Санкт-Петербургский государственный университет

Программная инженерия

Группа 24.М71-мм

Воспроизведение и анализ эффективности DetectGPT для русского текста

Ван Цзыхань

Отчёт по учебной практике
в форме «Сравнение»

Научный руководитель:
ст. преподаватель кафедры ИАС, к.ф.-м.н. Азимов Р. Ш.

Санкт-Петербург
2025

Оглавление

Введение

После значительного прорыва в области искусственного интеллекта, крупномасштабные модели, предварительно обученные на больших данных, быстро стали широко распространенными. 30 ноября 2022 года компания OpenAI выпустила чат-бота ChatGPT, основанного на мощной модели GPT 3.5 (Generative Pre-trained Transformer). В отличие от традиционных чат-ботов, ChatGPT завоевал мировое признание благодаря своей способности эффективно понимать контекст, генерировать текст и обладать обширными знаниями. Причина того, что ответы ChatGPT звучат так близко к человеческим, заключается в использовании метода обучения с подкреплением с обратной связью от человека (Reinforcement Learning with Human Feedback, RLHF) [1]. Это как основное преимущество GPT и подобных моделей, так и одна из самых спорных их характеристик.

Несмотря на то, что эти модели продемонстрировали свои мощные возможности в генерации текста, их широкое применение также вызвало множество обсуждений. Билевский Павел Геннадиевич [2] считает, что история искусственного интеллекта полна случаев, когда его потенциал чрезмерно преувеличивался, и такие завышенные оценки обычно использовались для привлечения инвестиций через агрессивный маркетинг. Билевский Павел Геннадиевич выделил несколько потенциальных угроз, связанных с GPT:1.Риски распространения ложной информации и «фейковых новостей»: он утверждает, что работа ChatGPT основана на заранее подготовленных текстовых данных, и ответы генерируются с помощью алгоритмов, при этом эти сгенерированные материалы не имеют указания на источники и не могут быть проверены на «фактическую достоверность».2.Проблемы с авторским правом и авторством: контент, сгенерированный ChatGPT, не имеет четкого автора, что может вызвать вопросы о праве собственности и моральной ответственности. Для ученых это также связано с возможностью академического мошенничества.3.Чрезмерная зависимость от технологий: излишняя зависимость от искусственного интеллекта и уменьшение роли человека могут при-

вести к тому, что люди не смогут сохранять ведущую роль в процессе технологического развития.

В отличие от предыдущих утверждений, Людмила Анатольевна Иванова [3] занимает более сбалансированную позицию. Она отмечает, что технологии, такие как GPT, обладают существенными преимуществами, в частности, в повышении эффективности написания текстов. Однако, по её мнению, использование искусственного интеллекта должно быть сопряжено с соблюдением строгих ограничений и стандартов, что позволит минимизировать возможные риски и обеспечить этическую корректность применяемых решений.

В настоящее время уже существует множество соответствующих технологий, таких как GPTZero [4], Originality.AI [5], DetectGPT [6], которые используют методы глубокого обучения для обнаружения искусственно сгенерированных текстов. В данной работе будет проведен анализ сильных и слабых сторон основных нейронных сетевых моделей, а также проведены улучшения для модели, ориентированной на русский язык, в области обнаружения, и в конечном итоге модель будет упакована в исполнимую программу.

1. Постановка задачи

Целью данного исследования является воспроизведение метода DetectGPT и проведение систематического сравнительного анализа его эффективности для обнаружения сгенерированных ИИ текстов на русском языке. Для достижения этой цели были поставлены следующие задачи, решаемые с использованием языка программирования Python:

1. Собрать и подготовить датасет русскоязычных текстов, включающий тексты, написанные человеком, и тексты, сгенерированные языковыми моделями.

2. Реализовать метод DetectGPT для русскоязычных текстов, адаптировав стратегии возмущения и оценки вероятности.

3. Провести абляционные эксперименты для анализа влияния различных компонентов метода (стратегии возмущения, количество возмущений, модель оценки) на итоговую производительность.

