Аннотация

В данной работе исследуется способ уменьшения размерности пространства обучаемых параметров в задаче детектирования ai текстов(задача многоклассовой классификации). Для fine tuning использовалась модель RoBerta с LoRA адаптером. Было проведено несколько экспериментов, чтобы выяснить, является ли использование LoRA для аппроксимации матрицы весов эффективным с точки зрения времени, ресурсов или точности. Было показано, что при меньших ресурсах модель distilled RoBerta base с LoRA адаптером может получить те же показатели метрик для классификации текстов, написанных человеком, что и vanilla distilled RoBerta base на наборе данных с 4 классами.

Ключевые слова: машинное обучение; линейная алгебра; аппроксимация матриц; уменьшение размерности пространств; классификация AI текстов; многоклассовая классификация текстов; большие языковые модели.

Содержание

1 Введение **4**

2 Постановка задачи классификации текстов **7**

3 Предложенный метод и его корректность **9**
3.1 Теорема 1 [12] 9
3.2 Теорема 2 12

4 Вычислительный эксперимент **15**
4.1 Данные 15
4.2 Предобработка 15
4.3 Эксперименты 15
4.4 Эксперимент (1) 16
4.5 Эксперимент (2) 17
4.6 Эксперимент (3) 18

5 Заключение **21**

1 Введение

Актуальность Уменьшение размерности пространства обучаемых параметров в задаче адаптации к домену упрощает процесс обучения и улучшает вычислительную эффективность. Путем сокращения количества параметров, которые необходимо обновить во время обучения, модель может потенциально быстрее сходиться и затрачивать меньше вычислительных ресурсов; это связано с структурой нейронной сети - ее сложность и потребление ресурсов зависят от количества обучаемых параметров. Уменьшение размерности может быть особенно важным в сценариях адаптации области, где обрабатываются большие объемы данных и происходит обучение с большим числом параметров.

Анализ методов решения задачи понижения размерности пространства обучаемых параметров Методы, направленные на решение проблемы снижения размерности: метод главных компонент [23] и его адаптации: тензорное разложение [15], [17], каноническое полиадическое разложение [27] выбирают наиболее важные векторы признаков из набора данных, используя сингулярное разложение матрицы для нахождения первых K собственных векторов с наибольшим собственным значением. Методы, осуществляющие отбор признаков: регуляризация LASSO (L1) [6], оценка Фишера [7] или Тест Хи-квадрат [28]. Метод снижения размерности, основанный на дообучении больших текстовых моделей - дистилляция [10]; в этом методе большая генеративная модель является «учителем», а меньшая - «учеником». Модель «ученика» обучают с использованием прогнозов "учителя". Эти идеи были впервые представлены в работах Дж. Хинтона [9] и В.Н. Вапника [20].

Метод, рассмотренный в данной работе - низкоранговое разложение(англ. Low Rank Adaptation) [11], который разработан на основе идеи о том, что предварительно обученные языковые модели имеют низкую "внутреннюю размерность"и могут эффективно обучаться, несмотря на проецирование на меньшее подпространство [2]. Данный метод, как и метод главных компонент, использует сингулярное

разложение матрицы для нахождения низкоранговых приближений матрицы весов.

Здесь, метод LoRA применяется к одной из наиболее ресурсозатратных задач: обнаружение текстов, написанных искуственным интеллектом(или человеком). Проблема обнаружения текстов, написанных искусственным интеллектом, не теряет популярности с годами в научном сообществе. Особенно сейчас, когда отличить тексты, написанные человеком, от текстов, написанных искусственным интеллектом, является основной проблемой для агентств по борьбе с плагиатом. В данной статье мы работали над fine tuning'ом популярной LLM RoBerta с использованием LoRA адаптеров с целью понижения размерности пространства обучаемых параметров. Предполагается, что LoRA может быть так же эффективен в решении задач классификации, как и в задачах генерации: LoRA доказала свою эффективность во многих задачах генерации [5], [11], [4].

Анализ методов решения задачи классификации ai текстов Как отмечено в работах [8] и [1], все подходы к решению задачи классификации ai текстов, разделяются на два типа: ориентированные на анализ признакового пространства и ориентированные на дообучение моделей.

