УДК 004.89 **DOI** 10.15622/ia.24.5.7

А.М. Федотова, А.С. Романов

МЕТОДИКА ИДЕНТИФИКАЦИИ ТЕКСТОВ, СГЕНЕРИРОВАННЫХ БОЛЬШИМИ ЯЗЫКОВЫМИ МОДЕЛЯМИ

Федотова А.М., Романов А.С. **Методика идентификации текстов, сгенерированных большими языковыми моделями.**

Аннотация. В статье представлена методика идентификации русскоязычных текстов, сгенерированных большими языковыми моделями (LLM). Методика разработана с фокусом на короткие сообщения длиной от 100 до 200 символов. Актуальность работы обусловлена широким распространением генеративных моделей, таких как GPT-3.5, GPT-4o, LLaMA, GigaChat, DeepSeek, Yandex GPT. Методика основана на ансамбле моделей машинного обучения, также используются признаки трех уровней: лингвистические (структура, пунктуация, морфология, лексическое разнообразие), статистические (энтропия, перплексия, частотность n-грамм), семантические (эмбеддинги RuBERT). В качестве базовых моделей применяются LightGBM, BiLSTM и предобученная трансформерная модель RuRoBERTa, объединенные стеккингом через логистическую регрессию. Выбор гибридного ансамблевого подхода обусловлен стремлением учесть признаки на разных уровнях иерархии текста и обеспечить надежность классификации в условиях разных тематик генерируемых текстов, различных версий и видов языковых моделей. Применение ансамбля является преимуществом при анализе коротких текстов, поскольку LightGBM, опирающаяся на усредненные показатели, менее чувствительна к длине (метрика перплексии уже усреднена по всему тексту), тогда как BiLSTM и RoBERTa, способны выявлять локальные признаки LLM-текста, а не только глобальные. Набор данных естественных текстов включает более 2,8 млн пользовательских комментариев из социальной сети «ВКонтакте». Набор данных LLM-текстов содержит 700 тыс. текстов, сгенерированных семью актуальными большими языковыми моделями. При проведении генерации текстов применялись тематическое моделирование (LDA) и ролевая генерация с использованием промпт-инжиниринга. Проведена оценка методики на открытых датасетах русскоязычных LLM-текстов. Результаты экспериментов показали точность до 0,95 в задаче бинарной классификации («Человек–LLM») и до 0,89 в многоклассовой задаче определения модели-генератора. Методика демонстрирует устойчивость к разнообразию источников, стилей и версий LLM.

Ключевые слова: большие языковые модели, нейронные сети, машинное обучение, генерация текста, ансамбль классификаторов, признаки текста.

1. Введение. Стремительное развитие и широкое распространение больших языковых моделей (LLM) радикально изменили способы генерации текстовой информации. Такие модели, как GPT, LLaMA, GigaChat и другие способны создавать тексты, которые по стилю, структуре и смысловой составляющей практически неотличимы от человеческих. Использование таких моделей сопряжено с серьезными рисками, в том числе дезинформацией, нарушением авторских прав и плагиатом в академической среде [1, 2],

включая использование методов генерации исходного кода программы [3].

Особую актуальность приобретает проблема выявления текстов, созданных с помощью LLM, в социальных сетях и медиа платформах. На момент 2025 года наблюдаются случаи, когда под новостными публикациями появляются комментарии, очевидно написанные искусственным интеллектом, зачастую с целью манипуляции общественным мнением. Например, в ответ на сообщения о политических событиях можно увидеть серию синтаксически корректных, но искусственно позитивных или негативных высказываний, являющихся результатом работы ботов, использующих LLM.

Проблема усугубляется тем, что существующие методы обнаружения LLM-текстов оказываются малонадежными. В ранних исследованиях [4] показано, что точность выявления сгенерированных новостных текстов составляла лишь 0,73. С тех пор качество генерации текстов языковыми моделями значительно улучшилось, а следовательно, задача обнаружения LLM-текстов усложнилась.

Проблема идентификации LLM-текстов является актуальной и в образовательной среде. Обнаружение плагиата, определение источника текста и верификация авторства становятся все более сложными задачами. Технически подтвердить использование LLM зачастую невозможно, потому что новые версии моделей появляются быстрее, чем способные их идентифицировать инструменты.

В условиях стремительного роста объемов автоматически сгенерированного контента необходимо разрабатывать и совершенствовать надежные методики его идентификации.

Целью исследования является создание подхода к автоматическому определению коротких русскоязычных текстов, созданных большими языковыми моделями.

Научная новизна исследования заключается в разработке ансамблевого подхода, объединяющего классические методы машинного обучения (LightGBM) и глубокие нейросетевые модели (RuRoBERTa, BiLSTM), что позволяет одновременно учитывать высокоуровневые контекстные представления текста и статистико-лингвистические характеристики. Предложено комбинированное признаковое пространство, включающее лингвистические, статистические и семантические признаки, адаптированные для русскоязычных коротких текстов. Впервые представлена методика выявления коротких LLM-текстов на русском языке, протестированная на корпусе из семи актуальных и популярных моделей, включая

отечественные и зарубежные. Дополнительно проведен эксперимент по исключению каждой из рассматриваемых LLM-моделей, позволяющий оценивать устойчивость метода к появлению новых версий и типов генеративных нейросетей, отсутствующих в обучающей выборке.

2. Анализ предметной области

2.1. Программные решения. Современные методы обнаружения сгенерированных текстов делятся на две категории: статистические (вероятностные) критерии и классификационные модели [5].

Принцип действия статистических подходов заключаются в выявлении характерных признаков (артефактов) генерации без обучения на предварительно размеченном наборе данных. Например, проводится анализ распределения вероятностей слов в тексте и выявляются некоторые закономерности. В частности, сгенерированные LLM фрагменты нередко содержат слишком монотонно растущее или убывающее распределение без свойственного человеку спонтанного выбора редких и частотных слов [6]. К подобным методам относятся оценка перплексии (perplexity – мера, обратная среднему геометрическому вероятностей текста в модели языка) образцов естественных и LLM-текстов, выявление аномально высокой предсказуемости слов [7], анализ «кривизны» функции вероятности текста [8]. Низкая перплексность означает, что модель легко предсказывает слова в тексте, а высокая – что текст необычен для модели.

Главный недостаток подобных методов – направленность на конкретную языковую модель. Известные признаки и отличительные черты конкретной генеративной модели могут быть легко устранены перефразированием [9], поэтому чисто вероятностные оценки аномалий не гарантируют устойчивости к новейшим генеративным моделям.

Классификационные модели, в свою очередь, обучаются выявлять машинный текст по множеству признаков, сформированном на этапе обучения. Для этого необходимо наличие размеченного репрезентативного набора данных, содержащего достаточное количество образцов, включающих естественные тексты и образцы, сгенерированные различными типами LLM. Независимые исследования показывают, что подобный инструмент идентификации генеративных текстов, настроенный на тексты одной модели, значительно теряет эффективность, когда сталкивается с текстом, сгенерированным LLM или в ином стиле [10]. Так, детектор,

обученный на текстах GPT-3, может быть неэффективен на тексте от GPT-4 [10].

Ниже приведены наиболее известные программные решения для обнаружения LLM-текстов.

DetectGPT был предложен в 2023 г. [8]. Алгоритм основывается на гипотезе о кривизне распределения вероятностей в текстах, созданных языковой моделью. Авторы выявили, что LLM-текст обычно находится в области отрицательной кривизны функции правдоподобия этой модели, а естественный имеет другое представление функции. DetectGPT вычисляет приближенную кривизну (через метод максимального правдоподобия) по LLM и решает, является ли текст образцом этой модели. Преимущество подхода – отсутствие необходимости обучающих данных. При проведении проверок на текстах из Википедии DetectGPT получил метрику AUROC 0,95. Однако метод рассчитан на то, что для достоверного результата он должен в своей конфигурации использовать ту же (или очень похожую) языковую модель, которая могла генерировать текст. В реальных условиях заранее неизвестно, какой именно LLM создан образец, что ограничивает практическое применение инструмента. Также метод не подходит для русского языка в исходной версии, публикаций или готовых адаптаций DetectGPT под русский язык также нет.