4. Сравнить эффективность DetectGPT с базовыми методами обнаружения, включая анализ перплексии и другие zero-shot подходы, проведя комплексный бенчмаркинг.

5. Проанализировать преимущества и ограничения метода DetectGPT в контексте русского языка и выработать рекомендации по его дальнейшему улучшению и оптимизации.

2. Обзор

Существующие методы обнаружения можно условно разделить на два типа:(1) Методы обнаружения на основе признаков(2) Методы обнаружения на основе глубокого обучения[7].Основное различие между ними заключается в способах извлечения признаков и возможностях модели.

2.1. Методы обнаружения на основе признаков

Первая группа методов обнаружения основана на извлечении и анализе различных статистических характеристик машинно-сгенерированных текстов для определения их источника (то есть, человек или машина). Эти характеристики обычно связаны с языковыми паттернами текста, синтаксической структурой, использованием лексики, генерационными закономерностями и т. д. Одним из существующих моделей является GLTR (Giant Language model Test Room) [8], который использует ряд статистических методов на основе признаков для обнаружения машинно-сгенерированных текстов. Конкретно, его ключевая технология включает три теста: вероятность генерации слов, их ранжирование и энтропия контекста. Эти характеристики выявляют особенности машинно-сгенерированных текстов — они обычно сосредоточены на словах с высокой вероятностью и слишком уверены в генерации слов в условиях низкой энтропии. Эксперименты показали, что GLTR значительно повышает точность распознавания фальшивых текстов пользователями, с 54% без использования инструмента до более 72% при использовании. Однако у GLTR есть свои ограничения, особенно при работе с враждебно сгенерированными текстами и новыми генеративными моделями, его эффективность может снизиться. Kristina Schaaff и другие [9] обобщили часто используемые текстовые признаки, которые включают восемь основных категорий, таких как сложность (Perplexity), семантика (Semantic), читаемость (Readability), текстовый вектор (Text Vector) и т. д. Заключения экспериментов показали, что текстовые векторы, извлечённые с использованием предобученной модели BERT, и

структурные характеристики документа играют решающую роль в обучении модели. Методы обнаружения на основе статистики текстовых признаков сильно зависят от выбора признаков, что требует высокого уровня лингвистических знаний от исследователей, что затрудняет их распространение. Это привело к развитию методов обнаружения на основе глубокого обучения.

2.2. Методы обнаружения на основе нейронных сетей

Методы обнаружения на основе глубокого обучения сначала требуют предварительной обработки текста, затем преобразуют текст в вектор слов, а затем с помощью глубоких нейронных сетей извлекают характеристики текста, а не вручную отбирают их, что значительно увеличивает допустимость ошибок. ZhiWu Fan и другие [10] исследовали традиционные методы, такие как TextCNN, TextRNN, а также активные в области обработки естественного языка модели, такие как Transformer и DPCNN, и обнаружили, что использование глубокой пирамидальной свертки (DPCNN) обеспечивает высокую точность на уровне 96,85%. Кроме того, Mitchell и другие [11] предложили модель для обнаружения без обучающих примеров — DetectGPT. DetectGPT — это метод обнаружения без обучающих примеров, основанный на кривизне вероятности, предназначенный для выявления текста, сгенерированного большими языковыми моделями (LLM). Он не требует дополнительных обучающих данных, отдельного классификатора или технологии водяных знаков текста, а использует анализ вероятностной структуры LLM и возмущений текста для выполнения обнаружения. Исследования показали, что текст, сгенерированный машиной, и текст, написанный человеком, имеют различия в кривизне распределения вероятности, особенно в области отрицательной кривизны. Тексты, написанные человеком, обычно не подвергаются значительным изменениям после добавления небольших возмущений, в то время как машинное обучение оказывает значительное влияние на изменения после возмущений. Результаты показали, что

DetectGPT всегда обеспечивал наивысшую точность AUROC в задаче обнаружения на нескольких наборах данных (таких как XSum, SQuAD и WritingPrompts) и на нескольких моделях (например, GPT-2, Neo-2.7, NeoX и др.), что еще раз подчеркивает преимущества глубокого обучения по сравнению с традиционными методами машинного обучения для поиска признаков.

