УДК 004.912:004.8
https://doi.org/10.37661/1816-0301-2026-23-3-76-91

Поступила в редакцию | Received 09.07.2026
Подписана в печать | Accepted 30.07.2026
Опубликована | Published 30.09.2026

Трехвекторный алгоритм обнаружения русскоязычных нейросетевых текстовых фрагментов на основе вероятностного анализа

К. С. Крез✉, Е. Н. Шнейдеров, И. П. Галяк, Д. Д. Ефименко
✉E-mail: karinakrez04@gmail.com

Белорусский государственный университет информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь

Аннотация

Цели. Целями работы являются разработка и программная реализация трехвекторного алгоритма обнаружения русскоязычных нейросетевых текстовых фрагментов на основе вероятностного анализа. Объект исследования – русскоязычные тексты, предмет – статистические признаки их происхождения. Особое внимание уделяется количественному сравнению предложенного подхода с базовыми одновекторными методами обнаружения – детектором на основе перплексии и методом рангового анализа GLTR.

Методы. Предложенный алгоритм объединяет три вектора признаков: предсказуемость токенов, долю редких лексем и информационную плотность текста, оцениваемую через коэффициент алгоритмического сжатия. В качестве вероятностного ядра использована авторегрессионная языковая модель rugpt3small. Реализация выполнена на языке Python с применением библиотек transformers, PyTorch, NLTK и zlib. Экспериментальная проверка проводилась на выборке из 100 смешанных документов (16 000 предложений) пяти тематических групп, нейросетевые фрагменты которых сгенерированы четырьмя языковыми моделями: DeepSeek, GPT, Qwen и YandexAI. Качество оценивалось по отклонению от эталонной доли нейросетевого текста и классификационным метрикам.

Результаты. После применения калибровочной поправки средний индекс по выборке составил 45,2 балла при эталонном уровне 50, а значения для всех четырех моделей находились в узком диапазоне 44,4–45,9 балла, что свидетельствует об устойчивости алгоритма к выбору генеративной модели. Отклонение среднего индекса от эталона составило 4,8 балла против 11,1 балла у метода на основе перплексии и 7,6 балла у GLTR; F1-мера на уровне предложений достигла 0,76 против 0,63 и 0,66 соответственно.

Заключение. Трехвекторный алгоритм подтвердил свою эффективность в задаче обнаружения нейросетевых фрагментов и может служить основой для инструментов проверки академических текстов и систем антиплагиата. Дальнейшее развитие исследования связано с расширением выборки, проверкой устойчивости к перефразированию и редактированию, а также проведением абляционного анализа.

Ключевые слова: нейросетевой текст, вероятностный анализ, энтропия, частотный анализ, язык программирования Python, плагиат

Для цитирования. Трехвекторный алгоритм обнаружения русскоязычных нейросетевых текстовых фрагментов на основе вероятностного анализа / К. С. Крез, Е. Н. Шнейдеров, И. П. Галяк, Д. Д. Ефименко // Информатика. – 2026. − Т. 23, № 3. – С. 76–91. – https://doi.org/10.37661/1816-0301-2026-23-3-76-91.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

A three-vector algorithm for detecting Russian-language neural-network-generated text fragments based on probabilistic analysis

Karyna S. Krez✉, Yevgeny N. Shneiderov, Ivan P. Galyak, Daniil D. Efimenko
✉E-mail: karinakrez04@gmail.com

Belarusian State University of Informatics and Radioelectronics,
st. P. Brovki, 6, Minsk, 220013, Belarus

Abstract

Objectives. The purpose of the work is to develop and programmatically implement a three-vector algorithm for detecting Russian-language neural network text fragments based on probabilistic analysis. The object of the study is Russian-language texts, the subject is statistical signs of their origin. Special attention is paid to the quantitative comparison of the proposed approach with the basic single-vector detection methods – the perplexy detector and the GLTR rank analysis method.

Methods. The proposed algorithm combines three feature vectors: token predictability, the proportion of rare lexemes, and the information density of the text, estimated via the algorithmic compression ratio. The author's regression language model rugpt3small is used as a probabilistic core. The implementation is made in Python using the transformers, PyTorch, NLTK, and zlib libraries. The experimental validation was conducted on a sample of 100 mixed documents (16,000 sentences) of five thematic groups, the neural network fragments of which were generated by four language models – DeepSeek, GPT, Qwen and YandexAI. The quality was assessed by the deviation from the reference proportion of the neural network text and by classification metrics.

Results. After applying the calibration adjustment, the average index for the sample was 45.2 points against a reference level of 50, and the values for all four models were in the narrow range of 44.4–45.9 points, which indicates the algorithm's stability to the choice of a generative model. The deviation of the average index from the standard was 4.8 points versus 11.1 points for the method based on perplexity and 7.6 points for GLTR; the F1 measure at the sentence level reached 0.76 versus 0.63 and 0.66, respectively.

Conclusion. The three-vector algorithm has proven its effectiveness in the task of detecting neural network fragments and can serve as the basis for tools for verifying academic texts and anti-plagiarism systems. Further development of the research is related to the expansion of the sample, testing the resistance to paraphrasing and editing, as well as conducting ablative analysis.

Keywords: neural-network text, probabilistic analysis, entropy, frequency analysis, Python programming language, plagiarism

For citation. Krez K. S., Shneiderov Y. N., Galyak I. P., Efimenko D. D. *A three-vector algorithm for detecting Russian-language neural-network-generated text fragments based on probabilistic analysis*. Informatika [*Informatics*], 2026, vol. 23, no. 3, pp. 76–91 (In Russ.). https://doi.org/10.37661/1816-0301-2026-23-3-76-91.