GPTZero [11] представлен в 2023 г. и представляет собой прикладную систему определения созданных LLM-текстов для английского языка. Алгоритм GPTZero основан на перплексности текста и метрике неравномерности (вариативность сложности по фрагментам текста). GPTZero рассчитывает перплексность по собственному языковому модулю для каждого предложения и для текста в целом. Второй показатель, неравномерность (burstiness), отражает неравномерность распределения перплексности по предложениям. Человеку свойственно чередовать простые и сложные фрагменты, изменяя стиль изложения, а модель обычно поддерживает более однородный стиль во всем тексте. Первоначальная версия GPTZero была достаточно простой реализацией указанных метрик, но впоследствии разработчики добавили ансамбль, включающий глубокие нейросетевые модели. Тем не менее, перплексность и burstiness остались базовыми признаками инструмента. По заявлениям авторов, современные версии детекторов на основе GPTZero достигают точности 0,95-0,98 на длинных английских текстах. Однако при нестандартных входных данных их надежность ниже. В частности, академические обзоры отмечают, что исключительно перплексностные

инструменты уязвимы к даже небольшому перефразированию [12], например, вставке синонимов или перестановке частей предложения. Кроме того, метрика неравномерности требует объема текста от 200 слов.

Система GLTR [13] визуализирует для каждого слова его ранг вероятности по языковой модели (например, GPT-3): если текст состоит преимущественно из слов, которые модель бы выбрала как самые вероятные, то это явный признак генерации. Основной недостаток состоит в том, что для интерпретации результатов необходимо привлекать эксперта, то есть это не автономный инструмент.

OpenAI [14] в 2023 году выпустила AI Text Classifier – классификатор на базе RoBERTa, обученный различать тексты, написанные человеком, и тексты, сгенерированные существующими языковыми моделями OpenAI (в том числе разные версии ChatGPT). Модель обучалась на тройках текстов (вопрос, ответ человека, и ответ модели). Точность оказалась невысокой – классификатор правильно размечал 0,26 LLM-текстов как машинные (при пороге до 1/10 ложных срабатываний на человека). В результате, сама компания не рекомендовала полагаться на этот инструмент как на основной метод и использовать только для длинных английских текстов. Через полгода OpenAI вовсе отключила публичный доступ к классификатору.

Также можно отметить коммерческие сервисы Originality.AI [15], Copyleaks AI Detector [16], Writer.com AI Content Detector [17] и др. В основе большинства из них используют либо собственные версии перплексностных алгоритмов (близких к GPTZero), либо обученные на искусственных данных классификаторы, либо комбинацию подходов. Заявленные показатели точности для английских текстов у некоторых продуктов достигают 0,95-0,99, однако эти цифры часто отражают тесты на ограниченных наборах с конкретной LLM.

Практически все программные решения демонстрируют низкую точность на коротких образцах. Генеративный текст малой длины входит в рамки вероятностной нормы и может не содержать достаточных аномалий, чтобы его выявить. Многие сервисы устанавливают минимальный порог анализируемого текста от 200 слов, иначе результаты детекции будут недостоверными. Также появление новых версий LLM существенно осложняет задачу детекции. Так, если ранние поколения LLM (например, GPT-2) создавали тексты с заметными повторяющимися паттернами речи

и ограниченным словарем, то последние версии генерируют человекоподобный текст.

2.1. Исследования, направленные на выявление генеративных образцов на русском языке. Касаемо русского языка тема определения LLM-текстов находится на ранней стадии развития. Большинство решений опирается на зарубежные модели и алгоритмы. Для значимого прогресса требуются корпуса русскоязычных генераций и экспертиза в отладке моделей под нюансы русского языка. Группы российских исследователей из МФТИ и Сколтех участвуют в разработке новых методов. Так, Э.С. Тульчинский и соавторы [18] предложили признак внутренней размерности текстового представления, показывающий определенную устойчивость к смене домена и даже языка. Хотя общий уровень качества у этого признака пока уступает многим другим, сама идея породила обсуждение в научном сообществе.

Отдельно можно отметить систему «Антиплагиат», направленную на выявление цитирований и заимствований в академических работах. В 2024 году разработчики добавили функционал распознавания текстов, сгенерированных нейронными сетями. Система анализирует текст на наличие фрагментов, созданных с помощью языковых моделей версий GPT-2 и новее. Среди недостатков системы можно отметить то, что сервис не определяет конкретную модель, с помощью которой сгенерирован текст, а также то, что функционал обнаружения LLM-текстов доступен только в бета версии.

В 2022 г. Г. Грицай и соавторы опубликовали открытый корпус из 900 тысяч документов [19, 20] для задачи распознавания LLM-текста на русском языке. Половина текстов в этом корпусе взята из Википедии (как написанные человеком), а другая часть – сгенерирована моделями (SberAI-GPT3 small/large, Facebook XGLM), причем при генерации варьировались параметры декодирования (top-k, nucleus и пр.) и схема первичного контекста (начало предложения или отдельное слово).

Одновременно проводились конференции, стимулирующие развитие методик определения LLM-текстов. Крупнейшим из них стало соревнование Russian Artificial Text Detection (RuATD 2022) в рамках конференции «Диалог-2022» [21]. Организаторы сформировали корпус, включающий тексты 14 источников: помимо естественных текстов из открытых ресурсов, были использованы 13 моделей, генерирующих текст для различных задач: машинный перевод (напр. OPUS-MT), перефразирование, автоматическое реферирование, упрощение текста, а также генерация «с нуля» (безусловная) и через

обратный перевод [21]. Соревнование включало две задачи. Первая – определить, является ли данный текст машинным или естественным (бинарная классификация). Вторая – определить, какая именно модель (из 13 возможных) сгенерировала данный фрагмент (мультиклассовая классификация). Для обеих задач были предоставлены baseline: TF-IDF и дообученная модель ruBERT [22]. Лучшим результатом RuATD на бинарной задаче стало получение 0,83 точности командой МГУ [21]. Приглашенные эксперты-лингвисты показали лишь 0,66 точности при попытке отличить сгенерированный текст, что даже ниже базового BERT-классификатора. Лидером в мультиклассовой задаче стала команда Университета ИТМО [23]. Они протестировали несколько предобученных моделей: RuBERT, RuRoBERTa, а также крупные генеративные модели RuGPT-2 и RuGPT-3, в итоге выбрали связку из токенизации Byte-Pair Encoding и дообученного трансформера RuRoBERTa. Достижению высокой точности (0,65 на 13 классов) способствовало использование данных в исходном байтовом представлении и трансферного обучения на русскоязычном корпусе.

Помимо соревнований, публикуются и аналитические обзоры. Так, Г. Грицай и др. [24] отмечают, что ряд методик определения ИИ созданных текстов демонстрировали точность до 0,99 на тестовых наборах, однако в реальных условиях их качество резко снижается. В качестве основной причины указывается ограниченность и однородность большинства доступных корпусов.

Другой активно обсуждаемый метод – встраивание «водяных знаков» в текст на этапе генерации, позволяющих по специальным статистическим шаблонам распознать факт генерации [25]. Однако такой подход требует прямого контроля над LLM, что невозможно при использовании открытых или сторонних моделей. Для русского языка подобные решения пока не анонсированы.

OpenAI в рекомендациях к своему классификатору отмечала, что в других языках (помимо английского) его применять не следует из-за значительно низких результатов. Основных причин несколько: недостаток обучающих данных и лингвистические отличия. Англоязычные корпусы и модели превалируют в сфере обработки естественного языка, поэтому методики идентификации LLM-текстов обычно предназначены для английских текстов. Русский язык отличается свободным порядком слов, богатой морфологией и синтаксисом, что означает куда больше вариантов выразить ту же мысль. Простые шаблонные признаки (например, избыточная однотипность конструкций), которые могут указывать на машинный текст в английском, в русском языке менее надежны: человек может

переставлять части предложения, не меняя смысла, и наоборот – модель может случайно генерировать формально разнообразный, но семантически шаблонный текст.

Некоторые коммерческие сервисы заявляют поддержку русского (Copyleaks, ZeroGPT и др.), однако подробных данных об их точности нет. Также для русского языка отсутствует большая база данных широко известных «LLM-клише». Если для английского уже определены типичные фразы и шаблоны, часто выдаваемые ChatGPT, то в русской речи подобные маркеры менее изучены.

3. Методика. Методика идентификации текстов, сгенерированных большими языковыми моделями, приведена на рисунке 1.