Однако текущие модели недостаточно адаптированы для русского языка. Например, для моделей DetectGPT и GPTzero ниже мы покажем три изображения, иллюстрирующих реальные примеры.

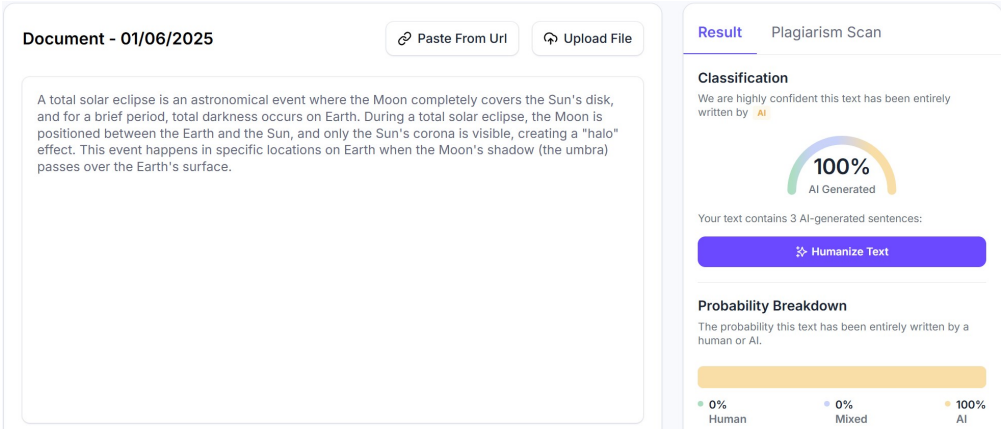

Рис. 1: Эффективность модели DetectGPT в контексте английского языка

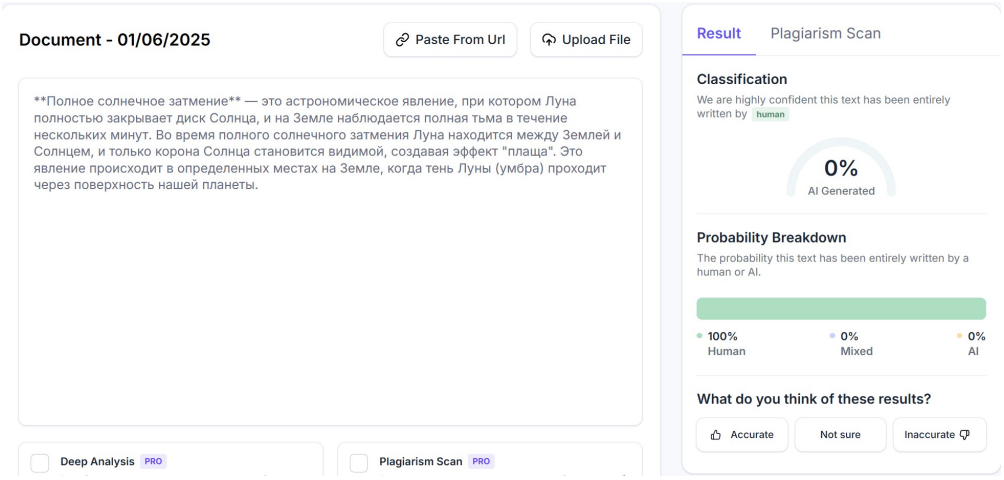

Рис. 2: Эффективность модели DetectGPT в контексте русского языка

На первом изображении модель успешно идентифицировала, что английский текст, сгенерированный GPT, был произведен искусственным интеллектом. Однако на втором изображении модель ошибочно