Conflict of interests. The authors declare of no conflict of interest.

Введение

В последние годы наблюдается стремительное развитие языковых моделей, способных создавать связные, грамматически корректные и стилистически однородные тексты. Появление и массовое распространение систем на основе языковых моделей, включая ChatGPT, GPT, LLaMA, DeepSeek, Qwen, GigaChat и др., привело к существенному изменению подходов к созданию текстов. Если ранее автоматическая генерация текста воспринималась преимущественно как вспомогательная технология, то сегодня нейросетевые модели способны формировать научные, публицистические, учебные тексты, которые во многих случаях трудно отличить от материалов, написанных человеком [1].

Одновременно с этим возникает проблема достоверного определения происхождения текста. Особенно она проявляется в образовательной и научной среде. Использование генеративных моделей при подготовке курсовых, дипломных, диссертационных и научных работ может нарушать принципы академической добросовестности, если вклад искусственного интеллекта (ИИ) не обозначен явно. В сфере научных публикаций наличие немаркированных нейросетевых фрагментов способно снижать доверие к результатам исследований. В средствах массовой информации и цифровой коммуникации подобные технологии могут использоваться для создания фейковых новостей, манипулятивных сообщений и иных форм дезинформации.

Существующие алгоритмы обнаружения нейросетевых текстов обладают рядом ограничений. Многие детекторы ориентированы преимущественно на англоязычные данные и демонстрируют нестабильные результаты при анализе русскоязычных материалов. Это связано со сложной морфологической системой русского языка и высокой вариативностью синтаксических конструкций. Кроме того, значительная часть существующих алгоритмов опирается на один основной признак, например перплексию, частотность *n*-грамм или ранговое распределение токенов. Такие модели могут быть уязвимы к перефразированию, ручному редактированию, изменению стиля, а также к использованию специальных запросов, направленных на «очеловечивание» текста [2].

Цель статьи заключается в разработке и программной реализации трехвекторного алгоритма обнаружения русскоязычных нейросетевых текстовых фрагментов на основе вероятностного анализа.

Теоретические основы обнаружения нейросетевых текстов

Под нейросетевым текстом в рамках данного исследования понимается текстовый фрагмент, полностью или частично созданный генеративной языковой моделью. Современные языковые модели строятся на архитектуре трансформеров и обучаются на больших массивах текстовых данных. Основной принцип их работы заключается в предсказании следующего токена на основе предыдущего контекста [3].

Главная особенность таких моделей состоит в том, что они не обладают собственным пониманием текста в человеческом смысле, а воспроизводят статистически вероятные языковые последовательности. В результате формируемый текст преимущественно опирается на устойчивые языковые конструкции и распространенные речевые паттерны. Это способствует обеспечению высокой грамматической корректности и стилистической согласованности. Вместе с тем зависимость от статистически наиболее вероятных решений может приводить к снижению лингвистической вариативности. На количественном уровне данная особенность выражается в повышенной предсказуемости текста, однородности его структуры, ограниченном использовании редких лексических единиц и высокой степени алгоритмической сжимаемости [4, 5].

Тексты, написанные человеком, как правило, обладают большей вариативностью. Человек может использовать редкие слова, авторские обороты, асимметричные синтаксические конструкции, непредсказуемые переходы между мыслями и индивидуальные стилистические особенности. Даже в академическом тексте, где стиль является более формализованным, сохраняются признаки человеческой языковой вариативности.

В основе обнаружения нейросетевых текстов лежит выявление статистических закономерностей и отклонений. Однако использования только одного признака недостаточно, поскольку современные языковые модели постоянно совершенствуются и все успешнее имитируют человеческую манеру письма.

Анализ существующих алгоритмов обнаружения ИИ-фрагментов

Среди распространенных алгоритмов и инструментов обнаружения нейросетевых текстов можно выделить две группы методов.

Первую группу составляют методы, основанные на перплексии. Перплексия показывает, насколько неожиданным является текст для языковой модели. Если текст имеет низкую перплексию, это означает, что его последовательность токенов хорошо предсказывается моделью. Поскольку генеративные системы часто создают статистически вероятные фразы, низкая перплексия может указывать на ИИ-происхождение текста. Однако данный подход имеет существенный недостаток: научные и учебные тексты также могут быть достаточно предсказуемыми из-за терминологической устойчивости, шаблонных конструкций и формального стиля изложения.

Вторая группа включает методы рангового анализа токенов, например Giant Language Model Test Room (GLTR). Подобные алгоритмы оценивают, насколько часто реальные слова текста входят в список наиболее вероятных продолжений, предложенных языковой моделью. Если значительная часть слов находится в верхней зоне вероятностного распределения, текст может быть классифицирован как нейросетевой. Недостатком данных методов является зависимость от жестких пороговых значений и выбранной языковой модели. При анализе русскоязычных академических текстов возможно большое количество ложноположительных срабатываний из-за терминологической специфики [6, 7].

Среди значимых зарубежных исследований следует отметить работу Gehrmann et al. [9], в которой предложен инструмент GLTR для анализа рангового распределения токенов в тексте, а также метод DetectGPT, основанный на оценке кривизны функции логарифмического правдоподобия [10]. Отдельное направление исследований образуют методы цифровой маркировки, предусматривающие внедрение водяных знаков непосредственно в процессе генерации текста [11].

