строгой очередности, при котором каждый новый запрос направляется на следующий сервер в заранее определенном списке, а после достижения конца списка указатель сбрасывается в начало. Этот подход отличается простотой реализации и минимальными вычислительными затратами ($O(1)$) на опера-

цию выбора сервера), однако он имеет существенное ограничение, которое заключается в полном игнорировании текущего состояния серверов, что может приводить к значительной неравномерности распределения нагрузки при колебаниях интенсивности запросов. Это также упоминается в работах [10-15].

Алгоритм Least Connections относится к динамическим методам балансировки и в отличие от Round Robin учитывает актуальную загрузку серверов, направляя каждый новый запрос на узел с минимальным количеством активных соединений. Такой подход демонстрирует значительно лучшее качество распределения нагрузки и особенно эффективен для систем с долгоживущими соединениями. Однако алгоритм Least Connections имеет высокие вычислительные затраты – ($O(N)$ на операцию выбора сервера) и требует постоянного мониторинга состояния всех узлов, что делает его менее предпочтительным для крупномасштабных систем с числом серверов, превышающим несколько сотен.

Алгоритм Weighted Round Robin представляет собой модификацию классического Round Robin, в которой учитываются различия в производительности серверов через систему весовых коэффициентов. Каждый сервер получает количество запросов, пропорциональное назначенному ему весу, что позволяет более эффективно использовать ресурсы. Время выбора сервера составляет $O(1)$ при простейшей реализации или $O(\log M)$ при использовании оптимизированных структур данных, где M – количество уникальных весов. Основным недостатком данного подхода является отсутствие реакции на динамические изменения нагрузки – весовые коэффициенты требуют ручной настройки и не адаптируются автоматически к изменению состояния серверов.

В ядре операционной системы Linux реализован планировщик задач Completely Fair Scheduler (CFS), основанный на использовании самобалансирующейся структуры данных – красно-черного дерева, и это обеспечивает справедливое распределение процессорного времени между конкурирующими потоками с логарифмической сложностью выбора следующей задачи [16].

Исходя из эффективности этого алгоритма, рассмотрим алгоритм красно-черных деревьев в качестве основы для алгоритмов балансировки нагрузки в распределённых системах. Такой подход мог бы обеспечить не только предсказуемую сложность операций вставки, выбора и удаления узлов, но и естественный механизм учёта веса или загруженности узлов, аналогично тому, как CFS учитывает накопленное время выполнения задач.

Красно-черные деревья – это класс сбалансированных бинарных деревьев поиска, где строгая система цветовой маркировки узлов и набор инвариантов обеспечивают логарифмическую временную сложность операций и способность поддерживать сбалансированность дерева, что обеспечивается за счёт соблюдения строгих правил:

- Корень дерева остается черным, что исключает вариативность в обработке корневого элемента, тем самым обеспечивая предсказуемость поведения всей структуры.
- NIL-узлы, являющиеся специальными листовыми элементами и представляющие собой отсутствие данных, также являются черными, что создает единую систему координат для измерения высоты дерева.
- Новые узлы при вставке всегда инициализируются красным цветом, что минимизирует нарушение инвариантов, при этом красные узлы не могут находиться друг за другом по вертикали. Это необходимо для того, чтобы избежать возникновения цепочек из красных узлов, что привело бы к деградации дерева до линейного списка. Для соблюдения этого правила может быть применена система коррекций, включающая в себя перекрашивание ближайших узлов или повороты поддеревьев.
- Во всех ветвях дерева сохраняется одинаковое количество черных узлов на пути от корня к листьям, что обеспечивает равномерное распределение элементов.

Использование красно-чёрных деревьев способствует поддержанию оптимального уровня загрузки серверов: за счёт оперативного перераспределения задач между исполнителями исключаются как простой вычислительных ресурсов, так и их перегрузка. Структура дерева адаптивно реагирует на изменения внешней нагрузки – поступление новых задач или завершение текущих – без ухудшения производительности, поскольку балансировка осуществляется локально и строго ограничена количеством необходимых операций. Подобные характеристики особенно значимы в условиях высоконагруженных распределённых систем, где стабильность, масштабируемость и предсказуемость поведения являются критическими требованиями к архитектурным решениям. Они также упомянуты в источниках [17-20].

Для того чтобы определить, какой из рассматриваемых алгоритмов наиболее эффективно решает проблему распределения нагрузки для обслуживания очередей задач в динамических кластерах с большим количеством серверов, необходимо провести теоретическое сравнение их характеристик. Такой анализ позволит выявить зависимость производительности алгоритмов от масштабов системы и особенностей распределения нагрузки. В качестве методологической основы исследования будут использованы подходы, описанные в источниках [21-22], что обеспечивает объективность и воспроизводимость полученных результатов. Сравнение алгоритмов будет проводиться по трём ключевым параметрам, определяющим их эффективность в распределённых вычислительных средах:

- Время выполнения операции, которое основывается на вычислительной сложности алгоритма (количестве элементарных операций) и тактовой частоте CPU. Для примера возьмём современный процессор Intel Core i7, который работает на частоте 3.0 ГГц. Время выполнения одной элементарной операции соответствует одному такту и определяется по формуле: длительность 1 такта (T) = 1/Тактовая частота (f), исходя из определения тактовой частоты, где Тактовая частота (f) = Количество тактов в секунду [Гц]. Таким образом, за время выполнения одной элементарной операции примем следующее значение: $T = 1 / (3.0 \times 10^9 \text{ Гц}) = 0.33 \times 10^{-9} \text{ сек} = 0.33 \text{ нс}$.

- Потребление памяти

- Адаптивность (приблизительное время, за которое система возвращается к равновесию после скачка нагрузки)

Сравнение будет проведено для трех характерных сценариев: малое количество серверов (N=10), средний кластер (N=100), крупная система (N=1000).

Алгоритм Round Robin поддерживает циклический указатель, последовательно направляя запросы на серверы. При достижении конца списка указатель сбрасывается. Вычислительная сложность: 1 операция (инкремент) + 1 операция (взятие модуля) = 2 – не зависят от N – O(1). Для системы с любым количеством серверов время выполнения операции будет одинаковым (табл. 1).

Таблица 1

Алгоритм Round Robin

N	Операции	Время (с учётом задержек)
10	2	1 нс
100	2	1 нс
1000	2	1 нс

Потребление памяти: фиксированное, 8 байт для 64-битного указателя текущей позиции (табл. 2).

Таблица 2

Потребление памяти при использовании Round Robin

N	Память
10	8 байт
100	8 байт
1000	8 байт

Адаптивность: отсутствует, так как алгоритм не учитывает текущую загрузку серверов и распределяет запросы строго по циклу. Независимо от того, перегружен ли какой-либо сервер, Round Robin продолжает отправлять на него новые запросы, поэтому система не способна самостоятельно реагировать на изменение нагрузки.

Алгоритм Least Connections требует поддержки актуального счетчика соединений для каждого сервера. Вычислительная сложность операции поиска счетчика с минимальным количеством подключений составляет N сравнений + (N-1) условных переходов = O(N) (табл. 3).

Таблица 3

Алгоритм Least Connections

N	Операции	Время (с учётом задержек)
10	10	10 нс
100	100	100 нс
1000	1000	1000 нс

Потребление памяти: $N \times$ (размер счётчика, 8 байт для 64-битного указателя текущей позиции) = $8N$ байт (табл. 4).