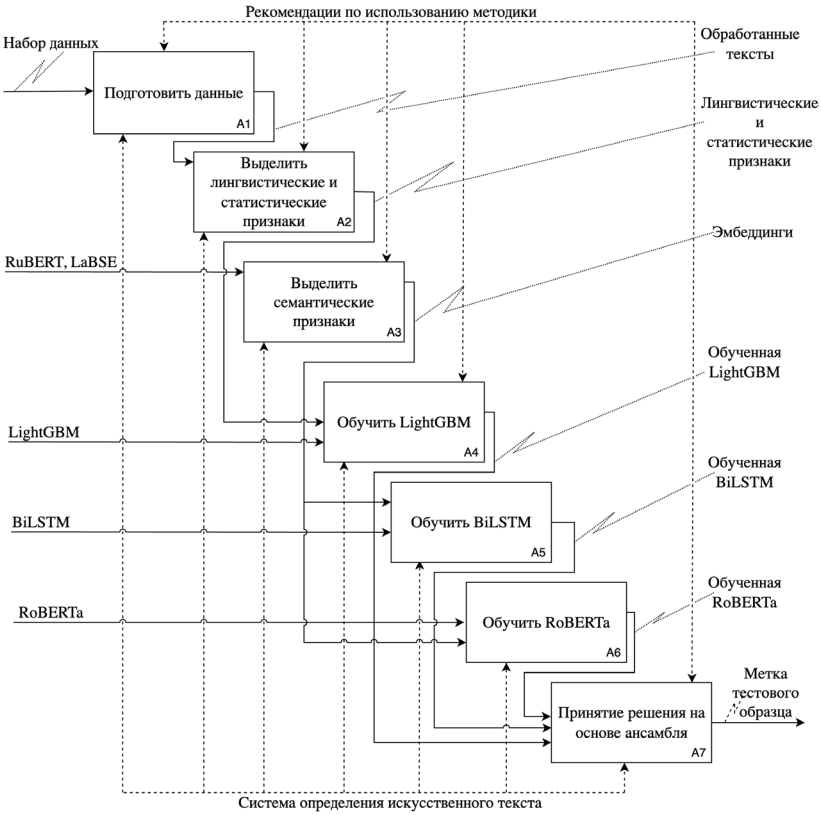

Рис. 1. Методика идентификации текстов, сгенерированных большими языковыми моделями

Пусть $D = \{(x_i, y_i)\}_{y=1}^{N}$ – обучающая выборка, где x_i – вектор признаков текста, а $y_i \in \{0,1\}$ – метка класса в задаче бинарной классификации («человек»/«LLM»), в случае многоклассовой классификации y_i имеет вид $y_i \in \{0, K\}$, где K – количество классов (LLM-моделей). Каждая модель ансамбля реализует отображение:

$$f_\theta : x \to \hat{p}_l \in [0,1]^K , \tag{1}$$

где $\hat{p}_l$ – вектор, содержащий вероятности определения текста к классам, θ – параметры модели. Целевой функцией является минимизация усредненной кросс-энтропии:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}[y_i = k]\cdot \log \hat{p}_{l,k} , \tag{2}$$

В случае бинарной классификации K =2, и формула принимает вид:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{p}_l + (1-y_i)\log(1-\hat{p}_l)], \tag{3}$$

где $y_i \in \{0,1\}$ – истинная метка класса, а $\hat{p}_l = f_\theta(x_i)$ – предсказанная вероятность принадлежности текста к классу «LLM».

К условиям достижения цели относятся минимизация $L(\theta)$ с помощью стохастического градиентного спуска и его модификаций с адаптивным шагом. Регуляризация, ранняя остановка и кросс-валидация используются для предотвращения переобучения. Для метамодели стеккинга входом служат вероятности базовых моделей, а целевая функция – кросс-энтропия.

3.1. Формирование и подготовка данных. На этапе подготовки данных сформированы два класса текстов: естественные и LLM-тексты.

Естественные тексты были собраны из социальной сети «ВКонтакте». Важно отметить, что использовались только комментарии верифицированных пользователей, то есть подтвердивших свой аккаунт с помощью учетной записи банковского приложения или Единого портала государственных услуг РФ. Обработка естественных текстов включала очистку – удаление ссылок

URL, эмодзи, тэгов. Тексты предварительно токенизировались и очищались: удалялись числовые и однобуквенные токены, исключались распространенные стоп-слова, а оставшиеся слова подвергались лемматизации. Также вводились ограничения на размер: отбирались тексты длиной от 100 до 200 символов, поскольку целью исследования было фокусирование на коротких текстах. Информация о собранном наборе приведена в таблице 1.

Таблица 1. Набор естественных текстов

Характеристика	Значение
Количество текстов	2815104
Размер набора, символов	1776835735
Размер набора, слов	27815234
Размер набора, предложений	3150806
Средняя длина текста, символов	103
Средняя длина предложения, слов	7

LLM-тексты были получены с помощью нескольких актуальных версий языковых моделей, поддерживающих русский язык, чтобы охватить различные стили генерации. Для повышения реалистичности и разнообразия синтетических данных применялись методы тематического моделирования и промпт-инжиниринга. Перед генерацией был проведен тематический анализ корпуса естественных текстов (таблица 1) с использованием Линейного дискриминантного анализа (LDA) [26]. В результате было выделено 1500 тематических кластеров. LDA-модель обучалась с параметрами α = 'symmetric', η = 'auto', passes = 10, iterations = 100 [27]; перед обучением словарь фильтровался методом filter_extremes (no_below = 50, no_above = 0.95). Каждой теме вручную присваивалась ролевая метка (например, «пенсионер», «журналист», «студент»), что позволяло формировать словарь вида «тема: роль» и передавать его в качестве контекста при генерации текстов.

Тексты генерировались с использованием API соответствующих моделей. Для повышения вариативности использовались параметры, повышающие случайность и ненаправленность ответов (temperature, frequency_penalty, presence_penalty) в соответствии с рекомендациями API. Дополнительно вводилось ограничение на длину генерируемого текста через параметр max_tokens, чтобы приблизить длину синтетических текстов к распределению длин комментариев в корпусе естественных текстов (средняя длина – 103 символа, таблица 1). Это

позволило сделать тексты более сопоставимыми по структуре и объему.

Каждому сгенерированному тексту присваивалась метка, отражающая источник (конкретная LLM), однако в рамках бинарной классификации все такие тексты рассматривались как единый обобщенный класс «сгенерировано LLM». Информация о наборе LLM-текстов приведена в таблице 2.

Таблица 2. Набор LLM-текстов

Модель	Количество текстов
LLaMA (Яндекс) [29]	100000
Yandex GPT [30]	100000
Yandex GPT-lite [30]	100000
GPT-4o [23]	100000
GPT-3.5 [23]	100000
GigaChat Pro [31]	100000
DeepSeek [32]	100000
Итого	700000

Также методика была протестирована на наборе данных Г. Грицай и др. [20] и датасете RuATD-2022 [21]. Информация о датасетах представлена в таблице 3.

Таблица 3. Характеристики датасетов Г. Грицай и RuATD-2022

Тип текста	Характеристика набора	Значение для датасета Г. Грицай	Значение для датасета RuATD-2022
Ест.	Кол-во текстов	451572	59674
Иск.		451572	155436
Ест.	Кол-во символов	1359568468	15679465
Иск.		1684485630	33655209
Ест.	Ср. длина текста, символов	3011	263
Иск.		3730	217
Ест.	Мин. длина текста, символов	500	7
Иск.		302	6
Ест.	Макс. длина текста, символов	429800	2963
Иск.		15281	3560

3.2. Признаковое пространство. Чтобы выявить неявные различия между естественными и сгенерированными текстами, из каждого текста извлекался набор признаков разного типа:

1. Лингвистические признаки. Данная группа включает признаки, характеризующие стиль и структуру текста на уровне явных отличимых человеком лингвистических особенностей. В частности, рассчитывалась средняя длина предложения – предполагалось, что LLM могут продуцировать более формализованные или однотипные по длине предложения. Также рассчитывалось количество знаков препинания (общее число и среднее) для отражения сложности синтаксической структуры. Морфологические признаки включали частотные распределения частей речи. Также в данную категорию признаков входили метрики лексического разнообразия: доля редких и распространенных слов. Для этого использовался частотный словарь русского языка [33].