Ограничения рассмотренных методов свидетельствуют о необходимости совместного анализа нескольких взаимодополняющих характеристик текста. В связи с этим в статье предложен трехвекторный алгоритм, объединяющий показатели предсказуемости токенов, доли редких лексем и информационной плотности текста. Для оценки его эффективности проведено прямое количественное сравнение с детектором на основе перплексии и методом GLTR. Все три метода исследовались на единой тестовой выборке с использованием одинаковой предварительной обработки и общей эталонной разметки.

Обоснование выбора признаков трехвекторного алгоритма

В основу разработанного алгоритма положен анализ трех групп признаков: вероятностной предсказуемости текста, характера распределения лексических единиц и информационной плотности. Выбор данного набора признаков обусловлен их способностью

выявлять аномалии на различных уровнях текстовой организации. Комплексное использование указанных показателей обеспечивает многомерный анализ текстового материала и позволяет фиксировать ключевые различия между машинно-сгенерированными текстами и текстами, созданными человеком.

Первый признак – предсказуемость – отражает степень соответствия текста наиболее вероятным языковым продолжениям. Нейросетевые модели, особенно при стандартных параметрах генерации, часто выбирают токены из верхней части вероятностного распределения. В ИИ-текстах доля слов, входящих в число наиболее вероятных вариантов, обычно выше.

Для человеческого текста характерна большая вариативность, поэтому доля токенов, попадающих в верхнюю зону вероятностного распределения, обычно ниже.

Второй признак связан с долей редких слов. Под редкими понимаются токены, которые языковая модель оценивает как низковероятные в конкретном контексте. Человеческие тексты чаще включают такие слова.

Нейросетевые тексты, напротив, часто характеризуются большей лексической монотонностью. Даже если они выглядят грамматически корректными, их словарный состав может быть более сглаженным и статистически безопасным. Низкая доля редких лексем может рассматриваться как один из признаков ИИ-генерации.

Третий признак основан на оценке алгоритмической сжимаемости текста. Если текст содержит много повторяющихся или структурно однотипных элементов, он лучше сжимается стандартными алгоритмами компрессии. Нейросетевые тексты нередко обладают более равномерной структурой, повторяемыми конструкциями и меньшей степенью информационной неоднородности. Это может приводить к более высокой сжимаемости.

Человеческий текст, особенно в научной или аналитической работе, может содержать больше локальных смысловых переходов, терминологических вставок и индивидуальных формулировок. В результате его информационная плотность может быть выше, а коэффициент сжатия – менее однозначным [8].

Предлагаемый алгоритм основан на анализе текста с помощью авторегрессионной языковой модели, используемой в качестве вероятностного ядра. В практической реализации применялась модель sberbank-ai/rugpt3small_based_on_gpt2, позволяющая получать распределение вероятностей для токенов русскоязычного текста.

В статье единицей анализа является предложение. Последовательные предложения объединяются в более крупные текстовые единицы, для обозначения которых далее используется единый термин «фрагмент».

Эталонная метка задается для каждого предложения. Эталонная доля S_e и выявленная доля S_a рассчитываются в единой символьной шкале. Эти величины определяются следующим образом:

$$S_e = \frac{\sum_{j \in A} l_j}{\sum_{j=1}^{n_п} l_j}, \quad S_a = \frac{\sum_{j \in B} l_j}{\sum_{j=1}^{n_п} l_j}, \tag{1}$$

где A – множество предложений с эталонной меткой ИИ; B – множество предложений, отнесенных к ИИ алгоритмом; l_j – длина j-го предложения в символах; $n_п$ – количество предложений в документе.

Равенство количества человеческих и сгенерированных предложений само по себе не гарантирует долю 50 % по символам, поэтому S_e вычисляется отдельно для каждого документа и используется при расчете Δ.

Для каждого фрагмента рассчитываются три независимых вектора:

1. Вектор предсказуемости отражает долю токенов, которые входят в топ-50 наиболее вероятных продолжений, предсказанных языковой моделью. Обозначим эту долю как R_{50}. Чем выше значение данного показателя, тем более предсказуемым является текст.

Для нормализации используется сигмоидальная функция

$$P = \frac{1}{1 + e^{-\alpha(R_{50} - 0{,}85)}}, \tag{2}$$

где P – значение вектора предсказуемости; R_{50} – доля токенов, входящих в топ-50 наиболее вероятных предсказаний; 0,85 – эмпирический порог; α – коэффициент крутизны сигмоидальной функции; e – основание натурального логарифма.

Если доля предсказуемых токенов превышает 0,85, оценка нейросетевого происхождения фрагмента возрастает.

2. Вектор монотонности оценивает долю редких лексем. Под редкими понимаются токены, ранг которых в вероятностном распределении превышает 1000. Обозначим долю таких токенов как R_r.

Для оценки используется инвертированная сигмоидальная функция

$$M = \frac{1}{1 + e^{\beta(R_r - 0{,}01)}}, \tag{3}$$

где M – значение вектора монотонности, R_r – доля редких токенов, 0,01 – эмпирический центр функции, β – коэффициент крутизны, e – основание натурального логарифма.

Чем больше редких слов содержит текст, тем ниже значение M, а следовательно, тем ниже оценка его нейросетевого происхождения.

3. Вектор информационной плотности основан на коэффициенте алгоритмического сжатия текста. Для его вычисления используется библиотека zlib. Коэффициент сжатия определяется как отношение объема сжатого текста к объему исходного текста:

$$R_C = \frac{V_C}{V_O}, \tag{4}$$

где R_C – коэффициент сжатия, V_C – объем текста после сжатия, V_O – исходный объем текста.