Таблица 4

Потребление памяти при использовании алгоритма Least Connections

N	Память
10	80 байт
100	800 байт
1000	8 Кбайт

Адаптивность алгоритма заключается в его способности перераспределять нагрузку, отдавая предпочтение серверам с меньшим количеством активных подключений. При увеличении нагрузки на один из серверов LC быстро обнаруживает перегрузку (обычно за 1-2 цикла балансировки) и перенаправляет новые запросы на менее загруженные узлы, достигая равномерного распределения за 3–5 циклов. Однако при кратковременных скачках нагрузки или в больших кластерах ($N > 1000$) эффективность LC снижается из-за инертности. Для высокодинамичных систем лучше подходят более чувствительные алгоритмы, например, взвешенные методы или алгоритмы, учитывающие время отклика. Эти системы также рассмотрены в источниках [23-34].

Алгоритм Weighted Round Robin проходит по списку серверов, пока не будет найден подходящий, в худшем случае вычислительная сложность операции составит $O(N)$ (табл. 5).

Таблица 5

Алгоритм Weighted Round Robin

N	Операции	Время (с учётом задержек)
10	10	10 нс
100	100	100 нс
1000	1000	1000 нс

Адаптивность алгоритма Weighted Round Robin ограничена статическими весами серверов: он эффективно распределяет нагрузку согласно заданным весам, но не реагирует на реальное состояние серверов (например, перегрузку или время отклика). При изменении весов требуется ручное обновление конфигурации и пересчёт внутренних структур (сложность $O(N)$), что делает алгоритм инертным к динамическим изменениям нагрузки.

Потребление памяти в алгоритме Weighted Round Robin заключается в хранении следующей структуры: веса серверов в формате int32 (например) – $4N$ байт, текущий 64-битный указатель – 8 байт (табл. 6).

Балансировка на основе алгоритма красно-чёрного дерева заключается в хранении и динамическом изменении серверов с ключом, отражающим метрику загрузки сервера, например, количество соединений, и поиске наименее загруженного из них. Вычислительная сложность поиска минимума в красно-чёрном дереве составляет $O(\log N)$ благодаря свойствам самобалансировки: дерево всегда поддерживает высоту, не превышающую $2\log_2(N+1)$, за счёт строгих правил окраски узлов и автоматической балансировки при операциях вставки и удаления.

Таблица 6

Потребление памяти при использовании алгоритма Weighted Round Robin

N	Вес	Счётчик	Общий объем памяти
10	40 байт	8 байт	48 байт
100	400 байт	8 байт	408 байт
1000	4 Кбайт	8 байт	4 Кбайт

Минимум находится в крайнем левом узле, доступ к которому требует последовательного прохождения не более $\log(N)$ уровней, что гарантирует логарифмическое время поиска даже в худшем случае. Информация об алгоритме представлена на таблицах 7-8.

Таблица 7

Алгоритм красно-черного дерева

N	Операции	Время (с учётом задержек)
10	$\log_2 10 \approx 4$	4 нс
100	$\log_2 100 \approx 7$	7 нс
1000	$\log_2 1000 \approx 10$	10 нс

Таблица 8

Потребление памяти при алгоритме красно-черного дерева

N	Память
10	410 байт
100	4.1 КБ
1000	41 КБ

На основании полученных данных составим полную таблицу сравнения всех алгоритмов (табл. 9).

Заключение

Среди рассмотренных алгоритмов балансировки нагрузки красно-черное дерево демонстрирует наиболее сбалансированные характеристики по скорости работы, адаптивности и масштабируемости, особенно в высоконагруженных системах. В отличие от Round Robin ($O(1)$), который не учитывает состояние серверов, и Least Connections ($O(N)$), чья производительность снижается в крупных кластерах, красно-черное дерево обеспечивает стабильную скорость работы ($O(\log N)$) при любом количестве серверов.

Красно-чёрное дерево обеспечивает эффективный поиск наименее загруженного сервера и скорость работы – минимум находится в крайнем левом узле (фактически $O(1)$ при кэшировании), а обновление счётчиков требует $O(\log N)$ операций благодаря автоматической балансировке [27].

Таблица 9

Сравнительная таблица алгоритмов

Параметр	Round Robin	Least Connections	Weighted Round Robin	Красно-чёрное дерево
Вычислительная сложность	$O(1)$	$O(N)$	$O(N)$	$O(\log N)$
Время операции (N=10)	1 нс	10 нс	10 нс	4 нс
Время операции (N=100)	1 нс	100 нс	100 нс	7 нс
Время операции (N=1000)	1 нс	1000 нс	1000 нс	10 нс
Память (N=10)	8 байт	80 байт	48 байт	410 байт
Память (N=100)	8 байт	800 байт	408 байт	4.1 КБ
Память (N=1000)	8 байт	8 КБ	4 КБ	41 КБ
Адаптивность	Нет	Средняя	Низкая	Высокая

Как видно на графике «Время выполнения операций» (рис. 1), время работы красно-черного дерева остаётся низким даже при увеличении количества серверов до 1000, уступая по скорости лишь Round Robin, но значительно опережая Least Connections и Weighted Round Robin (рис. 1).

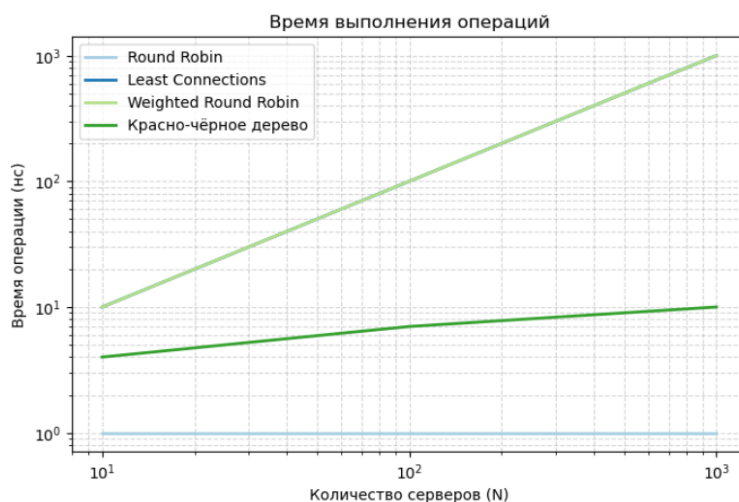


Рис. 1. График зависимости времени выполнения операций от количества серверов

Красно-чёрное дерево быстро перестраивается при изменении нагрузки, перераспределяя соединения между серверами за логарифмическое время.

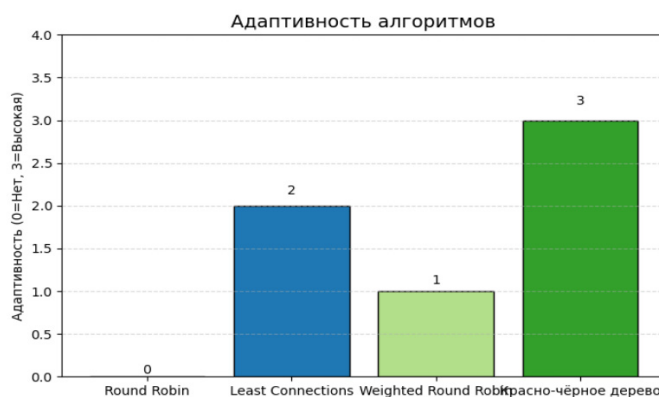


Рис. 2. Сравнение адаптивности алгоритмов

Несмотря на то, что красно-черное дерево использует больше памяти, чем Round Robin или Weighted Round Robin, это потребление остаётся умеренным и оправданным с точки зрения высокой адаптивности и скорости работы. На графике «Потребление памяти» (рис. 3) показано, что объем используемой памяти красно-черного дерева растёт с увеличением количества серверов, но остаётся приемлемым для практического применения в больших кластерах.