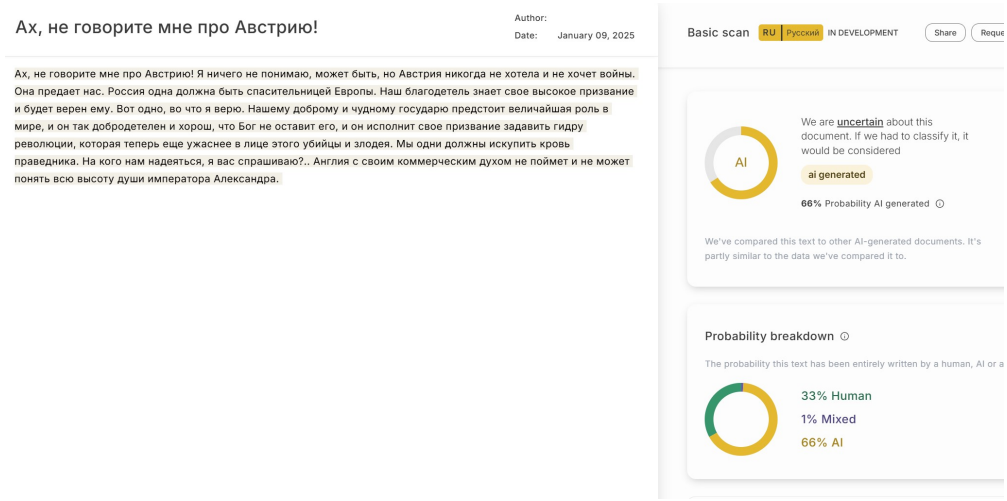

Рис. 3: Эффективность модели GPTzero в контексте русского языка

оценила вероятность того, что русский текст был сгенерирован ИИ, как 0%, что является недостоверным выводом. На третьем изображении, при использовании модели GPTzero для анализа фрагмента из ”Войны и мира”, система оценила вероятность того, что этот текст был сгенерирован ИИ, на уровне 66%, что также представляет собой ошибочную интерпретацию.

3. Описание решения

3.1. Сбор данных

Сбор данных является ключевым этапом исследовательского процесса, качество которого напрямую определяет надежность и достоверность экспериментальных результатов. В рамках текстового анализа данные будут извлекаться из различных источников: на первом этапе планируется использовать существующие наборы вопросов и ответов, например диалоговые материалы из мессенджеров; далее источники будут расширены за счёт авторитетных новостных изданий, включая телеканалы и их цифровые платформы.

Получение машинно-сгенерированного текста осуществляется в основном путём разработки подсказок и прямой массовой генерации текстовых данных с использованием локальной модели.

3.2. Стратегия генерации возмущений

Для изучения влияния возмущённых текстов на распределение предсказаний больших языковых моделей в этом исследовании используется модель T5-3B для генерации возмущённых текстов. Причина выбора T5[12] заключается в том, что метод DetectGPT требует создания текстов, сохраняющих исходный смысл, но с небольшими различиями, чтобы проверить стабильность вероятностной кривой исходного текста. Иными словами, T5 помогает нам создавать «легкие возмущения», которые не изменяют смысл предложения, но позволяют моделировать чувствительность модели к небольшим изменениям.

Пусть исходный текст обозначен как $X = [x_1, x_2, \ldots, x_n]$, где x_i — i-й токен. Процесс генерации возмущений описывается следующим образом:

1. Случайным образом выбираются k токенов для маскирования, где $k \in \{1, 2, 3\}$ и составляет примерно 30% длины текста.

2. Маскированный текст X_{masked} подается в модель T5-3B для генерации предсказанных токенов, в результате чего получается

возмущённый текст $X'_j = [x'_1, x'_2, \ldots, x'_n]$.

3. Для каждого исходного текста повторно генерируются $m = 5$ различных версий возмущений, образуя множество возмущённых текстов $\{X'_1, X'_2, \ldots, X'_5\}$.

Формально это можно записать как:

$$X'_j = \text{T5-3B}(\text{Mask}(X, k)), \quad j = 1, 2, \ldots, m$$

Анализируя изменения распределения вероятностей множества возмущённых текстов $\{X'_j\}$, можно исследовать чувствительность модели к незначительным изменениям текста. Это обеспечивает теоретическую основу для метода DetectGPT, основанного на кривизне вероятностного распределения.