Для получения нормализованной оценки используется функция

$$C = \frac{K_L}{1 + e^{\gamma(R_C - 0{,}45)}}, \tag{5}$$

где C – значение вектора информационной плотности, R_C – коэффициент алгоритмического сжатия, 0,45 – эмпирический порог, γ – коэффициент крутизны, K_L – коэффициент штрафа за малую длину текста.

Коэффициент штрафа за длину текста можно задать следующим образом:

$$K_L = \min\left(1, \frac{N}{1000}\right), \tag{6}$$

где N – количество символов в анализируемом фрагменте текста. При длине фрагмента менее 1000 символов надежность итоговой оценки снижается, поскольку короткие тексты хуже поддаются устойчивому статистическому анализу.

Коэффициент сжатия zlib обладает достаточной информативностью преимущественно при анализе текстов большого объема, тогда как на коротких фрагментах его значения характеризуются высокой вариативностью. Для большинства отдельных предложений применяется штрафной коэффициент K_L, учитывающий недостаточную длину текста, вследствие чего значимость третьего вектора существенно снижается. Поэтому значение компонента C, рассчитанное для фрагментов объемом менее 1000 символов, не может рассматриваться как самостоятельный надежный признак происхождения текста. Для снижения влияния данного ограничения компонент C рассчитывался на объединенных последовательностях соседних фрагментов общей длиной не менее 1000 символов.

Итоговая оценка нейросетевого происхождения фрагмента рассчитывается как линейная комбинация трех векторов:

$$S_b = w_P \cdot P + w_M \cdot M + w_C \cdot C \tag{7}$$

при условии

$$w_P + w_M + w_C = 1, \tag{8}$$

где S_b – итоговая оценка нейросетевого происхождения фрагмента, P – вектор предсказуемости, M – вектор монотонности, C – вектор информационной плотности, w_P, w_M, w_C – весовые коэффициенты.

В программной реализации использовались следующие значения коэффициентов: $\alpha = 20$, $\beta = 300$, $\gamma = 15$, $w_P = 0{,}6$, $w_M = 0{,}3$, $w_C = 0{,}1$. Пороговые значения 0,85, 0,01 и 0,45, а также указанные коэффициенты определялись на отдельной калибровочной выборке, которая не пересекалась с основной тестовой выборкой из 100 документов.

Калибровочная выборка включала 40 документов, состоявших из 6400 предложений, половина из которых была сгенерирована с помощью ИИ. Документы относились к пяти тематическим группам и были равномерно распределены с учетом тематики и использованных генеративных моделей. Выборка разделялась на обучающую и валидационную части в соотношении 70/30. Значения параметров подбирались на обучающей части таким образом, чтобы минимизировать отклонение Δ, после чего полученные результаты проверялись на валидационной части.

Общая оценка нейросетевого происхождения определяется как средневзвешенное значение по всем фрагментам текста. В качестве веса используется длина соответствующего фрагмента в символах:

$$S_d = \frac{\sum_{i=1}^{n} S_i \cdot L_i}{\sum_{i=1}^{n} L_i}, \quad (9)$$

где S_d – итоговая оценка документа, S_i – оценка i-го фрагмента, L_i – длина i-го фрагмента, n – количество фрагментов.

Такой подход позволяет анализировать как полностью сгенерированные тексты, так и документы со смешанной структурой, где нейросетевые фрагменты могут быть встроены в авторский текст.

Практическая реализация алгоритма

Практическая реализация алгоритма выполнена на языке программирования Python. В качестве основных инструментов использовались следующие библиотеки и технологии: transformers – для работы с языковой моделью, PyTorch – для выполнения вычислений и получения выходных вероятностных распределений, zlib – для расчета коэффициента сжатия, NLTK – для предварительной обработки и разделения текста, стандартные средства Python – для обработки файлов, агрегации результатов и формирования итоговой оценки.

Обработка текста включает четыре этапа:

1. Выполняется предварительная подготовка текста: удаление лишних пробелов, нормализация строк и разделение текста на предложения. Далее предложения группируются во фрагменты, что позволяет анализировать длинные документы без превышения ограничений языковой модели. Текст разделялся на фрагменты длиной не более 500 символов. Для сохранения локального контекста соседние фрагменты перекрывались на 50 символов. Фрагменты объемом менее 100 символов исключались из анализа, а максимальная длина контекста языковой модели ограничивалась 1024 токенами.

2. Каждый фрагмент токенизируется и передается на вход авторегрессионной модели. Модель формирует вероятностное распределение токенов, на основе которого рассчитываются показатели предсказуемости и доли редких лексем.

3. Для каждого фрагмента вычисляется коэффициент сжатия с использованием библиотеки zlib. Это позволяет получить независимую оценку информационной плотности текста.

4. Выполняется расчет итоговой оценки нейросетевого происхождения для каждого фрагмента. После этого результаты агрегируются в итоговую оценку документа.

Важным преимуществом реализации алгоритма является использование открытых инструментов и библиотек, допускающих применение в исследовательских и прикладных программных системах.

Методология эксперимента

Экспериментальная оценка алгоритма проводилась на выборке из 100 смешанных документов. Каждый документ содержал 160 предложений: 80 написанных человеком и 80 сгенерированных четырьмя языковыми моделями – DeepSeek, GPT, Qwen и YandexAI – по 20 предложений от каждой модели.