Таким образом, по совокупности всех ключевых показателей – время работы, память и адаптивность – красно-чёрное дерево демонстрирует наилучший баланс, особенно в условиях высоких нагрузок и больших распределённых систем. Этот алгоритм обеспечивает оптимальное сочетание скорости, гибкости и масштабируемости, что делает его предпочтительным выбором для динамических кластеров с большим числом серверов.

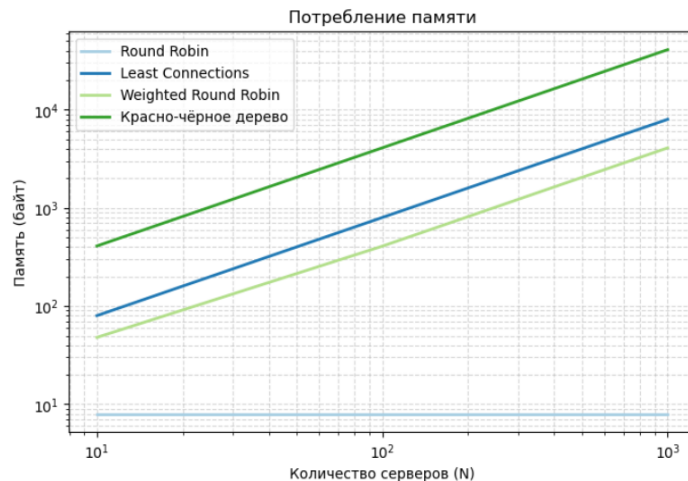


Рис. 3. График зависимости потребляемой памяти от количества серверов

Литература

1. Tremasova L. A., Korovushkina V. M., Panteleeva K. A., Gadasin D. V. Distributed Information System Software Code Reliability Evaluation // 2024 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF), St. Petersburg, Russian Federation, 2024, pp. 1-5, doi: 10.1109/WECONF61770.2024.10564614.
2. Марченко Д.О., Клыгина О.Г., Гадасин Д.В., Шведов А.В. Обеспечение механизмов балансировки нагрузки в сетях с сегментной маршрутизацией на основе данных мониторинга // В сборнике: Перспективные технологии в средствах передачи информации. Материалы 14-ой международной научно-технической конференции. Владимир, 2021. С. 419-422.
3. Кочетов Ю.А., Кочетова Н.А. Задача балансировки нагрузки на серверы // Вестник НГУ. Серия: Информационные технологии. 2013. №4. URL: <https://cyberleninka.ru/article/n/zadacha-balansirovki-nagruzki-na-servery> (дата обращения: 05.09.2025).
4. Кукарцев А. М. Применение спаренных красно-черных деревьев для снижения пространственных характеристик алгоритмов частотного анализа информационных сообщений экспоненциального размера // Решетневские чтения. 2014. №18. URL: <https://cyberleninka.ru/article/n/primenenie-sparenykh-krasno-chernyh-dereviev-dlya-snizheniya-prostranstvennykh-harakteristik-algoritmov-chastotnogo-analiza> (дата обращения: 06.09.2025).
5. Бадасян Т.С., Авагян С.К. Красно-чёрное дерево: балансирование и сложность // Наука, техника и образование. 2020. №3 (67). URL: <https://cyberleninka.ru/article/n/krasno-chyornoe-derevo-balansirovanie-i-slozhnost> (дата обращения: 07.09.2025).
6. Гадасин Д.В., Багдасарян А.С., Тремасова Л.А., Яблокова С.А. Выполнение лабораторной работы по курсу «Мультимедийные информационные системы» с использованием программного комплекса // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2024. Т. 13, № 1. С. 18-30. EDN HGKCOY.
7. Гадасин, Д.В., Шустов С.А. Исследование эффективности протоколов маршрутизации в условиях сетей с высокой нагрузкой // Теория и практика экономики и предпринимательства : Труды XXI Международной научно-практической конференции, Симферополь – Гурзуф, 18-20 апреля 2024 года. Симферополь: ИП Зуева Т.В., 2024. С. 236-237. EDN WNVEOC.
8. Гадасин Д.В., Шведов А.В. Применение транспортной задачи для балансировки нагрузки в условиях нечеткости исходных данных // Т-Comm: Телекоммуникации и транспорт. 2024. Т. 18, № 1. С. 13-20. DOI 10.36724/2072-8735-2024-18-1-13-20. EDN WKNPIX.
9. Гадасин Д.В., Бессолицын А.Д. Виды и методы структурирования данных из различных информационных систем: анализ и применение // Актуальные проблемы и перспективы развития экономики, Симферополь - Гурзуф, 12-14 октября 2023 года. Симферополь: ИП Зуева Т.В., 2023. С. 202-204. EDN UGZRXL.
10. Гадасин Д.В., Назаренко С.С., Тремасова Л.А. Особенности проведения практических занятий по дисциплине «Принципы построения систем управления базами данных и знания» // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2023. Т. 12, № 1. С. 21-31. EDN FGSGBK.
11. Шульгина П.Д., Гадасин Д.В., Тремасова Л.А. Взвешивание признаков как предварительная обработка исходных наборов данных // Системы синхронизации, формирования и обработки сигналов. 2024. Т. 15, № 3. С. 40-47. EDN BLOWRB.