3.3. Стратегия вычисления Log-Likelihood

Для текста вычисляется log-likelihood с использованием авторегрессионной модели Suzume-llama-3-8B. Методика вычисления следующая:

- Берётся логарифм вероятности каждого токена в тексте.
- Вычисляется среднее log-likelihood по всем токенам и берётся отрицательное значение.
- Среднее значение нормируется по длине текста T.
- Вычисление осуществляется с использованием функции потерь перекрёстной энтропии (cross-entropy loss) в режиме no_grad для ускорения расчётов.

Пусть текст обозначен как $X = [x_1, x_2, \ldots, x_T]$, где x_t — t-й токен, и вероятность каждого токена вычисляется на основе всех предыдущих токенов. Формула среднего log-likelihood:

$$\text{LL}(X) = -\frac{1}{T} \sum_{t=1}^{T} \log p(x_t \mid x_1, \ldots, x_{t-1})$$

Этот расчёт позволяет получить среднее log-likelihood текста по модели. Чем выше значение, тем лучше текст соответствует распределению языковой модели.

3.4. DetectGPT Score

Score DetectGPT используется для оценки того, является ли текст машинно-сгенерированным. Рассчитывается следующим образом:

1. Вычисляется среднее логарифмическое правдоподобие исходного текста x, обозначается как $\mathrm{LL}(x)$.

2. Для N версий текста с возмущениями вычисляется среднее логарифмическое правдоподобие, после чего берётся среднее этих значений:
$$\overline{\mathrm{LL}}_{\mathrm{perturb}} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{LL}(\tilde{x}_i)$$

3. Рассчитывается Score:
$$\mathrm{Score}(x) = \mathrm{LL}(x) - \overline{\mathrm{LL}}_{\mathrm{perturb}}$$

Объяснение: - Если текст сильно «теряет» вероятность при небольших изменениях (исходный текст выглядит естественнее, чем возмущённые версии), Score будет высоким — это часто означает, что текст создан ИИ. - Если изменения почти не влияют на вероятность текста, Score будет низким, что указывает на то, что текст, вероятно, написан человеком.

3.5. Влияние параметров генерации возмущений

В рамках эксперимента по оценке DetectGPT была проведена серия тестов для изучения влияния параметров генерации возмущённых текстов на процесс вычисления Score. В качестве ключевого параметра

рассматривалась доля маскирования, определяющая, какая часть токенов в исходном тексте случайным образом заменяется на маску при генерации возмущённых версий.

Для каждого значения доли маски генерировались соответствующие версии текстов с возмущениями с использованием модели T5-3B. Дополнительно проводилось сравнение стратегий маскирования: замаскированные позиции заполнялись случайными словами из русского словаря для проверки альтернативного подхода к генерации возмущённых текстов.

3.6. Базовая проверка с RoBERT

В качестве базового метода мы использовали предобученную модель RoBERT для задачи классификации текстов. Основная часть сети RuBERT была зафиксирована, обучалась только верхняя классификационная голова, чтобы избежать переобучения на небольшом наборе данных и сохранить стабильность предобученных языковых знаний[13].

Для классификации, при наличии векторного представления текста $h = \text{RoBERT}([CLS])$, верхняя голова представляет собой полносвязный слой W и softmax-функцию для вычисления вероятности каждого класса:

$$\hat{y} = \text{softmax}(Wh + b)$$

где $\hat{y}$ — вектор прогнозируемых вероятностей, b — вектор смещений. Цель обучения — минимизация функции перекрёстной энтропии:

$$\mathcal{L} = -\sum_{c=1}^{C} y_c \log \hat{y}_c$$

где C — количество классов, а y_c — one-hot кодировка истинной метки.

4. Эксперимент

4.1. Анализ применимости DetectGPT

Для оценки эффективности предложенного метода мы воспроизвели оптимальные параметры, использованные в оригинальной статье DetectGPT, и вычислили результаты детекции. Результаты представлены на рисунке 4. Как видно, DetectGPT способен различать большинство AI-текстов, однако сохраняется определённый уровень ошибок.