Для обеспечения тематического разнообразия выборка была разделена на пять смысловых групп по 20 документов в каждой:

– курсовые работы по теме разработки программного средства;

– предложения из опубликованных произведений русских писателей, в том числе Л. Н. Толстого, М. А. Булгакова, А. П. Чехова, Ф. М. Достоевского, И. С. Тургенева, А. С. Пушкина и др.;

– курсовые работы по теме экономики электронного бизнеса;

– курсовые работы по теме международных отношений в туризме;

– научно-популярные статьи по информационным технологиям.

Нейросетевые фрагменты создавались с использованием одинакового по структуре запроса и с сохранением стиля соответствующей смысловой группы. Это обеспечивало сопоставимость условий внутри основного эксперимента.

Варианты перефразирования, ручного редактирования сгенерированного текста и использования запросов, направленных на имитацию естественного авторского стиля, в рамках настоящего эксперимента не исследовались. Все нейросетевые фрагменты создавались в стандартном режиме с применением унифицированного запроса. Поэтому устойчивость предложенного комплексного подхода к указанным способам модификации текста рассматривается как исследовательская гипотеза, требующая самостоятельной экспериментальной проверки. Для подтверждения основного положения исследования разработанный алгоритм был сопоставлен с двумя базовыми одновекторными методами на той же выборке из 100 документов. Первый базовый метод использовал перплексию, рассчитанную той же авторегрессионной моделью; второй метод реализовывал ранговый анализ GLTR. Все методы использовали одинаковую предварительную обработку и единую эталонную разметку.

Для каждого фрагмента определялась перплексия, а итоговое значение для документа рассчитывалось как средневзвешенное с учетом длины фрагментов. Фрагмент относился к сгенерированным с помощью ИИ, если значение его перплексии было ниже порога $\theta_P = 38{,}5$.

В методе GLTR вычислялась доля токенов, входящих в топ-50 наиболее вероятных вариантов. Если она превышала порог $\theta_G = 0{,}72$, фрагмент классифицировался как нейросетевой. Оба пороговых значения определялись на калибровочной выборке и не изменялись при обработке тестовых данных.

Итоговый индекс документа для каждого базового метода рассчитывался как доля символов во фрагментах, классифицированных как сгенерированные с помощью ИИ. Полученное значение переводилось в шкалу от 0 до 100. Сравнение методов проводилось на одних и тех же документах с использованием парных результатов.

В ходе эксперимента оценивались следующие показатели:

– доля текста, классифицированного как нейросетевой;

– отклонение полученного результата от эталонной доли нейросетевого текста;

– продолжительность анализа;

– пиковое потребление оперативной памяти;

– устойчивость алгоритма при обработке текстов, созданных различными генеративными моделями;

– доля правильных классификаций (Accuracy), точность (Precision), полнота (Recall) и F1-мера на уровне предложений;

– средняя абсолютная ошибка определения доли нейросетевого текста;

– 95%-ные доверительные интервалы (ДИ) разностей между результатами сравниваемых методов;

– результаты парного статистического теста с поправкой на множественные сравнения.

Отклонение от эталонной доли нейросетевого текста рассчитывалось по формуле

$$\Delta = | S_e - S_a |, \tag{10}$$

где Δ – абсолютное отклонение выявленной доли от эталона; S_e – фактическая эталонная доля символов в предложениях с меткой ИИ в конкретном документе; S_a – доля символов в предложениях, отнесенных алгоритмом к ИИ. Использование постоянной эталонной доли $S_e = 50$ % допустимо только при подтвержденном равенстве человеческой и ИИ-частей по числу символов.

Результаты экспериментальной оценки и показатели вычислительной эффективности алгоритма представлены в табл. 1.

Таблица 1

Детализация индекса нейросетевого происхождения по пяти смысловым группам и четырем генеративным моделям

Table 1

Neural-origin index by five thematic groups and four generative models

Модель *Model*	Среднее, баллы *Average, points*	SD	Диапазон, баллы *Range, points*	95 % ДИ, баллы *95 % CI, points*	*t, c*	RAM, *МБ*
1. Курсовые работы: разработка программного средства						
DeepSeek	45,26	1,68	41,87–48,05	44,47–46,05	3,44	0,3
GPT	44,69	2,08	40,61–49,10	43,72–45,66	3,16	0,4
Qwen	45,94	1,63	42,68–48,35	45,18–46,70	2,64	0,3
YandexAI	46,13	1,53	43,06–49,32	45,41–46,85	3,24	0,3
2. Произведения русских писателей						
DeepSeek	43,96	1,79	40,53–47,41	43,12–44,80	3,34	0,3
GPT	43,64	2,23	38,54–48,23	42,60–44,68	3,08	0,5
Qwen	44,69	1,76	41,83–48,01	43,87–45,51	2,56	0,3
YandexAI	45,23	1,65	41,90–48,68	44,46–46,00	3,16	0,3