12. *Гадасин Д.В., Пак Е.В., Коровушкина В.М., Мелькова Е.К.* Предобработка текстовой информации на основе термов естественного языка // REDS: Телекоммуникационные устройства и системы. 2022. Т. 12, № 1. С. 4-11. EDN PDGAVP.
13. *Гадасин Д.В., Шведов А.В., Егорова Ю.Д., Шайдулина И.П.* Применение метода мажоритарного кодирования для определения оптимального маршрута передачи данных в сети // DSPA: Вопросы применения цифровой обработки сигналов. 2023. Т. 13, № 1. С. 20-30. EDN IECPBA.
14. *Tremasova L.A., Andriyanova A.K., Gadasin D.V., Gadasin D.D.* Modeling and Solving the Problem of Load Balancing in Data Transmission Networks Using the Stepping Stone Method // 2024 Systems of Signals Generating and Processing in the Field of on Board Communications, Moscow, Russian Federation, 2024, pp. 1-7, doi: 10.1109/IEEECONF60226.2024.10496718.
15. *Gadasin D.V., Shvedov A.V., Klygina O.G.* Organization of Interaction Between the Concept of Fog Computing and Segment Routing for the Provision of IoT Services in Smart Grid Networks // Wave Electronics and Its Application in Information and Telecommunication Systems. 2022. Vol. 5, No. 1, pp. 141-146. EDN UQSHRH.
16. *Золотарева П.Ю., Гадасин Д.В., Маклачков К.А.* Методы обработки информации в распределенных информационных системах // Тенденции развития Интернет и цифровой экономики : Труды VI Международной научно-практической конференции, Симферополь-Алушта, 01-03 июня 2023 года. Симферополь: ИП Зуева, 2023. С. 187-189. EDN LGONZK
17. *Gadasin D.V., Shvedov A.V., Yudin A.A.* Clustering methods in large-scale systems // Synchroninfo Journal. 2020. Vol. 6, No. 5, pp. 21-24. DOI 10.36724/2664-066x-2020-6-5-21-24. EDN XHNSYV
18. *Gadasin D.V., Shvedov A.V., Kuzin I.A.* A model for representing the color and depth metric characteristics of objects in an image // 2021 Systems of Signal Synchronization, Generating and Processing in Telecommunications, SYNCHROINFO 2021 – Conference Proceedings, Svetlogorsk, Kaliningrad Region, 30 июня – 02 июля 2021 года. Svetlogorsk, Kaliningrad Region, 2021. P. 9488349. – DOI 10.1109/SYNCHROINFO51390.2021.9488349. EDN YAYZVP.
19. *Zolotukhin P.A., Melkova E.K., Gadasin D.V., Korovushkina V.M.* Using Intelligent Testing as a Tool to Improve the Quality of Transport Information Systems // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 – Conference Proceedings, Moscow, 15-17 марта 2022 года. Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744348. EDN NOMJLX.
20. *Гадасин Д.В., Шведов А.В., Алексеева Е.А.* Информационная энтропия в стохастических сетях связи // Телекоммуникационные и вычислительные системы 2020 : Труды международной научно-технической конференции, Москва, 14-17 декабря 2020 года / Московский технический университет связи и информатики. М.: Горячая линия – Телеком, 2020. С. 108-116. EDN IOGLQH
21. *Гадасин Д.В.* Построение бинарного дерева минимальной цены // T-Comm: Телекоммуникации и транспорт. 2024. Т. 18, № 11. С. 38-44. DOI 10.36724/2072-8735-2024-18-11-38-44. EDN GMCEWG
22. *Гадасин Д.В., Шведов А.В.* Проблемы интеграции концепции "Интернет вещей" и облачных вычислений // Технологии информационного общества : Материалы XIII Международной отраслевой научно-технической конференции, Москва, 20-21 марта 2019 года. Том 2. М.: Издательский дом Медиа Паблишер, 2019. С. 22-23. EDN MEQRFA
23. *Gadasin D.V., Shvedov A.V., Vakurin I.S.* Determination of Semantic Proximity of Natural Language Terms for Subsequent Neural Network Training // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 – Conference Proceedings, Moscow, 15-17 марта 2022 года. Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744290. EDN LASMDY
24. *Гадасин Д.В., Шведов А.В., Мелькова Е.К.* Структурирование данных исходя из центра масс // Актуальные проблемы и перспективы развития экономики : Труды XXI Международной научно-практической конференции, Симферополь-Гурзуф, 20-22 октября 2022 года. Симферополь: Крымский федеральный университет им. В.И. Вернадского, 2022. С. 266-268. EDN RFCCST.
25. *Shvedov A.V., Gadasin D.V., Pak E.V.* Application of the Backman Model for the Distribution of Traffic Flows in Networks with Segment Routing // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 – Conference Proceedings, Moscow, 15-17 марта 2022 года. Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744344. EDN RBMTBQ
26. *Gadasin D.V., Koltsova A.V., Gadasin D.D.* Algorithm for Building a Cluster for Implementing the 'Memory as a Service' Service in the IoT Concept // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, Conference Proceedings, Moscow, 16-18 марта 2021 года. Moscow, 2021. P. 9416112. DOI 10.1109/IEEECONF51389.2021.9416112. EDN VRPCFG
27. *Shevelev S.V., Shvedov A.V., Gadasin D.V., Vakurin I.S.* Syntax and probability vectors in search query // Wave Electronics and Its Application in Information and Telecommunication Systems. 2023. Vol. 6, No. 1, pp. 407-414. EDN TVFKOH
28. *Гадасин Д.В., Шведов А.В., Кузин И.А.* Трехмерная реконструкция объекта по одному изображению с использованием глубоких сверточных нейронных сетей // T-Comm: Телекоммуникации и транспорт. 2022. Т. 16, № 7. С. 29-35. DOI 10.36724/2072-8735-2022-16-7-29-35. EDN YTLCNW.

29. *Shvedov A.V., Gadasin D.V., Alyoshintsev A.V.* Segment routing in data transmission networks // Т-Comm: Телекоммуникации и транспорт. 2022. Vol. 16, No. 5, pp. 56-62. DOI 10.36724/2072-8735-2022-16-5-56-62. EDN VAYLJQ.

30. *Alyoshintsev A.V., Gadasin D.V., Vakurin D.S., Chelyshkov P.D.* Methods for evaluating the noise immunity of modems // Т-Comm: Телекоммуникации и транспорт. 2025. Vol. 19, No. 9, pp. 50-58. DOI 10.36724/2072-8735-2025-19-9-50-58. EDN TGKCQD.

31. *Гадасин Д. В.* Способ определения основных узлов сети для анализа ее состояния // Т-Comm: Телекоммуникации и транспорт. 2025. Т. 19, № 12. С. 16-24. DOI 10.36724/2072-8735-2025-19-12-16-24. EDN FGAATI.

32. *Мелькова Е.К., Шведов А.В., Трemasова Л.А., Гадасин Д.В.* Организация кластера исходя из функции принадлежности // Системы синхронизации, формирования и обработки сигналов. 2023. Т. 14, № 1. С. 30-39. EDN CNVIJU.

33. *Яковенко Н.В., Гадасин Д.В., Коцич Л.* Повышение точности коэффициента влияния ошибок в информационных системах с применением метода обратного распространения ошибки // Системы синхронизации, формирования и обработки сигналов. 2024. Т. 15, № 4. С. 35-42. EDN CMFVNH.

34. *Шульпина П.Д., Гадасин Д.В., Трemasова Л.А.* Взвешивание признаков как предварительная обработка исходных наборов данных // Системы синхронизации, формирования и обработки сигналов. 2024. Т. 15, № 3. С. 40-47. EDN BLOWRB.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПРОТОКОЛОВ ПЕРЕДАЧИ ДАННЫХ ДЛЯ СИСТЕМ УМНОГО ДОМА НА БАЗЕ МИКРОКОНТРОЛЛЕРА ESP-32

Тимошенко Павел Алексеевич

Магистрант МТУСИ, Москва, Россия,

Pasha02ti@yandex.ru

Косичкина Татьяна Павловна

заведующий кафедрой СиСРТ МТУСИ,

кандидат технических наук, доцент, Москва, Россия,

t.p.kosichkina@mtuci.ru

Аннотация

В статье проведен комплексный сравнительный анализ проводных и беспроводных протоколов передачи данных для систем умного дома на базе микроконтроллера ESP-32. Рассмотрены варианты реализации и практического применения интерфейсов UART, I2C, Bluetooth и Wi-Fi для использования в системах умного дома. Предметом исследования являются технические характеристики, энергоэффективность и надежность различных протоколов связи. В результате проведенного анализа определены оптимальные области применения каждого интерфейса, выявлены их преимущества и ограничения, сформулированы практические рекомендации по выбору протокола передачи данных в зависимости от конкретных требований проекта. Результаты, полученные при исследовании, могут использоваться в проектировании систем домашней автоматизации, мониторинга и управления, а также в образовательном процессе при изучении технологий Интернета вещей.

Ключевые слова:

ESP-32, умный дом, передача данных, UART, I2C, Bluetooth, Wi-Fi, Интернет вещей.

Введение

Современный этап развития повседневных технологий характеризуется активным внедрением систем умного дома и Интернета вещей (IoT) в нашу жизнь. Динамичное развитие данных систем обусловлено повышением доступности микроконтроллерных платформ, среди которых особое место занимает микроконтроллер ESP-32 благодаря встроенным модулям Wi-Fi и Bluetooth [1].

Актуальность исследования определяется необходимостью решения важной проблемы, которая возникает при проектировании систем домашней автоматизации - выбор оптимального варианта интерфейса передачи данных между устройствами. Многообразие доступных интерфейсов, включая проводные (UART, I2C) и беспроводные (Bluetooth LE, Wi-Fi) технологии, создает сложности для разработчиков, требующие системного подхода к оценке их эффективности в конкретных условиях эксплуатации.

Анализ литературных источников показывает, что существующие исследования в основном фокусируются на отдельных аспектах работы протоколов передачи данных, однако комплексный сравнительный анализ, учитывающий все факторы, влияющие на работу систем умного дома, представлен недостаточно полно [4, 5]. В частности, недостаточно изучены вопросы энергоэффективности различных интерфейсов при длительной автономной работе, что является критически важным для устройств, работающих от автономных источников питания.