2. Статистические признаки. К этой категории относятся количественные меры энтропии и вероятностной непредсказуемости текста. Вычислялась энтропия символов и слов – как среднее значение по всему тексту. Энтропия для последовательности символов определялись стандартно через частоты появления каждого символа. Аналогично вычислялась энтропия распределения слов, рассматривая частоты слов. Низкая энтропия указывает на однообразие (повторяемость одних и тех же слов или букв), тогда как высокая – на вариативность. Предполагалось, что сгенерированные тексты могут обладать отличной от естественных текстов статистической структурой (например, некоторые модели могут иметь избыточную равномерность или, наоборот, неестественно низкое разнообразие, особенно на коротких отрывках). Также рассчитывалась перплексия. Для расчета данного признака была обучена простая статистическая языковая модель (5-граммная модель) на корпусе естественных русских текстов, используя KenLM [34]. Полученная модель рассчитывала вероятность последовательности символов, и на ее основе вычислялась перплексия каждого рассматриваемого текста. Естественные тексты должны быть «более предсказуемыми» (низкая перплексия) относительно статистики человеческого языка, тогда как LLM-тексты могут содержать менее характерные сочетания слов, повышающие перплексию.

Также рассчитывались частоты биграмм и триграмм для выявления их повторяемости в тексте. Высокое число повторяющихся последовательностей слов и символов может свидетельствовать о том, что модель склонна к использованию определенных фраз или шаблонов, тогда как у человека в таком коротком тексте обычно меньше точных повторов.

3. Семантические признаки. Для выявления неявных признаков использовались эмбеддинги текста, полученные с помощью современных трансформерных моделей. Применен RuBERT от DeepPavlov [35]. С помощью RuBERT каждый текст был преобразован в вектор фиксированной размерности (768-мерный). Модель RuBERT обучена на русском корпусе Википедии и новостей, поэтому ее скрытые состояния отражают контекстуальные зависимости слов на русском языке. LLM при генерации склонны включать лишнюю информацию или использовать более общие формулировки, что приводит к характерным отличиям в эмбеддингах. Некоторые координаты этих векторов (или их линейные комбинации) оказываются информативными для классификации и отражают стилистические и контекстуальные особенности текстов. Используются эмбеддинги RuBERT (*hidden size = 768*). Далее этот вектор подается на вход BiLSTM.

Для блока статистических и лингвистических признаков использовались доля знаков препинания, частота использования эмодзи, средняя длина слова, коэффициент лексического разнообразия (TTR), частоты стоп-слов, количество предложений, распределение частей речи, частота заглавных букв, частоты повторяющихся символов. Семантические признаки формировались на основе 768-мерных эмбеддингов RuBERT и косинусных расстояний между предложениями.

Веса признаков для LightGBM оценивались по показателю прироста информации (gain) – рисунок 2.

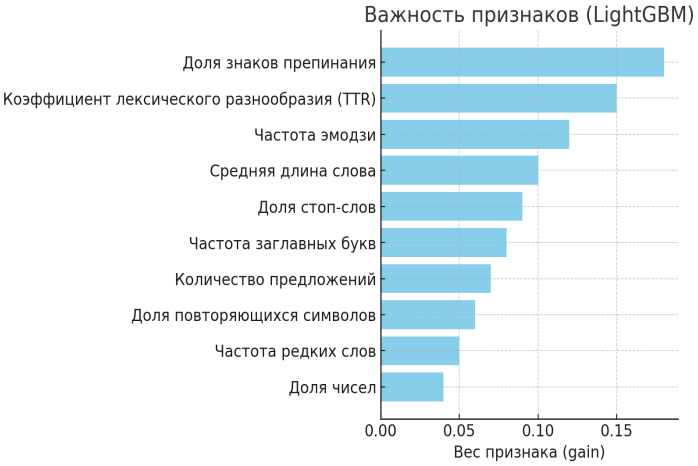

Рис. 2. Оценка важности признаков

Наибольший вклад внесли: доля знаков препинания (0,18), коэффициент лексического разнообразия (TTR) (0,15), частота эмодзи (0,12) и средняя длина слова (0,10). В Transformer-модели наибольшие веса пришлись на знаки препинания и редкие слова, что коррелировало с важностью признаков в LightGBM.

3.3. Архитектура. Для решения задачи классификации («Человек-LLM») предложена ансамблевая архитектура, объединяющая несколько моделей различного типа (рисунок 3).

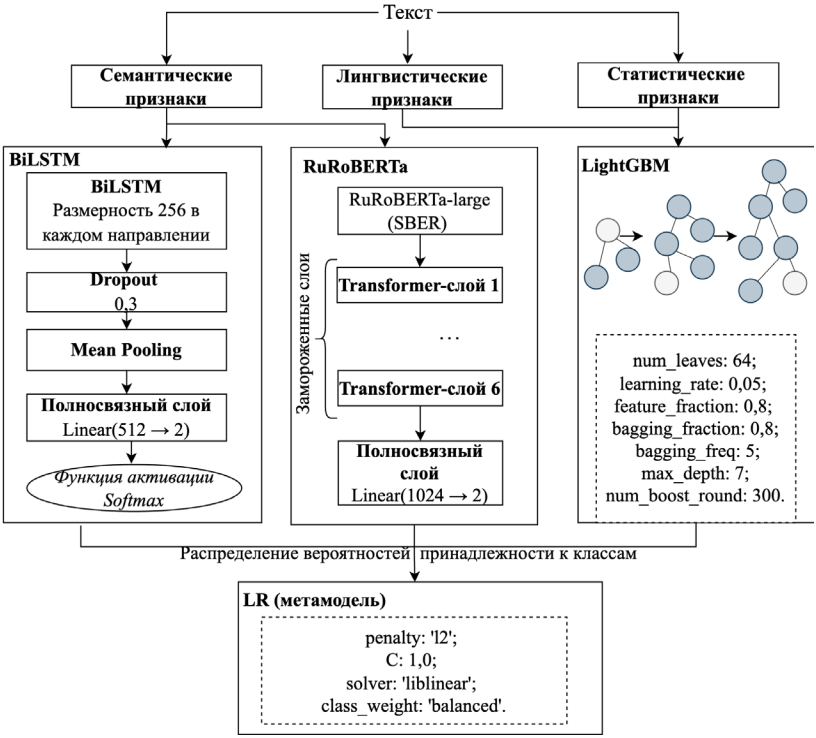

Рис. 3. Архитектура предложенной модели для выявления сгенерированных текстов

Такая комбинация позволяет учесть разнородные признаки текста (раздел 3.2). В состав ансамбля вошли следующие компоненты:

1. Модель градиентного бустинга LightGBM. LightGBM используется для обучения на лингвистических и статистических характеристиках, описанных в разделе 3.2. Вектор признаков каждого

текста подается на вход LightGBM. Модель выбрана из-за ее скорости обучения и способности справляться со смешанными по природе признаками, а также встроенной функции оценки важности признаков. Обученная модель выдает для каждого текста оценку принадлежности к классу «LLM-текст» (вероятность или логит). Параметры обучения LightGBM: objective: 'binary' (задача бинарной классификации (человек/LLM)); metric: 'binary_logloss' (логарифмическая функция потерь для бинарной классификации); boosting_type: 'gbdt' (градиентный бустинг по деревьям решений); num_leaves: 64 (максимальное количество листьев в одном дереве); learning_rate: 0,05 (шаг градиентного обновления); feature_fraction: 0,8 (доля признаков, случайно выбираемых для построения каждого дерева); bagging_fraction: 0,8; bagging_freq: 5 (обучение на случайных подвыборках каждые 5 итераций); max_depth: 7 (максимальная глубина дерева, ограничивающая переобучение); num_boost_round: 300 (общее количество итераций); early_stopping_rounds: 30 (остановка обучения при отсутствии улучшений).

2. BiLSTM. В качестве второго компонента выбрана модель двунаправленной рекуррентной нейросети с долгой краткосрочной памятью (LSTM) – BiLSTM. В данном случае на вход BiLSTM подается последовательность эмбеддингов, представляющих текст. К причинам выбора BiLSTM в качестве компонента ансамбля отнесем ее способность анализировать текст в двух направлениях, что особенно полезно для коротких текстов, где каждое слово особенно влияет на семантику всего текста; устойчивость к нестандартной грамматике и синтаксису – LLM часто генерируют тексты с гладкой, но шаблонной структурой. BiLSTM способна уловить тонкие различия в синтаксисе, которые могли бы ускользнуть от классических моделей.

Параметры модели BiLSTM: размерность входного эмбеддинга: 768; размер скрытого состояния: 256 (в каждом направлении, количество нейронов в каждом направлении рекуррентной сети); количество LSTM-слоев: 1; dropout: 0,3 (регуляризация, предотвращающая переобучение); агрегация выходов: усреднение (mean pooling); полносвязный классификационный слой: Linear(512 → 2) – полносвязный слой, объединяющий оба направления; активация: Softmax. Обучение производилось с использованием оптимизатора 'Adam', learning rate: 0,001 (скорость обучения), batch size: 32 (размер батча), число эпох: 10.