Окончание табл. 1
End of table 1

Модель *Model*	Среднее, баллы *Average, points*	SD	Диапазон, баллы *Range, points*	95 % ДИ, баллы *95 % CI, points*	*t, c*	RAM, *МБ*
3. Курсовые работы: экономика электронного бизнеса						
DeepSeek	45,46	1,70	42,07–48,16	44,66–46,26	3,40	0,3
GPT	44,84	2,10	40,80–50,04	43,86–45,82	3,14	0,4
Qwen	46,04	1,65	42,80–48,47	45,27–46,81	2,62	0,3
YandexAI	46,28	1,55	43,22–49,84	45,55–47,01	3,22	0,3
4. Курсовые работы: международные отношения в туризме						
DeepSeek	45,11	1,72	41,69–47,93	44,31–45,91	3,38	0,3
GPT	44,54	2,15	39,83–49,42	43,53–45,55	3,12	0,5
Qwen	45,79	1,69	42,39–48,38	45,00–46,58	2,58	0,3
YandexAI	46,03	1,58	42,74–49,37	45,29–46,77	3,18	0,3
5. Научно-популярные статьи по информационным технологиям						
DeepSeek	45,01	1,68	41,76–47,87	44,22–45,80	3,34	0,3
GPT	44,49	2,09	40,11–49,20	43,51–45,47	3,10	0,4
Qwen	45,49	1,63	42,31–48,20	44,73–46,25	2,60	0,3
YandexAI	45,98	1,53	42,81–49,20	45,26–46,70	3,20	0,3
Средние значения						
DeepSeek	44,96	1,76	40,53–48,16	44,61–45,31	3,38	0,3
GPT	44,44	2,13	38,54–50,04	44,02–44,86	3,12	0,4
Qwen	45,59	1,71	41,83–48,47	45,25–45,93	2,60	0,3
YandexAI	45,93	1,58	41,90–49,84	45,62–46,24	3,20	0,3
Все модели	45,23	0,97	43,05–47,25	45,04–45,42	3,08	0,3

Данные табл. 1 позволяют сделать следующие выводы:

– значения калиброванного индекса для четырех генеративных моделей находятся в узком диапазоне от 44,44 до 45,93 балла. Межмодельный разброс составляет менее 1,5 балла, что свидетельствует о слабой зависимости результатов от используемой генеративной модели. При этом калиброванный индекс является условным показателем и не должен интерпретироваться как точность классификации или процентная доля нейросетевого текста;

– наибольшее среднее значение индекса получено для YandexAI – 45,93 балла, наименьшее для GPT – 44,44 балла. Для Qwen и DeepSeek данный показатель составил 45,59 и 44,96 балла соответственно;

– среднее значение индекса по всей выборке составило 45,23 балла, что на 4,8 балла ниже эталонного уровня, равного 50;

– 95%-ные доверительные интервалы моделей с близкими средними значениями частично пересекаются, что указывает на незначительные различия между ними;

– продолжительность анализа одной парной подвыборки составила от 2,56 до 3,44 с, а обработка полного документа заняла около 12 с. Дополнительное потребление оперативной памяти при анализе одного документа составило 0,3–0,5 МБ без учета памяти, занимаемой языковой моделью. Полученные показатели свидетельствуют о вычислительной эффективности алгоритма.

Результаты сравнительного анализа трехвекторного алгоритма и базовых одновекторных методов на одной и той же выборке представлены в табл. 2.

Таблица 2

Калиброванные индексы трехвекторного алгоритма и базовых одновекторных методов на выборке из 100 документов

Table 2

Calibrated indices of the three-vector algorithm and baseline single-vector methods on a Sample of 100 Documents

Метод *Method*	Среднее значение индекса, баллы (0–100) *Average index value, points (0–100)*	Разность с уровнем 50, баллы *Difference from level 50, points*	СКО, баллы *Standard deviation, points*	95 % ДИ индекса, баллы *95% CI of the index, points*	Время, с *Time, s*
Перплексия	38,87	11,13	2,84	38,31–39,43	11,6
GLTR	42,36	7,64	2,51	41,86–42,86	11,9
Трехвекторный	45,23	4,77	0,97	45,04–45,42	12,3

Данные табл. 2 позволяют сделать следующие выводы:

– средний калиброванный индекс трехвекторного алгоритма отклоняется от условного эталонного уровня на 4,8 балла. Для метода на основе перплексии отклонение составляет 11,1 балла, а для GLTR – 7,6 балла;

– разброс значений трехвекторного алгоритма в 2,6–2,9 раза ниже, чем у базовых методов;

– средняя разность ошибок между трехвекторным алгоритмом и методом на основе перплексии составляет 6,4 балла в пользу предлагаемого алгоритма (95%-ный ДИ: 5,8–7,0 балла);

– по сравнению с GLTR средняя разность ошибок составляет 2,9 балла в пользу трехвекторного алгоритма (95%-ный ДИ: 2,3–3,5 балла);

– средняя абсолютная ошибка определения доли нейросетевого текста составляет 5,1 балла для трехвекторного алгоритма, 11,2 балла для метода на основе перплексии и 7,9 балла для GLTR.

Классификационные показатели сравнительных методов трехвекторного алгоритма на уровне предложений представлены в табл. 3.

Таблица 3
Классификационные метрики сравниваемых методов на уровне предложений
Table 3
Sentence-level classification metrics of the compared methods

Метод *Method*	Accuracy	Precision	Recall	F1
Перплексия	0,65	0,66	0,60	0,63
GLTR	0,68	0,70	0,62	0,66
Трехвекторный	0,78	0,81	0,72	0,76

Данные табл. 3 показывают, что трехвекторный алгоритм превосходит базовые методы по всем рассматриваемым классификационным показателям. Значение F1-меры превышает результат метода на основе перплексии на 0,13, а метода GLTR – на 0,10. Преимущество алгоритма обусловлено одновременным повышением точности и полноты классификации.

Для описательной оценки значимости отдельных векторов проведен компонентный анализ. Средние значения компонентов для сгенерированных фрагментов составили $P = 0{,}52$, $M = 0{,}38$ и $C = 0{,}14$. С учетом установленных весовых коэффициентов этим значениям соответствует средний некалиброванный индекс около 44 баллов, тогда как для фрагментов, написанных человеком, он составляет примерно 11 баллов.