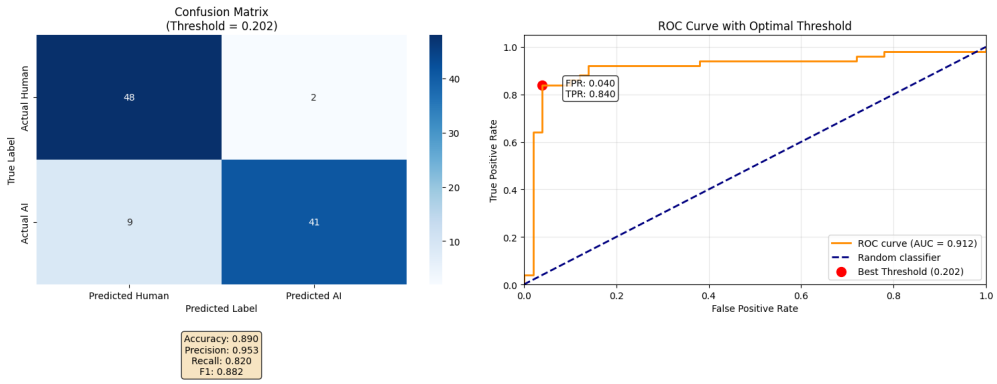

Рис. 4: Эффективность модели DetectGPT в контексте английского языка

4.2. Эксперимент с различными стратегиями маскировки

После вычисления баллов DetectGPT и оценки общей эффективности мы изучили влияние параметров генерации текстов с шумом на производительность детекции. Одним из ключевых параметров является доля маскирования, то есть какая часть токенов исходного текста случайным образом маскируется для генерации вариаций.

Эксперименты с разными долями маски показали, что при слишком высокой доле текста теряется смысл, и языковой модели сложно сохранить исходное содержание; при слишком низкой доле искажение недостаточно, и способность различать AI-тексты и тексты людей снижается. Результаты представлены на рисунке 5.

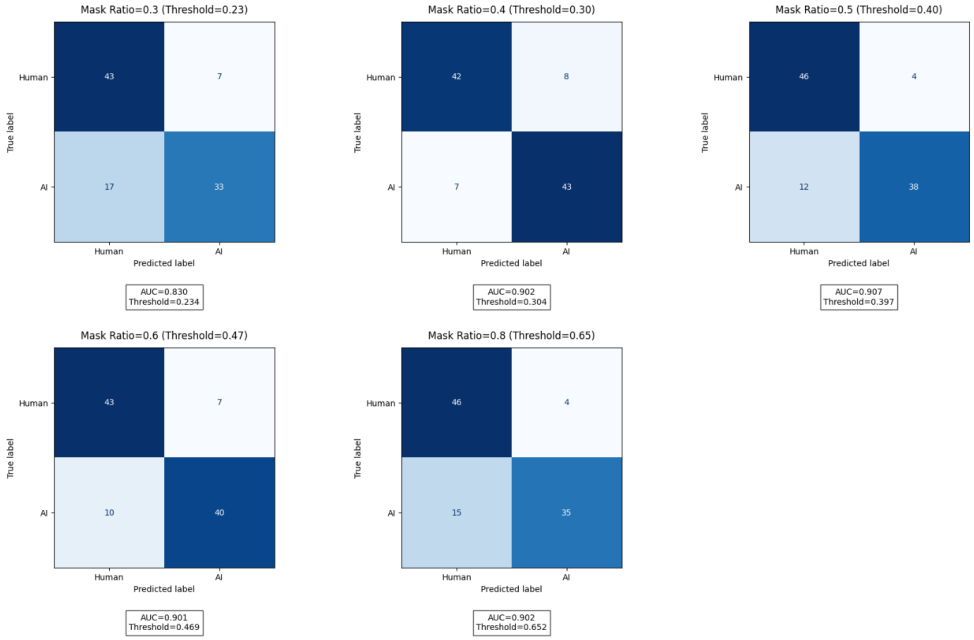

Рис. 5: Влияние доли маскирования на эффективность DetectGPT

Кроме того, по мере увеличения доли маскирования порог также растёт. Это показывает, что слишком большое соотношение ослабляет эффект малых изменений текста, и тогда сравнивается уже не разница между исходным и модифицированным текстом, а сам log-likelihood исходного текста.Поэтому даже при высоком значении AUC это уже не соответствует предположениям метода DetectGPT.