3. RuRoBERTa. Третий модуль ансамбля – предобученная модель-трансформер семейства RoBERTa, адаптированная под русский язык. В работе применена модель, предоставленная командой

SberDevices [36]. Для задачи бинарной классификации («Человек-LLM») к модели применялась техника fine-tuning, а именно, добавлен полносвязный классификационный слой, проведено дообучение модели методом градиентного спуска на размеченном датасете. Большие модели вроде RoBERTa способны обнаруживать тонкие стилистические различия, недоступные более простым моделям, поэтому данный компонент должен улавливать высокоуровневые признаки, специфичные для машинного текста. Модель дообучалась под задачу бинарной классификации «Человек-LLM» по следующей схеме: добавлен классификационный слой Linear(1024 → 2) – полносвязный слой; оптимизатор: 'AdamW'; learning rate: 2e-5 (небольшой шаг обучения, характерный для дообучения крупных моделей); batch size: 16 (размер батча); число эпох: 3; стратегия заморозки: первые 6 слоев трансформера заморожены; warmup steps: 500; scheduler: 'linear decay with warmup' (линейное уменьшение параметра learning rate после фазы warmup); weight decay: 0,01 (регуляризация весов); loss function: 'CrossEntropyLoss' (кросс-энтропия для задачи классификации); fine-tuning осуществлялся на размеченном корпусе с использованием библиотеки Hugging Face Transformers.

4. Метамодель (стеккинг). Финальное решение принимает ансамблевый мета классификатор, который агрегирует выходы трех перечисленных моделей. На этапе обучения для каждого объекта вычислялись предсказанные вероятности базовых моделей. Затем эти три значения использовались как входные признаки для обучающейся метамодели. В качестве мета классификатора была использована логистическая регрессия (LR) чтобы не переобучиться на выходах. Такой подход соответствует технике стеккинг – базовые алгоритмы обучаются отдельно, а затем их прогнозы комбинируются обученным способом. Параметры метамодели: penalty: 'l2' (L2-регуляризация для предотвращения переобучения); C: 1,0 (коэффициент регуляризации); solver: 'liblinear'; class_weight: 'balanced' (автоматическая корректировка весов классов для борьбы с дисбалансом). Настройка мета-классификатора осуществляется на основе обучающего корпуса коротких текстов (100–200 символов), что позволяет модели учитывать специфику признаков именно для этого диапазона длины. Динамическое изменение весов компонентов ансамбля в зависимости от длины текста в текущей версии не применяется, однако рассматривается как потенциальное улучшение при адаптации методики к более длинным образцам.

Выбор оптимизаторов «Adam» и «AdamW» обусловлен их стабильной сходимостью при работе с короткими текстами. Для снижения риска переобучения применялись регуляризация (dropout, weight decay), ранняя остановка (early stopping), а также подбор шага обучения. Альтернативные оптимизаторы (SGD, RMSProp) были протестированы на части выборки и показали более медленную сходимость и менее стабильные результаты в задаче бинарной классификации.

Выбор гибридного ансамблевого подхода обусловлен стремлением учесть признаки на разных уровнях иерархии текста и обеспечить надежность классификации в условиях разных тематик генерируемых текстов, различных версий и видов языковых моделей. Каждая из использованных моделей имеет свои сильные стороны, и их комбинация компенсирует слабости друг друга, а именно:

1. Лингвистические и статистические признаки дают понятные с точки зрения лингвистики индикаторы «неестественности» текста. Однако эти признаки могут быть не способны к выявлению сложных контекстуальных признаков и закономерностей. С другой стороны, эмбеддинги, как более глубокое признаковое представление, способны выявлять неявные семантические паттерны. Объединение через ансамбль позволяет учитывать и обрабатывать оба типа признаков.

2. В корпус включены тексты, созданные разными LLM, каждая из которых имеет свои характерные черты. Ансамбль лучше адаптируется к такому разнообразию источников текстов. Например, если новая модель генерации не употребляет редких слов (тем самым классифицируется одной из моделей как естественный текст), все еще могут сработать другие модели ансамбля. Таким образом, использование ансамбля, по сути, дает дополнительную защиту против разных видов отклонений LLM-текста, одни компоненты ансамбля реагируют на статистические аномалии, другие – на стилистические. Это особенно важно, поскольку LLM быстро развиваются, конкретная генеративная модель может утратить актуальность, если появится модель, генерирующая тексты в ином стиле. Представленная методика изначально более устойчива к появлению новых LLM: даже если они маскируют одни признаки, останутся задействованы другие.

3. Короткие тексты (в районе 100-200 символов) крайне сложны для классификации, т.к. признаковое пространство ограничено. Длинные тексты, наоборот, дают больше материала для анализа, но модели могут терять фокус на ключевых особенностях. Применение ансамбля является преимуществом в таких случаях,

поскольку LightGBM, опирающаяся на усредненные показатели, менее чувствительна к длине (метрика перплексии уже усреднена по всему тексту), тогда как BiLSTM и RoBERTa, способны выявлять локальные признаки LLM-текста, а не только глобальные. В случае коротких текстов ключевым фактором при принятии решения могут быть численные признаки (например, очень низкая энтропия символов сразу выдаст шаблонный сгенерированный ответ), а при анализе более длинных образцов – глубокие нейросетевые модели выявят несвойственный человеку ход рассуждения за счет выявления неявных локальных признаков через эмбеддинги.

Для свертки предсказаний разнородных моделей применяется логистическая регрессия, обучаемая на вероятностях, выдаваемых каждой из базовых моделей. Таким образом, веса (или степень доверия) к каждому классификатору в ансамбле не задаются вручную, а определяются в процессе обучения мета-классификатора, что позволяет учитывать специфику их работы на различных типах входных данных.

4. Результаты. Для надежной оценки качества классификатора и настройки гиперпараметров была применена процедура кросс-валидации на 5 фолдах. Итоговый корпус был сбалансирован и разделен на обучающую и тестовую выборки в соотношении 80:20, при этом тексты перемешаны случайным образом, чтобы ни одна конкретная тема или источник не преобладали только в одной из выборок.

Эксперименты проводились по двум задачам: бинарная классификация («Человек-LLM») и многоклассовая классификация (определение «Какой моделью создан текст?»). При проведении всех экспериментов наборы данных были сбалансированы, исключались тексты длиной менее 100 символов.

Результаты эксперимента представлены для задачи «Человек-LLM» в таблице 4, для задачи «Какой моделью создан текст?» – в таблице 5.

В дополнение к основным экспериментам мы провели leave-one-model-out тестирование, в котором на каждой итерации полностью исключали из обучения тексты одной LLM-семьи (например, Яндекс-модели или OpenAI) и тестировали только на ней. Аналогично, внутри выбранной LLM-семьи выполняли leave-one-version-out тест (например, обучали на GPT-3.5, тестировали на GPT-4o, и наоборот). Для каждой исключенной модели мы рассчитывали Accuracy, Precision, Recall и F1. Результаты данного эксперимента приведены в таблице 6.