Целью данной работы является проведение сравнительного анализа протоколов передачи данных для систем умного дома на базе микроконтроллера ESP-32 и разработка практических рекомендаций по их выбору в зависимости от требований конкретного применения. В данной работе решаются определенные задачи, перечисленные ниже.

1. Исследование технических характеристик и особенностей реализации интерфейсов UART, I2C, Bluetooth и Wi-Fi на платформе ESP-32.
2. Экспериментальная оценка энергопотребления, скорости передачи данных и надежности соединения для каждого протокола.
3. Анализ стабильности работы в домашних условиях.

4. Разработка критериев выбора и формулирование практических рекомендаций по применению протоколов передачи данных.

5. Практическая значимость работы выражается в возможности разработчикам систем умного дома обоснованно выбирать протоколы передачи данных, оптимизируя проектные решения по критериям энергоэффективности, стоимости и надежности.

1. Анализ интерфейсов передачи данных ESP-32 по кабелю

В основном для передачи данных с Arduino используются 2 стандарта: UART и I2C. Для начала разберемся в особенностях данных интерфейсов и используемых протоколов [2].

1.1. Интерфейс UART

Универсальный асинхронный приемо-передатчик (UART) представляет собой один из старейших и наиболее распространенных способов последовательной связи, сохраняющий свою актуальность несмотря на появление более современных интерфейсов. Его универсальность и простота реализации обеспечили широкое применение в различных электронных устройствах [6].

Ключевой особенностью UART является асинхронная передача данных, не требующая тактового сигнала. Каждое устройство, оснащенное этим интерфейсом, имеет два основных контакта:

TX (Transmit) – выход передатчика

RX (Receive) – вход приемника

Правильное соединение устройств предполагает перекрестное подключение (рис. 1): TX первого устройства к RX второго, и наоборот. Ошибочное соединение одноименных контактов (TX-TX или RX-RX) может привести к некорректной работе системы и потенциальному повреждению оборудования.

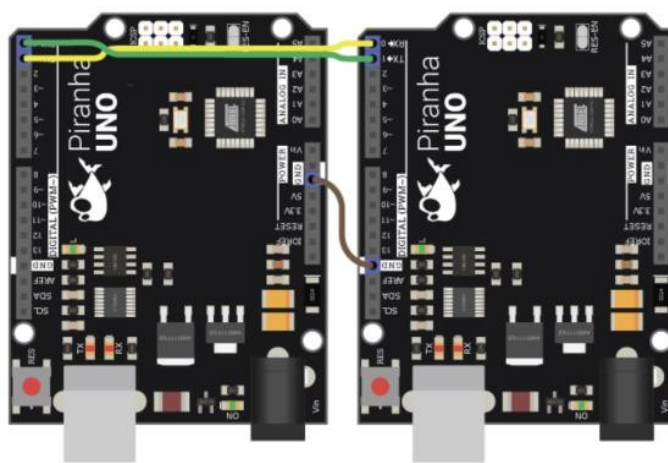


Рис. 1. Подключение двух плат Arduino через интерфейс UART

Чтобы понять, что роднит и отличает разные UART-интерфейсы, разберем принцип работы самого популярного протокола RS-232. Физическая реализация протокола RS-232 имеет существенные отличия от стандартной TTL-логики. В то время как в TTL-уровнях логические состояния определяются напряжениями 0В и 5В, в RS-232 используется значительно более широкая амплитуда сигналов. Логический "0" представлен положительным напряжением от +3В до +12В, тогда как логическая "1" соответствует отрицательному напряжению от -3В до -12В, соответственно. Промежуток от -3 до 3 вольт считается зоной неопределенности.

Общий рабочий диапазон составляет 24В, что обеспечивает повышенную помехозащищенность и позволяет использовать кабели длиной до 15 м.

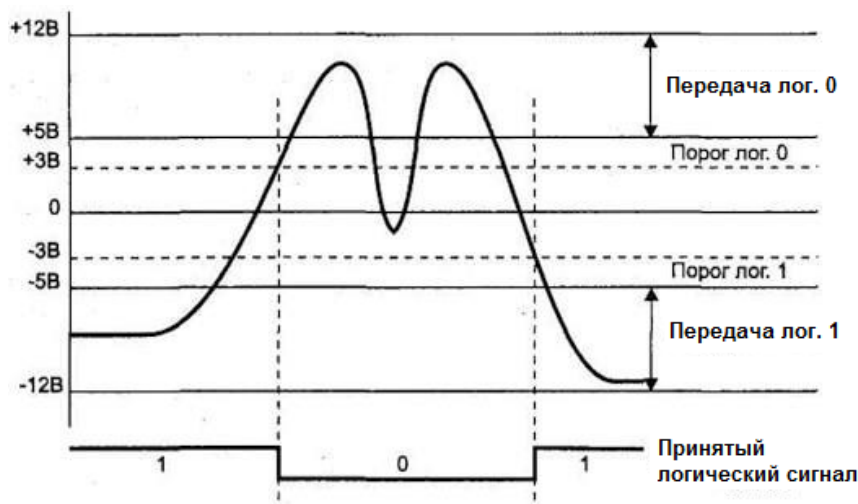


Рис. 2. График передачи логических данных в UART

Итак, ключевой особенностью интерфейса RS-232 является использование сигналов с увеличенной амплитудой – до 24 вольт между крайними значениями. Такое техническое решение в первую очередь направлено на обеспечение устойчивости передачи данных в условиях электромагнитных помех. Благодаря значительному перепаду напряжений, приемное оборудование может уверенно распознавать логические состояния даже при наличии существенных наводок в линии связи.

Параметры линии передачи Стандартом предусматривается использование кабелей длиной до 15 м, что делает интерфейс пригодным для организации связи между устройствами в пределах одного помещения или технологической установки. Электрические характеристики сигналов являются определяющим фактором, отличающим RS-232 от других протоколов семейства UART.

Формат кадра передачи данных Базовый формат данных включает 10 битов, однако эта конфигурация может варьироваться в зависимости от установок COM-порта.

В режиме ожидания линия находится под напряжением -12В (логическая "1"). Это состояние поддерживается постоянно при отсутствии передачи.

При передаче сигнала структура передаваемого кода выглядит следующим образом (рис. 3):

- 1) стартовый бит – всегда нулевой, сигнализирует о начале передачи;
- 2) биты данных – 8 информационных битов, передаются последовательно;
- 3) бит четности – служит для контроля ошибок передачи;
- 4) стоповый бит – указывает на завершение кадра (может занимать 1, 1.5 или 2 битовых интервала).

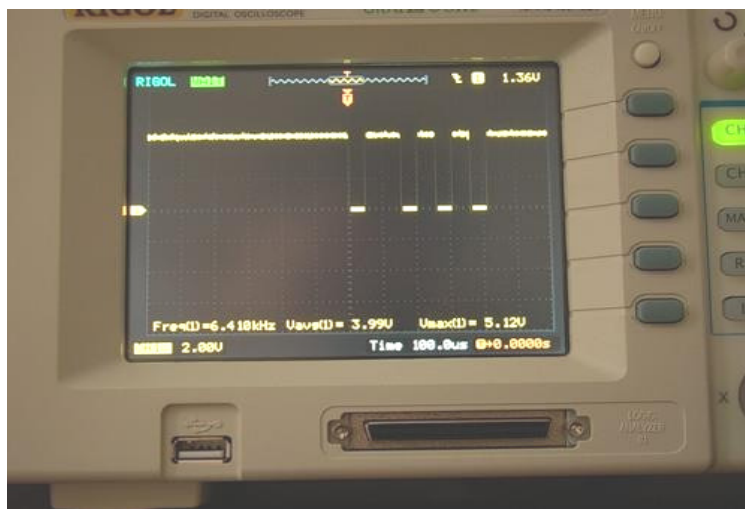


Рис. 3. Сигнал UART на экране осциллографа. Виден старт бит, данные и стоповый бит

Вариабельность количества стоп-битов и возможность отключения бита четности обеспечивают гибкость настройки протокола под конкретные задачи, сохраняя при этом базовые принципы работы, общие для всего семейства UART-интерфейсов.