После выбора подходящей доли маскирования мы сравнили различные стратегии замены маски. Замаскированные позиции заполнялись случайными словами из русского словаря. Эксперименты показали, что стратегия T5 более стабильна и лучше сохраняет смысл исходного текста, повышая точность детекции DetectGPT (рисунок 6).

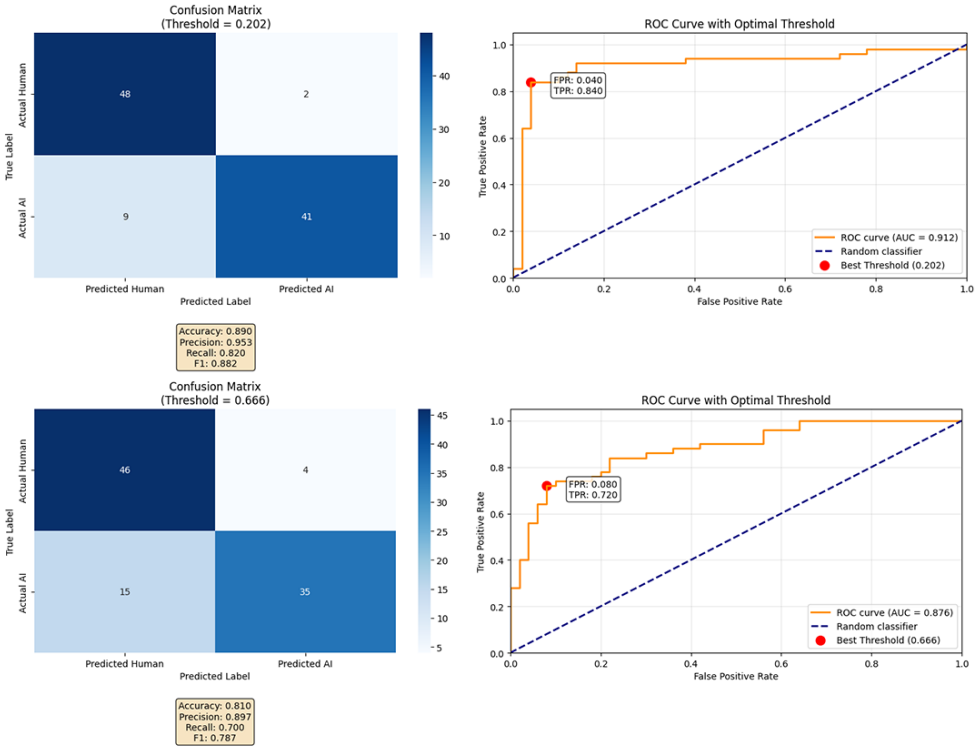

Рис. 6: Сравнение стратегий маскирования

4.3. Сравнение с базовыми методами

Наконец, мы сравнили эффективность DetectGPT с классификатором RoBERTa. В эксперименте основная часть сети RoBERTa была зафиксирована, обучалась только верхняя классификационная голова, что позволило избежать переобучения на небольшом наборе данных, сохранив предобученные знания. Результаты сравнения представлены на рисунке 7.