Относительно низкое значение вектора C обусловлено применением коэффициента K_L, который снижает оценку коротких фрагментов. Возникающее вследствие этого смещение частично компенсируется аддитивной калибровочной поправкой, о чем свидетельствует небольшое отклонение среднего калиброванного индекса от условного эталонного уровня. Обозначение K_L используется для коэффициента штрафа за недостаточную длину текста и не связано с дивергенцией Кульбака – Лейблера. Количественная оценка влияния каждого вектора посредством абляционного анализа не входила в задачи основного эксперимента и рассматривается как направление дальнейших исследований.

Заключение

В статье разработан и программно реализован трехвекторный алгоритм обнаружения русскоязычных нейросетевых текстовых фрагментов, объединяющий показатели предсказуемости токенов, доли редких лексем и информационной плотности текста. Экспериментальная проверка проводилась на текстах, сгенерированных моделями DeepSeek, GPT, Qwen и YandexAI.

На исследованной выборке отклонение среднего калиброванного индекса трехвекторного алгоритма от условного эталонного уровня составило 4,8 балла, метода на основе перплексии – 11,1 балла, а метода GLTR – 7,6 балла. Алгоритм также показал меньший разброс результатов. Значение F1-меры трехвекторного алгоритма составило 0,76, метода на основе перплексии – 0,63, а метода GLTR – 0,66. Полученные результаты свидетельствуют о преимуществе комбинированного подхода в исследованных условиях.

Средний калиброванный индекс по всей выборке составил 45,2 балла. Значения, полученные для моделей DeepSeek, GPT, Qwen и YandexAI, находились в узком диапазоне от 44,4 до 45,9 балла. Небольшой межмодельный разброс свидетельствует об устойчивости алгоритма, однако отклонение от эталонного уровня указывает на сохранение незначительного занижения результата. Компонентный анализ показал, что основное значение имеют векторы предсказуемости и доли редких лексем, тогда как значимость вектора информационной плотности на коротких фрагментах снижается.

Разработанный алгоритм может использоваться как основа для систем проверки академических работ и анализа происхождения текста. Дальнейшие исследования предполагают расширение выборки, проверку на независимых данных, изучение различных режимов генерации и редактирования, совершенствование показателя zlib, применение вероятностной калибровки и проведение абляционного анализа.

Вклад авторов. *К. С. Крез* подготовила данные и утвердила окончательную версию статьи. *Е. Н. Шнейдеров* внес значительный вклад в концепцию работы, научное руководство и редактуру содержания. *И. П. Галяк* разработал концепцию алгоритма и его программную реализацию. *Д. Д. Ефименко* осуществил сбор, анализ и оформление результатов эксперимента.

Список использованных источников

1. Ивахненко, Е. Н. ChatGPT в высшем образовании и науке: угроза или ценный ресурс? / Е. Н. Ивахненко, В. С. Никольский // Высшее образование в России. – 2023. – Т. 32, № 4. – С. 9–22.

2. Федотова, А. М. Методика идентификации текстов, сгенерированных большими языковыми моделями / А. М. Федотова, А. С. Романов // Информатика и автоматизация. – 2025. – Т. 24, № 5. – С. 1444–1470.

3. Мачковская, Л. Я. Генеративные модели искусственного интеллекта в преподавании русского языка как иностранного: возможности, ограничения и риски / Л. Я. Мачковская, А. В. Фатина, О. А. Ветошкина // Международный журнал гуманитарных и естественных наук. – 2025. – № 11-1. – С. 73–78.

4. Айдагулова, А. Р. Особенности текстов, сгенерированных искусственным интеллектом / А. Р. Айдагулова // Вестник Башкирского государственного педагогического университета им. М. Акмуллы. – 2023. – № 4 (72). – С. 154–156.

5. Брызгалина, Е. В. Вызовы технологий искусственного интеллекта для этической экспертизы исследований живых систем / Е. В. Брызгалина // Теоретическая и прикладная этика: традиции и перспективы : материалы XVI Междунар. конф., Санкт-Петербург, 16–18 нояб. 2023 г. – СПб. : Изд-во С.-Петерб. гос. ун-та, 2024. – С. 57–58.

6. Вейс, И. А. Распознавание паттернов применения искусственного интеллекта при создании текстов / И. А. Вейс // Современные инновации, системы и технологии. – 2025. – Т. 5, № 1. – С. 1033–1040.

7. Василевская, А. В. ИИ в количественном исследовании: проверка устойчивости к манипуляциям и искажениям фактов / А. В. Василевская // Теоретическая и прикладная этика: традиции и перспективы : материалы XVI Междунар. конф., Санкт-Петербург, 16–18 нояб. 2023 г. – СПб. : Изд-во С.-Петерб. гос. ун-та, 2024. – С. 67–68.

8. Никитина, А. С. Человек или AI: к вопросу об авторстве / А. С. Никитина // Вестник молодых ученых и специалистов Самарского университета. – 2024. – № 1 (24). – С. 182–186.

9. Gehrmann, S. GLTR: Statistical detection and visualization of generated text / S. Gehrmann, H. Strobelt, A. M. Rush // Proc. of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Florence, Italy, 28 July – 2 Aug. 2019. – Florence, 2019. – P. 111–116.

10. DetectGPT: Zero-shot machine-generated text detection using probability curvature / E. Mitchell, Y. Lee, A. Khazatsky [et al.] // Proc. of the 40th Intern. Conf. on Machine Learning (ICML 2023), Honolulu, Hawaii, USA, 23–29 July 2023. – Honolulu, 2023. – P. 24950–24962.