Скорость передачи измеряется в бодах и определяет время передачи одного бита. Например, при скорости 9600 бод длительность бита составляет $1/9600 \approx 104$ микросекунды. Наиболее распространенными скоростями являются 9600, 19200 и 115200 бит.

1.2. Интерфейс I2C

Интерфейс I2C (Inter-Integrated Circuit) представляет собой синхронный последовательный протокол, разработанный компанией Philips [3]. Ключевой особенностью данного протокола является использование всего двух линий для организации связи между несколькими устройствами:

- SCL (Serial Clock) – тактовый сигнал, генерируемый ведущим устройством
- SDA (Serial Data) – линия для передачи данных

Протокол поддерживает адресную систему, позволяя одному ведущему устройству взаимодействовать с несколькими ведомыми. Каждое ведомое устройство имеет уникальный адрес, что исключает конфликты при обращении к конкретному компоненту системы (рис. 4).

В любой момент времени только ведущий может начать процесс обмена данными. Поскольку в этом протоколе допускается несколько ведомых, то ведущий должен обращаться к ним, используя различные адреса. То есть только ведомый с заданным адресом должен отвечать на сигнал ведущего, а все остальные ведомые в это время не должны быть активны. Таким образом, мы можем использовать одну и ту же линию для обмена данными с несколькими устройствами.

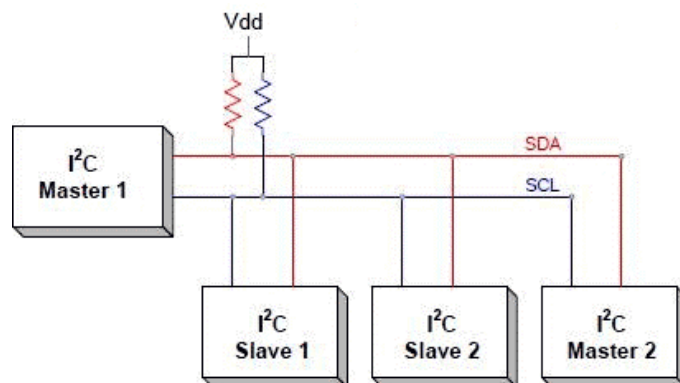


Рис 4. Схема связи с помощью протокола I2C [3]

Уровни напряжений для передаваемых сигналов в интерфейсе I2C жестко не определены. В этом плане I2C является достаточно универсальным, то есть если устройство питается от напряжения 5 В, оно для связи с помощью протокола I2C может использовать уровень 5 В, а если устройство питается от напряжения 3.3 В, то оно для связи с помощью протокола I2C может использовать уровень 3 В.

Существует несколько условий для осуществления передачи данных в протоколе I2C. Инициализация передачи начинается с падения уровня на линии SDA, которое определяется как условие для начала передачи ('START' condition) на представленной ниже осциллограмме (рис. 5). Как видно из этого рисунка, в то время как на линии SDA происходит падение уровня, в это же самое время на линии SCL поддерживается напряжение высокого уровня.

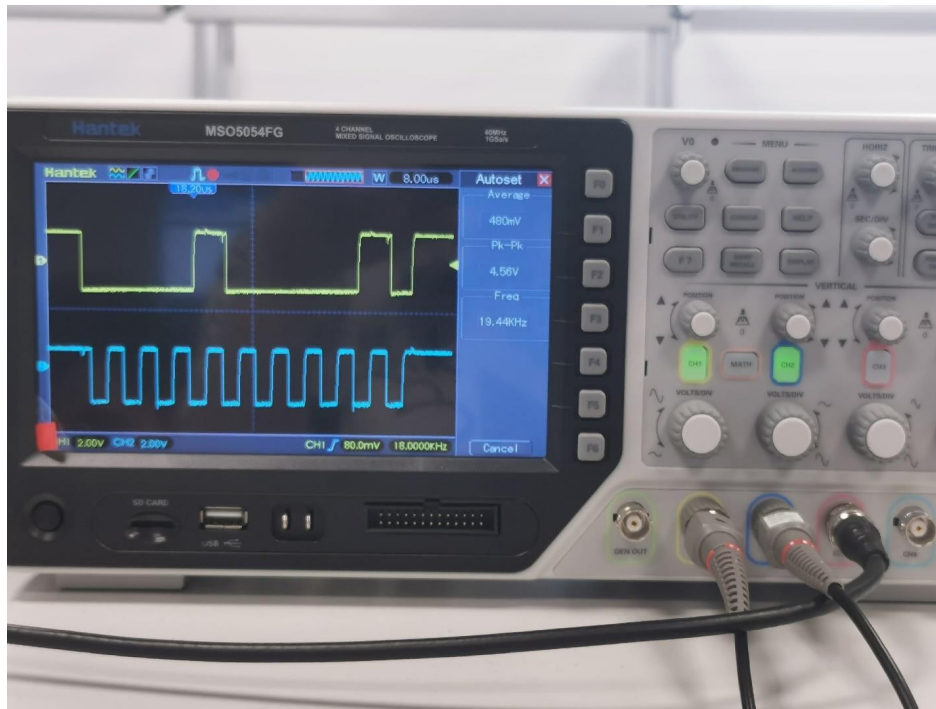


Рис. 5. Осциллограммы выходов SDA(желтый) и SCL(Синий)

Для использования в среде Arduino для I2C применяются специальные библиотеки, облегчающие работу.

2. Беспроводная передача данных для микроконтроллера ESP-32

2.1. Передача через Bluetooth

В плате, на основе микроконтроллера ESP-32, имеется не только обычный Bluetooth модуль, но и поддерживается технология BLE [1,3].

Технология BLE представляет собой энергоэффективную версию классического стандарта Bluetooth, ориентированную на приложения с ограниченным энергопотреблением. Основное ее отличие заключается в стратегии передачи данных - вместо постоянного поддержания соединения, BLE-устройства передают информацию короткими пакетами по требованию или с предопределенными интервалами.

Хотя BLE уступает классическому Bluetooth в пропускной способности и не предназначен для передачи объемных данных (аудио, видео), его энергоэффективность делает его идеальным решением для следующих применений:

- передача данных с датчиков и сенсоров;
- носимые портативные устройства;
- системы мониторинга;
- устройства умного дома [7].

Совместимость с Bluetooth 4.0 и более поздними версиями обеспечивает широкие возможности интеграции с современными мобильными устройствами и вычислительными платформами.

2.2. Передача через WiFi

Для работы с микроконтроллером EPS-32 через Wi-Fi (стандарт IEEE 802.11 b/g/n) имеется множество протоколов. При этом возможно несколько режимов использования беспроводного подключения, которые перечислены ниже.

1. Режим станции (STA) при котором ESP-32 подключается к существующей беспроводной сети (например, к домашнему роутеру), как ноутбук или телефон.

2. Режим точки доступа (AP), при котором ESP-32 создает свою собственную сеть Wi-Fi, к которой могут подключиться другие устройства (например, телефон, ноутбук). При этом доступа в интернет у сети может и не быть.

3. Смешанный режим (STA+AP) – это одновременная работа в двух режимах: подключение к роутеру и создание своей сети для управления.

4. Wi-Fi Mesh – возможность объединять несколько ESP-32 в самоорганизующуюся и самовосстанавливающуюся сеть с большим покрытием.