1) **Анализ признакового пространства** основывается на извлечении и анализе характеристик текста - лексических, синтаксических, семантических или стилистических характеристик: [13], [26], [25].

2) **Дообучение больших языковых моделей** основывается на изучении параметров и возможностей модели и последующем дообучении модели на данных к задаче классификации текстов: [24], [24], [16].

Цели Исследование снижения пространства обучаемых параметров при помощи разложения матриц.

Методы Низкоранговое разложение(англ. Low Rank Adaptation) матриц параметров в больших языковых моделях.

Теоретическая значимость В работе проведен теоретический анализ проблемы снижения размерности пространства обучаемых параметров. Доказана теорема об применимости модели BERT [21] с адаптером LoRA к задаче многоклассовой классификации.

Практическая значимость Проведен вычислительный эксперимент, показывающий улучшение качества и экономию ресурсов при решении задачи классификации текстов.

2 Постановка задачи классификации текстов

Модель семейства трансформеров решает проблему генерации последовательность в последовательность(англ. seq2seq), которая формулируется следующим образом согласно [19]:

$$f_\theta : \mathbb{R}^{n \times d} \to \mathbb{R}^{n \times d} \tag{2.1}$$

где модель f_θ отображает неупорядоченный набор из n элементов в $\mathbb{R}^d$ в другой неупорядоченный набор из n элементов в $\mathbb{R}^d$. Требуется переформулировать задачу в классификатор текстов, определяя, были ли они написаны человеком или одной из трех предложенных языковых моделей.

$$f_\theta : \hat{V} \to [N_c] \tag{2.2}$$

где $\hat{V} \subset V$; V — словарь токенов и V — его замыкание или множество всех текстов над V, $[N_c]$ - множество классов. Таким образом, модель отображает текст из $\hat{V}$ в класс из $[N_c]$.

Тогда $(X_i, c_i) \in V^* \times [N_c]$ для $i \in [N_{data}]$ является парой текст-класс, выбранной из $P(X, c)$, где X_i - входной текст, а c_i — его класс. Таким образом, наша цель - оценить $P(c|X)$.

Согласно [11] при дообучении модель инициализируется предварительно обученными весами Φ_0 и обновляется до $\Phi_0 + \delta\Phi$. Тогда задача минимизации функции потерь имеет вид:

$$\begin{aligned} \min_{\Phi} - \sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log\left(P_\Phi\left(c_i \mid X_i\right)\right) = \\ = \max_{\Phi} \sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log\left(P_\Phi\left(c_i \mid X_i\right)\right) \end{aligned} \tag{2.3}$$

Здесь $\mid \Delta\Phi \mid = \mid \Phi_0 \mid$.
В то время как при использовании LoRA: $\Delta\Phi = \Delta\Phi(\Theta)$, где Θ -

набор параметров намного меньшего размера, $\mid \Theta \mid \ll \mid \Phi_0 \mid$ и задача минимизации функции потерь имеет вид:

$$\begin{aligned} &\min_{\Theta} - \sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log \left(P_{\Phi_0 + \Delta\Phi(\Theta)} \left(c_i \mid X_i \right) \right) = \\ &= \max_{\Theta} \sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log \left(P_{\Phi_0 + \Delta\Phi(\Theta)} \left(c_i \mid X_i \right) \right) \end{aligned} \tag{2.4}$$

3 Предложенный метод и его корректность

Опираясь на оригинальную статью LoRA [11], в этой работе LoRA была применина к задаче классификации. Структура LoRA адаптера:

Fine tuning	LoRA fine tuning
$W_{upd} = W + \Delta W$	$W_{upd} = W + AB$
$\hat{y} = xW_{upd} = x(W + \Delta W)$	$\hat{y} = xW_{upd} = x(W + AB)$
$\hat{y} = xW + x\Delta W$	$\hat{y} = xW + xAB$

Где $W \in \mathbb{R}^{d\times k}$ - предобученные веса, $\Delta W \in \mathbb{R}^{d\times k}$ - матрица обновленных весов. ΔW is приближается с помощью метода LoRA произведением $A \cdot B$, где $A \in \mathbb{R}^{d\times r}$, $B \in \mathbb{R}^{r\times k}$ и r - гиперпараметр ранга. Здесь $A \sim \mathcal{N}(0,\, \sigma^2)$ и $B = [0]_{r\times k}$.