11. Watermark for large language models / J. Kirchenbauer, J. Geiping, Y. Wen [et al.] // Proc. of the 40th Intern. Conf. on Machine Learning (ICML 2023), Honolulu, Hawaii, USA, 23–29 July 2023. – Honolulu, 2023. – P. 17061–17084.

References

1. Ivakhnenko E. N., Nikolsky V. S. *ChatGPT in higher education and science: A threat or a valuable resource?* Vysshee obrazovanie v Rossii [*Higher Education in Russia*], 2023, vol. 32, no. 4, pp. 9–22 (In Russ.).

2. Fedotova A. M., Romanov A. S. *Methodology for identifying texts generated by large language models*. Informatika i avtomatizatsiya [*Computer Science and Automation*], 2025, vol. 24, no. 5, pp. 1444–1470 (In Russ.).

3. Machkovskaya L. Ya., Fatina A. V., Vetoshkina O. A. *Generative artificial intelligence models in teaching Russian as a foreign language: Opportunities, limitations, and risks*. Mezhdunarodnyy zhurnal gumanitarnykh i estestvennykh nauk [*International Journal of Humanities and Natural Sciences*], 2025, no. 11-1, pp. 73–78 (In Russ.).

4. Aydagulova A. R. *Features of texts generated by artificial intelligence*. Vestnik Bashkirskogo gosudarstvennogo pedagogicheskogo universiteta im. M. Akmully [*Bulletin of M. Akmulla Bashkir State Pedagogical University*], 2023, no. 4 (72), pp. 154–156 (In Russ.).

5. Bryzgalina E. V. *Challenges of artificial intelligence technologies for the ethical review of living systems research.* Teoreticheskaya i prikladnaya etika: traditsii i perspektivy: materialy XVI Mezhdunarodnoj konferencii, Sankt-Peterburg, 16–18 nojabrja 2023 g. [*Theoretical and Applied Ethics: Traditions and Prospects: Proceedings of the 16th International Conference, Saint Petersburg, 16–18 November 2023*]. Saint Petersburg, Izdatel'stvo Sankt-Peterburgskogo gosudarstvennogo universiteta, 2024, pp. 57–58 (In Russ.).

6. Veys I. A. *Recognition of artificial intelligence application patterns in text creation*. Sovremennye innovatsii, sistemy i tekhnologii [*Modern Innovations, Systems and Technologies*], 2025, vol. 5, no. 1, pp. 1033–1040 (In Russ.).

7. Vasilevskaya A. V. *AI in quantitative research: Testing resilience to manipulation and factual distortions*. Teoreticheskaya i prikladnaya etika: traditsii i perspektivy: materialy XVI Mezhdunarodnoj konferencii, Sankt-Peterburg, 16–18 nojabrja 2023 g. [*Theoretical and Applied Ethics: Traditions and Prospects: Proceedings of the 16th International Conference, Saint Petersburg, 16–18 November 2023*]. Saint Petersburg, Izdatel'stvo Sankt-Peterburgskogo gosudarstvennogo universiteta, 2024, pp. 67–68 (In Russ.).

8. Nikitina A. S. *Human or AI: On the issue of authorship*. Vestnik molodykh uchenykh i spetsialistov Samarskogo universiteta [*Bulletin of Young Scientists and Specialists of Samara University*], 2024, no. 1 (24), pp. 182–186 (In Russ.).

9. Gehrmann S., Strobelt H., Rush A. M. GLTR: Statistical detection and visualization of generated text. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Florence, Italy, 28 July – 2 August 2019.* Florence, 2019, pp. 111–116.

10. 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 (ICML 2023), Honolulu, Hawaii, USA, 23–29 July* 2023. Honolulu, 2023, pp. 24950–24962.

11. Kirchenbauer J., Geiping J., Wen Y., Katz J., Miers I., Goldstein T. A watermark for large language models. *Proceedings of the 40th International Conference on Machine Learning (ICML 2023), Honolulu, Hawaii, USA, 23–29 July* 2023. Honolulu, 2023, pp. 17061–17084.

Информация об авторах

Крез Карина Сергеевна, аспирант, ассистент кафедры проектирования информационно-компьютерных систем, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: karinakrez04@gmail.com

Шнейдеров Евгений Николаевич, кандидат технических наук, доцент, доцент кафедры проектирования информационно-компьютерных систем, проректор по учебной работе, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: shneiderov@bsuir.by

Галяк Иван Павлович, студент кафедры проектирования информационно-компьютерных систем, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: haliak.ivanba@gmail.com

Ефименко Даниил Дмитриевич, студент кафедры проектирования информационно-компьютерных систем, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: Dhmainer@gmail.com

Information about the authors

Karyna S. Krez, Postgraduate Student, Assistant of the Department of Information and Computer Systems Design, Belarusian State University of Informatics and Radioelectronics.
E-mail: karinakrez04@gmail.com

Yevgeny N. Shneiderov, Cand. Sci. (Eng.), Assoc. Prof., Assoc. Prof. of the Department of Information and Computer Systems Design, Vice-Rector for Academic Affairs, Belarusian State University of Informatics and Radioelectronics.
E-mail: shneiderov@bsuir.by

Ivan P. Galyak, Student at the Department of Information and Computer Systems Design, Belarusian State University of Informatics and Radioelectronics.
E-mail: haliak.ivanba@gmail.com

Daniil D. Efimenko, Student at the Department of Information and Computer Systems Design, Belarusian State University of Informatics and Radioelectronics.
E-mail: Dhmainer@gmail.com