Таблица 4. Результаты эксперимента «Человек-LLM»

Модель	Архитектура	Accuracy	Precision	Recall	F1
LLaMA (Яндекс)	RuRoBERTa	0,86±0,02	0,87±0,02	0,85±0,03	0,86±0,02
	BiLSTM	0,85±0,02	0,86±0,03	0,84±0,02	0,85±0,03
	LightGBM	0,82±0,03	0,83±0,02	0,81±0,02	0,82±0,02
	Ансамбль	0,9±0,03	0,91±0,02	0,89±0,03	0,9±0,03
Yandex GPT	RuRoBERTa	0,83±0,03	0,84±0,02	0,82±0,03	0,83±0,03
	BiLSTM	0,82±0,03	0,83±0,03	0,81±0,03	0,82±0,03
	LightGBM	0,78±0,03	0,79±0,03	0,77±0,02	0,78±0,03
	Ансамбль	0,87±0,03	0,88±0,03	0,86±0,02	0,87±0,03
Yandex GPT-lite	RuRoBERTa	0,83±0,03	0,84±0,02	0,82±0,03	0,83±0,03
	BiLSTM	0,83±0,02	0,84±0,03	0,82±0,02	0,83±0,02
	LightGBM	0,79±0,04	0,8±0,03	0,78±0,04	0,79±0,03
	Ансамбль	0,91±0,02	0,92±0,03	0,9±0,02	0,91±0,02
GPT-4o	RuRoBERTa	0,78±0,03	0,79±0,03	0,77±0,03	0,78±0,03
	BiLSTM	0,77±0,04	0,78±0,03	0,76±0,04	0,77±0,04
	LightGBM	0,75±0,03	0,76±0,03	0,74±0,03	0,75±0,03
	Ансамбль	0,82±0,02	0,83±0,03	0,81±0,02	0,82±0,02
GPT-3.5	RuRoBERTa	0,84±0,04	0,85±0,03	0,83±0,04	0,84±0,04
	BiLSTM	0,83±0,03	0,84±0,03	0,82±0,03	0,83±0,02
	LightGBM	0,78±0,03	0,79±0,03	0,77±0,03	0,78±0,03
	Ансамбль	0,88±0,04	0,89±0,03	0,87±0,04	0,88±0,03
GigaChat Pro	RuRoBERTa	0,9±0,04	0,91±0,04	0,89±0,04	0,9±0,03
	BiLSTM	0,89±0,03	0,9±0,03	0,88±0,02	0,89±0,03
	LightGBM	0,87±0,02	0,88±0,03	0,86±0,02	0,87±0,02
	Ансамбль	0,95±0,03	0,96±0,03	0,94±0,03	0,95±0,03
DeepSeek	RuRoBERTa	0,84±0,04	0,85±0,04	0,83±0,04	0,84±0,03
	BiLSTM	0,84±0,03	0,85±0,03	0,83±0,02	0,84±0,03
	LightGBM	0,8±0,02	0,81±0,02	0,79±0,02	0,8±0,03
	Ансамбль	0,89±0,03	0,9±0,03	0,88±0,03	0,89±0,03
Все модели	RuRoBERTa	0,8±0,02	0,81±0,02	0,79±0,02	0,8±0,02
	BiLSTM	0,81±0,03	0,82±0,03	0,8±0,03	0,81±0,03
	LightGBM	0,77±0,02	0,78±0,02	0,76±0,03	0,77±0,03
	Ансамбль	0,84±0,03	0,85±0,02	0,83±0,03	0,84±0,03

Таблица 5. Результаты эксперимента «Какой моделью создан текст?»

Модель	Accuracy	Precision	Recall	F1
Yandex модели (2 класса)	0,89±0,03	0,9±0,02	0,88±0,03	0,89±0,03
Open AI модели (2 класса)	0,76±0,02	0,77±0,03	0,75±0,02	0,76±0,02
Все модели (7 классов)	0,6±0,04	0,62±0,03	0,59±0,03	0,6±0,03

Таблица 6. Оценка обобщаемости методики при исключении моделей и семейств LLM-моделей

Исключенная модель	Accuracy	Precision	Recall	F1
Yandex GPT	0,8±0,04	0,79±0,03	0,8±0,03	0,8±0,03
Yandex GPT-lite	0,83±0,03	0,82±0,02	0,82±0,04	0,82±0,03
GigaChat Pro	0,9±0,04	0,88±0,03	0,89±0,03	0,89±0,03
DeepSeek	0,86±0,02	0,85±0,03	0,84±0,02	0,85±0,03
GPT-4o	0,76±0,04	0,74±0,03	0,75±0,03	0,75±0,02
GPT-3.5	0,83±0,04	0,83±0,02	0,81±0,02	0,82±0,02

Результаты подтверждают эффективность предложенной методики. В задаче «Человек-LLM» ансамбль продемонстрировал стабильную и высокую точность на различных LLM, включая модели, ориентированные на русский язык (Yandex GPT, GigaChat Pro) и мультилингвальные разработки (OpenAI, DeepSeek). Средняя точность распознавания для отдельных моделей достигала 0,95 (GigaChat Pro), при этом разброс по моделям составил 0,82–0,95 (таблица 5). Совокупная точность по всем LLM составила 0,84, что существенно превосходит другие решения, предназначенные для анализа русского языка за счет выбора моделей, входящих в состав ансамбля и комбинации различных видов признаков.

В задаче «Какой моделью создан текст» точность идентификации конкретной LLM достигала 0,89 для группы Yandex-моделей и 0,76 для моделей OpenAI (таблица 6). Несмотря на естественное снижение точности в эксперименте с 7 классами (до 0,6), результат остается приемлемым с учетом стилистического сходства между генераторами и их способности адаптировать стиль под естественный текст.

Эксперименты показали, что при тестировании на моделях, отсутствующих в обучении, качество снижается на 0,03-0,08. относительно классических сценариев эксперимента, но остается не менее 0,76.

Для сравнения предложенной методики с результатами других авторов было проведено два эксперимента на общедоступных наборах данных (датасет Г. Грицай и соавторов, RuATD-2022 [21]). Эксперимент (таблица 7) показывает, что методика способна демонстрировать высокую точность 0,89 на датасетах, содержащих тексты, отличающиеся от представленного в работе набора (по длине, стилистике содержимого, типам генеративных моделей и др.).

Таблица 7. Результаты экспериментов на общедоступных наборах данных

Датасет	Авторы	Метод	Результат
RuATD binary [21]	Г. Грицай и др. [38]	DeBERTa	F1-мера 0,76
	Команда МГУ [37]	Ансамбль моделей Transformer	Точность 0,83
	Авторы статьи	**Ансамбль RuRoBERTa, LightGBM, BiLSTM, LR как метамодель**	**Точность 0,87**
Набор данных Г. Грицай и др. [19]	Ю.В. Чехович и др. [38]	XLM-RoBERTa	Точность 0,89
	Авторы статьи	**Ансамбль RuRoBERTa, LightGBM, BiLSTM, LR как метамодель**	**Точность 0,89**

5. Выводы и обсуждение. Разработанная методика позволяет эффективно выявлять тексты, сгенерированные большими языковыми моделями. Комбинация лингвистических, статистических и семантических признаков, а также использование ансамбля классификаторов обеспечивают точность до 0,95. Полученные результаты подтверждают целесообразность применения данного подхода для задач модерации контента, анализа достоверности информации и противодействия дезинформации.

Среди ограничений методики можно отметить специфику разработки исключительно для анализа русскоязычных текстов, обучение на текстах разговорного стиля (комментариев пользователей социальных сетях).

Дальнейшие планы исследования включают проведение анализа и адаптацию методики к другим стилям текстов, включая научный, официально-деловой, публицистический и художественный, расширение существующего набора данных с появлением новых версий LLM. Также возможно совершенствование стеккинговой модели, в рамках которого добавляются дополнительные признаки (например, длина текста или степень перплексии), которые могут влиять на «доверие» к отдельным компонентам ансамбля.

Литература

1. Fedotova A., Romanov A., Kurtukova A., Shelupanov A. Digital authorship attribution in Russian-language fanfiction and classical literature // Algorithms. 2022. vol. 16. no. 1.
2. Романов А.С. Методология идентификации автора текста для решения задач информационной безопасности // Вопросы кибербезопасности. 2024. № 3(61). С. 120–128. DOI: 10.21681/2311-3456-2024-3-120-128.
3. Kurtukova A., Romanov A., Shelupanov A., Fedotova A. Complex cases of source code authorship identification using a hybrid deep neural network // Future Internet. 2022. vol. 14. no. 10. DOI: 10.3390/fi14100287.
4. Zellers R., Holtzman A., Rashkin H., Bisk Y., Farhadi A., Roesner F., Choi Y. Defending against neural fake news // Proceedings of the 33rd Int. Conf. on Neural Information Processing Systems. 2019. pp. 9054–9065.
5. Kuznetsov, K., Tulchinskii E., Kushnareva L., Magai G., Baranniko S., Nikolenko S., Piontkovskaya I. Robust AI-Generated Text Detection by Restricted Embeddings // Findings of the Association for Computational Linguistics: EMNLP. 2024. pp. 17036–17055. DOI: 10.18653/v1/2024.findings-emnlp.992.
6. Fraser K.C., Dawkins H., Kiritchenko S. Detecting AI-Generated Text: Factors Influencing Detectability with Current Methods // Journal of Artificial Intelligence Research. 2025. vol. 82. pp. 2233–2278. DOI: 10.1613/jair.1.16665.
7. Prajapati M., Baliarsingh S.K., Dora C., Bhoi A., Hota J., Mohanty J.P. Detection of AI-generated text using large language model // 2024 International Conference on Emerging Systems and Intelligent Computing (ESIC). 2024. pp. 735–740. DOI: 10.1109/ESIC60604.2024.10481602.
8. Mitchell E., Lee Y., Khazatsky A., Manning C.D., Finn C. Detectgpt: Zero-shot machine-generated text detection using probability curvature // Proceedings of the 40th International Conference on Machine Learning (PMLR). 2023. pp. 24950–24962.
9. Lau H.T., Zubiaga A. Understanding the Effects of Human-written Paraphrases in LLM-generated Text Detection // arXiv preprint arXiv:2411.03806. 2024.
10. Wu J., Yang S., Zhan R., Yuan Y., Chao L.S., Wong D.F. A survey on LLM-generated text detection: Necessity, methods, and future directions // Computational Linguistics. 2025. pp. 275–338. DOI: 10.1162/coli_a_00549.
11. GPTZero. URL: gptzero.me (дата обращения: 15.05.2025).
12. Kavian A., Pourhashem Kallehbasti M.M., Kazemi S., Firouzi E., Ghafari M. LLM security guard for code // Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering. 2024. pp. 600–603. DOI: 10.1145/3661167.366126.
13. Wu L. Y., Segura-Bedmar I. AI-generated Text Detection with a GLTR-based Approach // arXiv preprint. arXiv:2502.12064. 2025.