Возможным недостатком использования Wi-Fi является многообразие протоколов для различных версий стандарта. При этом зачастую приходится точно определять используемую версию стандарта, так как они не всегда являются совместимыми [8, 9].

3. Сравнение протоколов передачи

После того, как рассмотрены протоколы передачи для микроконтроллера ESP-32, необходимо сделать выводы и рекомендации по их применению для тех или иных приложений.

Для удобства сравнения все проанализированные данные были сведены в таблицу (табл. 1). По результатам анализа можно сделать следующие выводы: проводные решения оптимальны для локального обмена данными между компонентами системы, тогда как беспроводные технологии обеспечивают необходимую мобильность и возможность интеграции в облачные сервисы [10].

Для дальнейших исследований необходимо провести моделирование работы протоколов, которое заключается в настройке оборудования и написании программ.

Таблица 1

Сравнение протоколов [1-3]

Параметр	UART	I2C	Bluetooth LE	Wi-Fi
Скорость (макс.)	115кбит	3.4Мбит	1Мбит	150Мбит
Дальность	15м	1м	100м	100м
Энергопотребление	Низкое	Низкое	Очень низкое	Высокое
Сложность подключения	Простое	Среднее	Сложное	Сложное

4. Моделирование работы устройств

4.1. Пример программы, использующий Bluetooth с ESP-32.

Одним из самых простых способов для передачи данных на смартфон через Bluetooth являются разнообразные готовые программы, которые можно скачать в Google Play. В данном случае была использована программа Arduino Bluetooth Controller, меню которой приведено на рис. 6. С помощью меню была произведена настройка терминала (рис. 7).

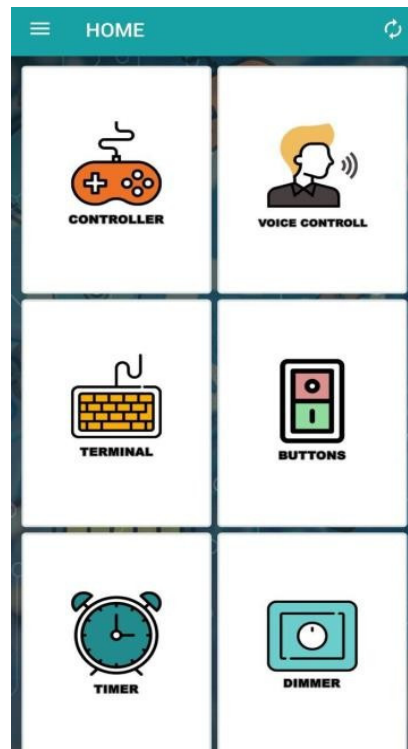


Рис. 6. Главное меню приложения

Для написания кода программы были использованы библиотеки для Bluetooth и датчиков температуры и влажности. Код для программы представлен ниже:

Алгоритм передачи данных датчика DHT11 по Bluetooth приведен ниже.

АЛГОРИТМ передачи данных датчика по Bluetooth

ПРОЦЕДУРА инициализация():

Настроить последовательный порт на 115200 бод

Инициализировать Bluetooth с именем "ESP32_Bluetooth"

Инициализировать датчик DHT11

КОНЕЦ ПРОЦЕДУРЫ

ПРОЦЕДУРА основной_цикл():

temperature = считать_температуру()

humidity = считать_влажность()

ЕСЛИ температура ИЛИ влажность не числа ТО

отправить_по_bluetooth("Ошибка чтения датчика!")

ВЕРНУТЬСЯ

КОНЕЦ ЕСЛИ

отправить_по_bluetooth("Температура: " + температура + " °C")

отправить_по_bluetooth("Влажность: " + влажность + " %")

Задержка(2000 мс)

КОНЕЦ ПРОЦЕДУРЫ

Для получения данных из ESP, надо подсоединиться к сети Bluetooth. Позже, мы заходим в терминал и получаем данные (рис. 7).

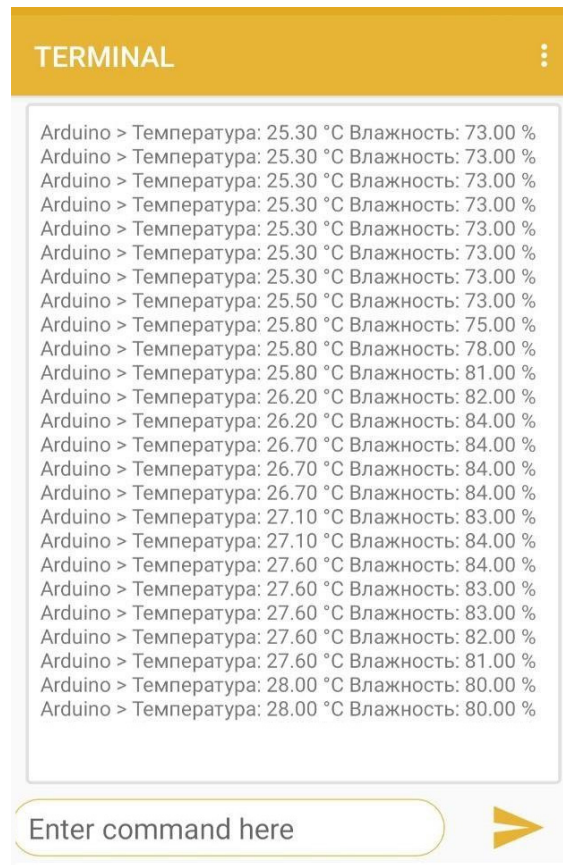


Рис. 7. Данные с терминала на телефоне

4.2. Передача по Wi-Fi через ESP-32

Для удобства использования Wi-fi можно использовать несколько вариантов вывода информации: сайт, приложение на телефоне и telegram бот.

Для примера подобного сайта был использован Arduino Cloud IoT.

Зарегистрировавшись на сайте и скачав Arduino Create Agent, мы можем легко создать специальную страницу, на которой мы видим данные с платы (рис. 8).

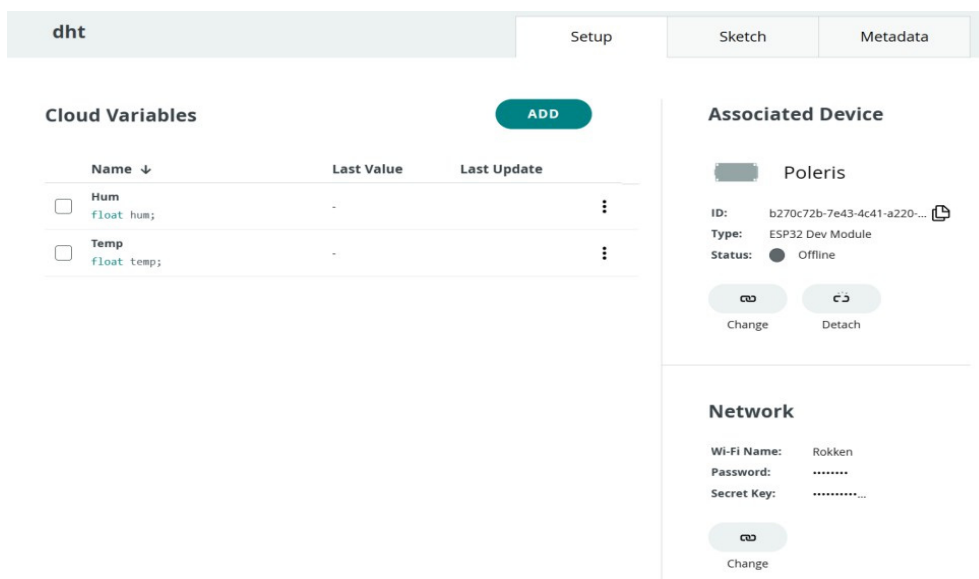


Рис. 8. Главная страница Arduino Cloud IoT

На главной странице мы можем задать переменные и подсоединиться к плате и Интернету. Код для программы заполняется в Sketch. Остается его только дополнить. Ниже приведён псевдокод программы для алгоритма облачного мониторинга влажности и температуры.