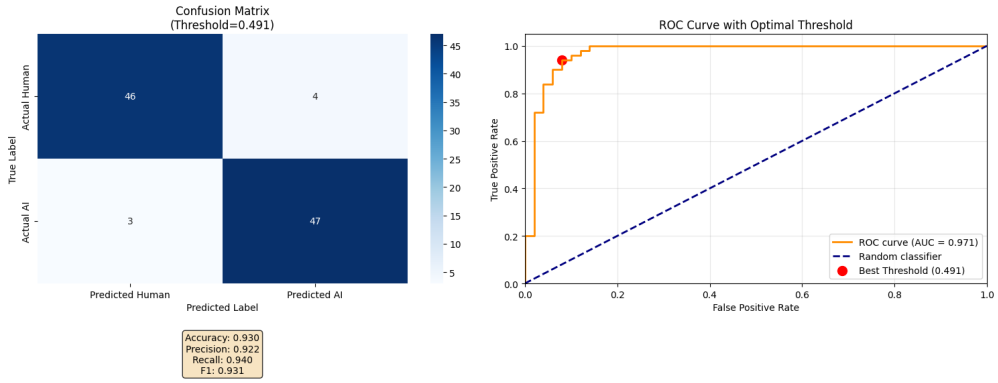

Рис. 7: Сравнение эффективности DetectGPT и классификатора RoBERTa

Сравнение показало, что хотя DetectGPT способен в определённой степени различать AI-тексты и тексты людей, RoBERTa при прямом дообучении демонстрирует более высокую классификационную эффективность. Это подтверждает преимущество методов глубокого обучения в задачах детекции.

Заключение

Основные выводы и направления развития модели:

- **Экспериментальное резюме:**

  1. DetectGPT позволяет быстро оценивать тексты без дополнительного обучения и помогает понять вероятностные характеристики генерации текста моделью.
  2. В русскоязычном контексте эффективность DetectGPT оказалась ограниченной: добавление шумовых вариаций может искажать оценку, а результаты зависят от способа маскирования и длины текста.
  3. DetectGPT может использоваться как вспомогательный инструмент для анализа характеристик модели, но из-за быстрого развития моделей вроде LLaMA его результаты не подходят в качестве стабильного эталона для детекции.
  4. Для более надёжного и точного распознавания текстов на русском языке рекомендуется использовать классификаторы на основе глубокого обучения, например RoBERTa.

- **Направления развития:**

  1. **Во-первых, оптимизация стратегии маскирования.** Необходимо учитывать особенности русского языка, такие как морфология и синтаксис, чтобы возмущения текста были более естественными и стабильными.
  2. **Во-вторых, интеграция с глубокими классификаторами.** DetectGPT-score можно использовать как признак в двухэтапном классификаторе: сначала оценка текста через DetectGPT, а затем более точная классификация с помощью глубоких моделей, например RoBERTa.
  3. **В-третьих, расширение набора данных.** Добавление текстов разных типов и разной длины и использование модели с

лучшей способностью к обобщению позволит вычислять log-likelihood точнее и выявлять даже незначительные различия.

Список литературы

[1] Ouyang, L., Wu, J., Jiang, X., et al. (2022). Training language models to follow instructions with human feedback [EB]. arXiv:2203.02155.

[2] Былевский, П. Г. (2023). Культурологическая деконструкция социально-культурных угроз ChatGPT информационной безопасности российских граждан. Философия и культура, 8.

[3] Иванова, Л. А. (доктор педагогических наук, доцент). (2024). Московский государственный технический университет гражданской авиации (Иркутский филиал). Коммунаров, 3, Иркутск, 664047, Россия.

[4] "GPTZero," [Online]. Available: https://en.wikipedia.org/wiki/GPTZero

[5] Originality.AI, [Online]. Available: https://originality.ai/

[6] "DetectGPT," [Online]. Available: https://detectgpt.com/

[7] Crothers, E. N., Japkowicz, N., & Viktor, H. L. (2023). Machine-generated text: A comprehensive survey of threat models and detection methods. IEEE Access, 11, 70977-71002.

[8] Gehrmann, S., Strobelt, H., & Rush, A. M., "GLTR: Statistical detection and visualization of generated text," arXiv preprint arXiv:1906.04043, 2019. [Online].

[9] K. Schaaff, T. Schlippe, and L. Mindner, "Classification of human and AI-generated texts for English, French, German, and Spanish," Proc. Int. Conf. on Natural Language and Speech Processing, 2023, pp. 1-10.

[10] Zhiwu, F., & Jinliang, Y. (2024). "Text detection method for ChatGPT-generated text based on deep pyramid convolutional neural network," Data Analysis and Knowledge Discovery, 8(7),

[11] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, and C. Finn, "DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature," arXiv preprint, arXiv:2301.11305, 2023.

[12] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer [EB]. arXiv:1910.10683. https://doi.org/10.48550/arXiv.1910.10683

[13] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding [EB]. arXiv:1810.04805. https://doi.org/10.48550/arXiv.1810.04805