14. OpenAI. URL: openai.com (дата обращения: 15.05.2025).
15. OriginalityAI. URL: https://originality.ai/ (дата обращения: 15.05.2025).
16. AI Detector & Content Checker By Copyleaks. URL: https://copyleaks.com/ai-content-detector (дата обращения: 15.05.2025).
17. Writer. AI content detector. URL: https://writer.com/ai-content-detector/ (дата обращения: 15.05.2025).
18. Tulchinskii E., et al. Intrinsic dimension estimation for robust detection of AI-generated texts // Advances in Neural Information Processing Systems. 2023. vol. 36. pp. 39257–39276.
19. Nikolaev K. Development of a Neural Network Model for Recognizing Russian-Language Generated Texts // 2024 IEEE Int. Multi-Conf. on Engineering, Computer and Information Sciences (SIBIRCON). IEEE, 2024. pp. 396–400. DOI: 10.1109/SIBIRCON63777.2024.10758447.
20. Gritsay G., Grabovoy A., Chekhovich Y. Open access dataset for machine-generated text detection in Russian. Mendeley Data. V2. 2023. DOI: 10.17632/4ynxfp3w53.2.
21. Shamardina T., et al. Findings of the the ruatd shared task 2022 on artificial text detection in Russian // arXiv preprint arXiv:2206.01583. 2022.
22. RuATD. URL: https://github.com/dialogue-evaluation/RuATD (дата обращения: 15.05.2025).
23. Skrylnikov S., Posokhov P., Makhnytkina O. Artificial text detection in Russian language: A BERT-based approach // Proc. Int. Conf. Dialogue. 2022. pp. 1–7.
24. Gritsai G., Voznyuk A., Grabovoy A., Chekhovich Y. Are AI Detectors Good Enough? A Survey on Quality of Datasets With Machine-Generated Texts // arXiv e-prints. 2024. arXiv:2410.14677.
25. Pan L., et al. MarkLLM: An Open-Source Toolkit for LLM Watermarking // Proc. of the 2024 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations. 2024. pp. 61–71. DOI: 10.18653/v1/2024.emnlp-demo.7.
26. Pham C.M., et al. TopicGPT: A Prompt-based Topic Modeling Framework // arXiv e-prints. 2023. arXiv:2311.01449.
27. Tong Z., Zhang H. A text mining research based on LDA topic modelling // International conference on computer science, engineering and information technology. 2016. pp. 201–210. DOI : 10.5121/csit.2016.60616.
28. Geroimenko V. Key Principles of Good Prompt Design // The Essential Guide to Prompt Engineering: Key Principles, Techniques, Challenges, and Security Risks. Cham: Springer Nature Switzerland. 2025. pp. 17–36.
29. Модели генерации текста. URL: https://yandex.cloud/ru/docs/foundation-models/concepts/yandexgpt/models (дата обращения: 15.05.2025).
30. Yandex GPT. URL: https://ya.ru/ai/gpt (дата обращения: 15.05.2025).
31. GiGaChat. URL: https://giga.chat/ (дата обращения: 15.05.2025).
32. DeepSeek. URL: https://www.deepseek.com/ (дата обращения: 15.05.2025).
33. Кузнецов С.А. Большой толковый словарь русского языка. Shangwu Yinshuguan, 2020. 1481 с.
34. KenLM. URL: https://github.com/kpu/kenlm (дата обращения: 15.05.2025).
35. Savkin M., Voznyuk A., Ignatov F., Korzanova A., Karpov D., Popov A., Konovalov V. DeepPavlov 1.0: Your Gateway to Advanced NLP Models Backed by Transformers and Transfer Learning // Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2024. pp. 465–474. DOI: 10.18653/v1/2024.emnlp-demo.47.
36. RuRoBERTa-large. URL: https://huggingface.co/ai-forever/ruRoberta-large (дата обращения: 15.05.2025).
37. Maloyan N., Nutfullin B., Ilyushin E. Dialog-22 ruatd generated text detection // arXiv preprint arXiv:2206.08029. 2022.

38. Gritsay G., Grabovoy A., Chekhovich Y. Automatic detection of machine generated texts: Need more tokens // 2022 Ivannikov Memorial Workshop (IVMEM). IEEE, 2022. pp. 20–26. DOI: 10.1109/IVMEM57067.2022.9983964.

Федотова Анастасия Михайловна — младший научный сотрудник, научно-инжиниринговый центр «интеллектуальные системы доверенного взаимодействия», ТУСУР. Область научных интересов: информационная безопасность, искусственный интеллект, машинное обучение, deep learning, NLP. Число научных публикаций — 20. afedotowaa@yandex.ru; проспект Ленина, 40, 634050, Томск, Россия; р.т.: +7(923)444-4125.

Романов Александр Сергеевич — канд. техн. наук, доцент кафедры, кафедра комплексной информационной безопасности электронно-вычислительных систем (кибэвс), ТУСУР. Область научных интересов: информационная безопасность, интеллектуальный анализ данных, искусственный интеллект, обработка текста. Число научных публикаций — 76. alexx.romanov@gmail.com; проспект Ленина, 40, 634050, Томск, Россия; р.т.: +7(923)444-4125.

Поддержка исследований. Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ в рамках базовой части государственного задания ТУСУРа на 2023–2025 гг. (проект № FEWM-2023-0015).

DOI 10.15622/ia.24.5.7

A. FEDOTOVA, A. ROMANOV

TECHNIQUE FOR IDENTIFYING TEXTS GENERATED BY LARGE LANGUAGE MODELS

Fedotova A., Romanov A. **Technique for Identifying Texts Generated by Large Language Models.**

Abstract. The article presents a method for identifying Russian-language texts generated by large language models (LLMs). The method was developed with a focus on short messages from 100 to 200 characters long. The relevance of the work is due to the widespread use of generative models, such as GPT-3.5, GPT-4o, LLaMA, GigaChat, DeepSeek, and Yandex GPT. The method is based on an ensemble of machine learning models, and features of three levels are also used: linguistic (structure, punctuation, morphology, lexical diversity), statistical (entropy, perplexity, n-gram frequency), and semantic (RuBERT embeddings). LightGBM, BiLSTM, and the pre-trained transformer model RuRoBERTa are used as basic models, combined by stacking through logistic regression. The choice of a hybrid ensemble approach is due to the desire to take into account features at different levels of the text hierarchy and to ensure the reliability of classification in the context of different topics of generated texts, versions, and types of language models. The use of an ensemble is an advantage in the analysis of short texts, since LightGBM, based on averaged indicators, is less sensitive to length (the perplexity metric is already averaged over the entire text), while BiLSTM and RoBERTa are able to identify local features of an LLM text, and not just global ones. The dataset of natural texts includes more than 2.8 million user comments from the VK social network. The LLM text dataset contains 700 thousand texts generated by seven relevant large language models. Topic modeling (LDA) and role generation using prompt engineering were used in the text generation. The methodology was evaluated on open datasets of Russian-language LLM texts. The experimental results showed an accuracy of up to 0.95 in the binary classification task (Human–LLM) and up to 0.89 in the multi-class task of determining the model-generator. The method demonstrates robustness to the diversity of sources, styles, and LLM versions.