АЛГОРИТМ облачного мониторинга через Arduino IoT

ПРОЦЕДУРА настройка():

Инициализировать последовательный порт

Задержка 1500 мс

Инициализировать свойства облачного объекта

Подключиться к Arduino IoT Cloud

Установить уровень отладки

Вывести отладочную информацию

КОНЕЦ ПРОЦЕДУРЫ

ПРОЦЕДУРА основной_цикл():

Обновить состояние облачного соединения

humidity = считать_влажность_с_датчика()

temperature = считать_температуру_с_датчика()

// Облако автоматически синхронизирует переменные

КОНЕЦ ПРОЦЕДУРЫ

ПРОЦЕДУРА при_изменении_температуры():

// Обработчик изменения температуры (пустой)

КОНЕЦ ПРОЦЕДУРЫ

ПРОЦЕДУРА при_изменении_влажности():

// Обработчик изменения влажности (пустой)

КОНЕЦ ПРОЦЕДУРЫ

Запустив программу, мы получаем данные с платы. Пример окна получения данных приведен на рис. 9.



Рис. 9. Данные с датчика

4.3. Использование Телеграм-бота

Еще одним универсальным средством для управления и вывода информации из среды Ардуино является специализированный Телеграм-бот.

Подобного бота очень легко создать с помощью встроенного в сам Телеграм botfather. После придумывания названия бота и его короткой ссылки, мы получаем его уникальный код, с помощью которого мы и можем редактировать бота, добавляя новые команды.

Для наглядного примера использования подобного бота, была подключена плата ESP-32 с датчиком температуры и влажности DHT-11. Для корректной работы было подключено две библиотеки – для работы бота и для работы датчика. Сам псевдокод программы алгоритма Телеграм-бота для мониторинга данных представлен ниже.

АЛГОРИТМ Telegram бота для мониторинга данных

ПРОЦЕДУРА инициализация():

Настроить последовательный порт

Инициализировать датчик DHT11

Подключиться к Wi-Fi сети

Установить токен Telegram бота
КОНЕЦ ПРОЦЕДУРЫ

ПРОЦЕДУРА основной_цикл():
 humidity = считать_влажность()
 temp_celsius = считать_температуру_цельсий()
 temp_fahrenheit = считать_температуру_фаренгейт()

ЕСЛИ получено_новое_сообщение_в_telegram() ТО
 ЕСЛИ текст_сообщения = "Влажность" ТО
 отправить_ответ("Влажность: " + humidity + " %")
 ИНАЧЕ ЕСЛИ текст_сообщения = "Температура" ТО
 отправить_ответ("Температура: " + temp_celsius + "°C, " + temp_fahrenheit + "°F")
 КОНЕЦ ЕСЛИ
КОНЕЦ ЕСЛИ

Задержка(10 мс)
КОНЕЦ ПРОЦЕДУРЫ

Если код работает корректно, то в Телеграм-боте, при написании команд «Температура» и «Влажность» нам в мессенджер, отправляются данные температуры или влажности воздуха, соответственно, на текущий момент.



Рис. 10. Данные, отображаемые в Телеграм-боте

Заключение

Проведенное исследование позволило проанализировать возможности микроконтроллера ESP32 в контексте построения систем автоматизации умного дома. В ходе работы была подтверждена высокая эффективность микроконтроллера для решения широкого круга задач, связанных с сбором, обработкой и передачей данных в распределенных системах.

Основные научные и практические результаты работы заключаются в следующем:

1. Комплексный анализ протоколов передачи данных выявил четкие области применения каждого интерфейса: проводные решения оптимальны для локального обмена данными между компонентами системы, тогда как беспроводные технологии обеспечивают необходимую мобильность и возможность интеграции в облачные сервисы.
2. Экспериментально подтверждена универсальность ESP32 как аппаратной платформы, сочетающей разнообразие интерфейсов связи и приемлемое энергопотребление, что делает ее оптимальным выбором для проектов интернета вещей.
3. Проверены различные методы удаленного доступа к данным, включая специализированные облачные платформы (Arduino IoT Cloud) и популярные мессенджер (Telegram), что демонстрирует гибкость платформы в части интеграции с различными сервисами.

Практическая значимость работы подтверждается возможностью применения полученных результатов при проектировании различных систем умного дома, мониторинга и других решений интернета вещей. ESP32 представляет собой не просто инструмент для любительских проектов, а серьез-

езную платформу для создания технологических продуктов, что открывает широкие перспективы для дальнейших исследований и разработок в данной области.

Литература

1. Espressif Systems. ESP32-C3: беспроводное подключение. Полное руководство по IoT / пер. с англ. Ю.В. Ревича. М.: ДМК Пресс, 2023. 442 с. ISBN 978-5-93700-248-8.
2. Демиденко А. ESP32 для начинающих: Умный дом своими руками. Санкт-Петербург: БХВ-Петербург, 2021. 256 с.
3. Kolban N. Kolban's Book on ESP32 [б. м.] : Leanpub, 2016.
4. Eridani D., Rochim A. F., Cesara F. N. Comparative Performance Study of ESP-NOW, Wi-Fi, Bluetooth Protocols based on Range, Transmission Speed, Latency, Energy Usage and Barrier Resistance // Proceedings of the 2021 International Conference on ICT for Smart Society (ICISS). Bandung, Indonesia, 2021. P. 1-5. DOI: 10.1109/iSemantic52711.2021.9573246
5. Wijayanto M., Razak A. A., Muttaqin A. Comparison of CoAP and HTTP Protocol Performance on ESP32 Microcontroller as an Air Quality Monitoring IoT Device // 2024 12th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS). Malang, Indonesia, 2024, pp. 124-129. DOI: 10.1109/EECCIS62037.2024.10840028
6. Рычков Е. Н. Устройства умного дома и Интернета вещей на основе плат семейства ESP32. [б. м.]: Издательские решения, 2024. 320 с. ISBN 978-5-0062-7225-5.
7. Rojnarong P., Wanchalerm P. Signal Strength and Energy Consumption on Various Internet of Things Communication Protocol Using Heltec ESP32 LoRa // Proceedings of the 2024 21st International Joint Conference on Computer Science and Software Engineering (JCSSE). Phuket, Thailand, 2024, pp. 626-630. DOI: 10.1109/JCSSE61278.2024.10613704.
8. Abdul Lateef Haroon P. S., Shafiulla M., Mohammed Naveed S., Ahmed S., Mohammed Nawaz S., Kumar U. Home Automation Using Wi-Fi: ESP32-Based System for Remote Control and Environmental Monitoring // 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE). Ballari, India, 2024. P. 1-7. DOI: 10.1109/ICDCECE60827.2024.10549726
9. Rotta R., Schulz J., Naumann B., Chatharajupalli N. S., Nolte J., Werner M. B.A.T.M.A.N. Mesh Networking on ESP32's 802.11 // 2024 IEEE 49th Conference on Local Computer Networks (LCN). Normandy, France, 2024. P. 1-7. DOI: 10.1109/LCN60385.2024.10639789
10. Сувханов Д.Д. Разработка метода автоматического управления системой «Умный дом» на основе микроконтроллера ESP32 и протокола MQTT // Повышение качества жизни и обеспечение конкурентоспособности экономики на основе инновационных и научно-технических разработок : сборник статей VII Международной научно-технической конференции «Минские научные чтения – 2024», Минск, 3-5 декабря 2024 г. : в 3 т. Минск: БГТУ, 2024. Т. 2. С. 317-321.