Keywords: large language models, neural networks, machine learning, text generation, ensemble of classifiers, text features.

References

1. Fedotova A., Romanov A., Kurtukova A., Shelupanov A. Digital authorship attribution in Russian-language fanfiction and classical literature. Algorithms. 2022. vol. 16. no. 1.
2. Romanov A.S. [Methodology for identifying the author of the text for solving information security problems]. Voprosy kiberbezopasnosti – Cybersecurity issues. 2024. no. 3(61). pp. 120–128. DOI: 10.21681/2311-3456-2024-3-120-128. (In Russ.).
3. Kurtukova A., Romanov A., Shelupanov A., Fedotova A. Complex cases of source code authorship identification using a hybrid deep neural network. Future Internet. 2022. vol. 14. no. 10. DOI: 10.3390/fi14100287.
4. Zellers R., Holtzman A., Rashkin H., Bisk Y., Farhadi A., Roesner F., Choi Y. Defending against neural fake news. Proceedings of the 33rd Int. Conf. on Neural Information Processing Systems. 2019. pp. 9054–9065.
5. Kuznetsov, K., Tulchinskii E., Kushnareva L., Magai G., Baranniko S., Nikolenko S., Piontkovskaya I. Robust AI-Generated Text Detection by Restricted Embeddings. Findings of the Association for Computational Linguistics: EMNLP. 2024. pp. 17036–17055. DOI: 10.18653/v1/2024.findings-emnlp.992.

6. Fraser K.C., Dawkins H., Kiritchenko S. Detecting AI-Generated Text: Factors Influencing Detectability with Current Methods. Journal of Artificial Intelligence Research. 2025. vol. 82. pp. 2233–2278. DOI: 10.1613/jair.1.16665.
7. Prajapati M., Baliarsingh S.K., Dora C., Bhoi A., Hota J., Mohanty J.P. Detection of AI-generated text using large language model. 2024 International Conference on Emerging Systems and Intelligent Computing (ESIC). 2024. pp. 735–740. DOI: 10.1109/ESIC60604.2024.10481602.
8. Mitchell E., Lee Y., Khazatsky A., Manning C.D., Finn C. Detectgpt: Zero-shot machine-generated text detection using probability curvature. Proceedings of the 40th International Conference on Machine Learning (PMLR). 2023. pp. 24950–24962.
9. Lau H.T., Zubiaga A. Understanding the Effects of Human-written Paraphrases in LLM-generated Text Detection. arXiv preprint arXiv:2411.03806. 2024.
10. Wu J., Yang S., Zhan R., Yuan Y., Chao L.S., Wong D.F. A survey on LLM-generated text detection: Necessity, methods, and future directions. Computational Linguistics. 2025. pp. 275–338. DOI: 10.1162/coli_a_00549.
11. GPTZero. Available at: gptzero.me (accessed 15.05.2025).
12. Kavian A., Pourhashem Kallehbasti M.M., Kazemi S., Firouzi E., Ghafari M. LLM security guard for code. Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering. 2024. pp. 600–603. DOI: 10.1145/3661167.366126.
13. Wu L. Y., Segura-Bedmar I. AI-generated Text Detection with a GLTR-based Approach. arXiv preprint. arXiv:2502.12064. 2025.
14. OpenAI. Available at: openai.com (accessed 15.05.2025).
15. OriginalityAI. Available at: https://originality.ai/ (accessed 15.05.2025).
16. AI Detector & Content Checker By Copyleaks. Available at: https://copyleaks.com/ai-content-detector (accessed 15.05.2025).
17. Writer. AI content detector. Available at: https://writer.com/ai-content-detector/ (accessed 15.05.2025).
18. Tulchinskii E., et al. Intrinsic dimension estimation for robust detection of AI-generated texts. Advances in Neural Information Processing Systems. 2023. vol. 36. pp. 39257–39276.
19. Nikolaev K. Development of a Neural Network Model for Recognizing Russian-Language Generated Texts. 2024 IEEE Int. Multi-Conf. on Engineering, Computer and Information Sciences (SIBIRCON). IEEE, 2024. pp. 396–400. DOI: 10.1109/SIBIRCON63777.2024.10758447.
20. Gritsay G., Grabovoy A., Chekhovich Y. Open access dataset for machine-generated text detection in Russian. Mendeley Data. V2. 2023. DOI: 10.17632/4ynxfp3w53.2.
21. Shamardina T., et al. Findings of the the ruatd shared task 2022 on artificial text detection in Russian. arXiv preprint arXiv:2206.01583. 2022.
22. RuATD. Available at: https://github.com/dialogue-evaluation/RuATD (accessed 15.05.2025).
23. Skrylnikov S., Posokhov P., Makhnytkina O. Artificial text detection in Russian language: A BERT-based approach. Proc. Int. Conf. Dialogue. 2022. pp. 1–7.
24. Gritsai G., Voznyuk A., Grabovoy A., Chekhovich Y. Are AI Detectors Good Enough? A Survey on Quality of Datasets With Machine-Generated Texts. arXiv e-prints. 2024. arXiv:2410.14677.
25. Pan L., et al. MarkLLM: An Open-Source Toolkit for LLM Watermarking. Proc. of the 2024 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations. 2024. pp. 61–71. DOI: 10.18653/v1/2024.emnlp-demo.7.
26. Pham C.M., et al. TopicGPT: A Prompt-based Topic Modeling Framework. arXiv e-prints. 2023. arXiv:2311.01449.

27. Tong Z., Zhang H. A text mining research based on LDA topic modelling. International conference on computer science, engineering and information technology. 2016. pp. 201–210. DOI : 10.5121/csit.2016.60616.
28. Geroimenko V. Key Principles of Good Prompt Design. The Essential Guide to Prompt Engineering: Key Principles, Techniques, Challenges, and Security Risks. Cham: Springer Nature Switzerland. 2025. pp. 17–36.
29. Модели генерации текста. Available at: https://yandex.cloud/ru/docs/foundation-models/concepts/yandexgpt/models (accessed 15.05.2025).
30. Yandex GPT. Available at: https://ya.ru/ai/gpt (accessed 15.05.2025).
31. GiGaChat. Available at: https://giga.chat/ (accessed 15.05.2025).
32. DeepSeek. Available at: https://www.deepseek.com/ (accessed 15.05.2025).
33. Kuznetsov S. A. Bol'shoj tolkovyj slovar' russkogo jazyka [Large explanatory dictionary of the Russian language]. Shangwu Yinshuguan, 2020. 1481 p. (In Russ.).
34. KenLM. Available at: https://github.com/kpu/kenlm (accessed 15.05.2025).
35. Savkin M., Voznyuk A., Ignatov F., Korzanova A., Karpov D., Popov A., Konovalov V. DeepPavlov 1.0: Your Gateway to Advanced NLP Models Backed by Transformers and Transfer Learning. Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2024. pp. 465–474. DOI: 10.18653/v1/2024.emnlp-demo.47.
36. RuRoBERTa-large. Available at: https://huggingface.co/ai-forever/ruRoberta-large (accessed 15.05.2025).
37. Maloyan N., Nutfullin B., Ilyushin E. Dialog-22 ruatd generated text detection. arXiv preprint arXiv:2206.08029. 2022.
38. Gritsay G., Grabovoy A., Chekhovich Y. Automatic detection of machine generated texts: Need more tokens. 2022 Ivannikov Memorial Workshop (IVMEM). IEEE, 2022. pp. 20–26. DOI: 10.1109/IVMEM57067.2022.9983964.

Fedotova Anastasiia — Junior researcher, Scientific and engineering center «intelligent systems of trusted interaction», TUSUR. Research interests: information security, artificial intelligence, machine learning, deep learning, NLP. The number of publications — 20. afedotowaa@yandex.ru; 40, Lenin Ave., 634050, Tomsk, Russia; office phone: +7(923)444-4125.

Romanov Aleksandr — Ph.D., Associate professor of the department, Department of integrated information security of electronic computing systems, TUSUR. Research interests: information security, data mining, artificial intelligence, text processing. The number of publications — 76. alexx.romanov@gmail.com; 40, Lenin Ave., 634050, Tomsk, Russia; office phone: +7(923)444-4125.

Acknowledgements. This research was funded by the Ministry of Science and Higher Education of Russia, Government Order for 2023–2025, project No. FEWM-2023-0015 (TUSUR).