Докажем состоятельность предложенной модели
Сходимость традиционной модели трансформер была доказана в работе [12]. Доказательство приведено для задачи классификации:

3.1 Теорема 1 [12]

Теорема 1. *Будем считать, что:*

- *Существует модель с набором параметров θ^*, которая может аппроксимировать достоверное распределение, сгенерированное данными, с минимальным расхождением по KL-дивергенции:*

$$\exists\theta^* : \theta^* = \arg\min_{\theta} D_{KL}(P_{true} \,||\, P_{model}(\cdot, \theta)) \tag{3.1}$$

- *По мере увеличения размера набора данных $\hat{V}$ эмпирическое распределение приближается к истинному распределению, генерирующему данные.*

- *$\mathscr{L}(\theta)$ - непрерывная, дифференцируемая. Где*

$$\mathscr{L}(\theta) = -\sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log\left(P_{\Phi_0 + \Delta\Phi(\theta)}\left(c_i \mid X_i\right)\right) \tag{3.2}$$

Тогда минимизация $\mathscr{L}(\theta)$ приводит к получению оценки истинного распределения, генерирующего данные.

Доказательство. Докажем сходимость следующей функции:

Согласно минимизации функции эмпирического риска, минимум $\mathscr{L}(\theta)$ приближается к минимуму матожидания риска, поскольку размер $\hat{V}$ стремится к бесконечности.

$$\begin{gathered} \lim_{|\hat{V}| \to \infty} \arg\min_{\theta} \mathbb{E}\, \mathscr{L}(\theta) = \\ \arg\min_{\theta} \mathbb{E}_{X_i \in P_{true}}[\sum_{c_i \in [N_c]} \log\left(P_{\Phi_0 + \Delta\Phi(\theta)}\left(c_i \mid X_i\right)\right] = \\ \arg\min_{\theta} D_{KL}(P_{true} \mid\mid P_{model}(\theta)) = \theta^* \end{gathered} \tag{3.3}$$

Равенство (3.3) верно в силу равномерной сходимости $\mathscr{L}(\theta)$ и в силу определения KL-дивергенции.

$$D_{KL}(P||Q) = \int_{-\infty}^{\infty} p(x) \log\left(\frac{p(x)}{q(x)}\right) dx = \mathbb{E}_{x \sim p(x)}[\log\left(\frac{p(x)}{q(x)}\right)] \tag{3.4}$$

■

Докажем, что LoRA применима к задаче классификации. LoRA используется для решения различных проблем seq2seq, таких как: [29], [5], [4]. Этот подход особенно популярен в задачах преобразования видео в текст, так как им(задачам) свойственны "богатые распределения входных данных и разнообразие задач, обусловленные дополнительным визуальным входным данным-[4].

В то же время для решения задачи классификации с помощью BERT [21] требуется не более чем дополнительный softmax слой после BERT [18]:

$$\begin{gathered} p(c \mid \mathbf{x}) = \text{softmax}(W^T \mathbf{x}) \\ \hat{\mathbf{y}} = \text{softmax}\left(W^T \mathbf{x}\right) = \frac{\exp\left(W^T \mathbf{x}\right)}{\sum_{i=1}^{k} \exp\left(W^T \mathbf{x}\right)_i} \end{gathered} \tag{3.5}$$

Где **x** - это выходной результат последнего слоя BERT.
Структура BERT:

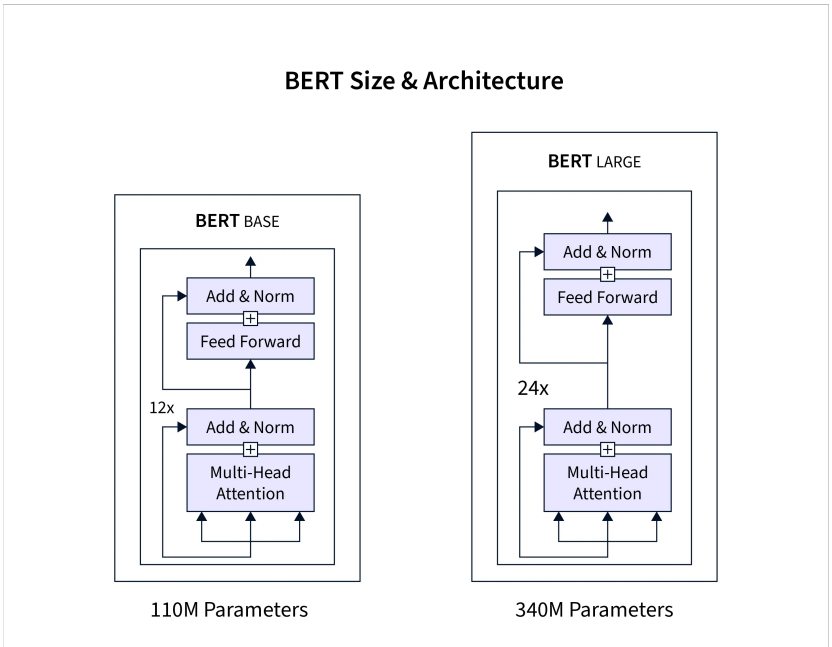


Рис. 1: BERT structure

Где, согласно [19],
Attention:

$$\begin{aligned} Q^{(h)}\left(\mathbf{x}_i\right) = W_{h,q}^T \mathbf{x}_i, \quad K^{(h)}\left(\mathbf{x}_i\right) = W_{h,k}^T \mathbf{x}_i, \\ V^{(h)}\left(\mathbf{x}_i\right) = W_{h,v}^T \mathbf{x}_i, \quad W_{h,q}, W_{h,k}, W_{h,v} \in \mathbb{R}^{d \times k} \end{aligned} \tag{3.6}$$

$$\begin{aligned} \alpha_{i,j}^{(h)} = \operatorname{softmax}_j \left(\frac{\left\langle Q^{(h)}\left(\mathbf{x}_i\right), K^{(h)}\left(\mathbf{x}_j\right) \right\rangle}{\sqrt{k}} \right), \\ \mathbf{u}_i' = \sum_{h=1}^{H} W_{c,h}^T \sum_{j=1}^{n} \alpha_{i,j}^{(h)} V^{(h)}\left(\mathbf{x}_j\right), \end{aligned} \tag{3.7}$$

LayerNorm, Feed Forward Network, LayerNorm:

$$\begin{aligned} \mathbf{u}_i &= \operatorname{LayerNorm}\left(\mathbf{x}_i + \mathbf{u}_i'; \gamma_1, \beta_1\right), \\ \mathbf{z}_i' &= W_2^T \operatorname{ReLU}\left(W_1^T \mathbf{u}_i\right), \\ \mathbf{z}_i &= \operatorname{LayerNorm}\left(\mathbf{u}_i + \mathbf{z}_i'; \gamma_2, \beta_2\right), \end{aligned} \tag{3.8}$$

Layer normalization может быть переписана в соответствии с оригинальной статьей [3]:

$$\begin{aligned} &\operatorname{LayerNorm}(\mathbf{z}; \gamma, \beta) = \gamma \frac{(\mathbf{z} - \mu_{\mathbf{z}})}{\sigma_{\mathbf{z}}} + \beta, \\ &\gamma, \beta \in \mathbb{R}^k. \\ &\mu_{\mathbf{z}} = \frac{1}{k} \sum_{i=1}^{k} \mathbf{z}_i, \quad \sigma_{\mathbf{z}} = \sqrt{\frac{1}{k} \sum_{i=1}^{k} \left(\mathbf{z}_i - \mu_{\mathbf{z}}\right)^2}. \end{aligned} \tag{3.9}$$

3.2 Теорема 2

Теорема 2. *В рамках задачи классификации, при заданных условиях:*

- *Модель семейства Bert с указанной выше математической структурой и дополнительным слоем*

$$\hat{\mathbf{y}} = \text{softmax}\left(W_{upd}^T \mathbf{x}\right) = \frac{\exp\left(W_{upd}^T \mathbf{x}\right)}{\sum_{i=1}^{k} \exp\left(W_{upd}^T \mathbf{x}\right)_i} \tag{3.10}$$

где

$$W_{upd} = \underset{(d\times k)}{W} + \underset{(d\times k)}{\Delta W} \tag{3.11}$$

и x - это выходной результат BERT, W - матрица весов, ΔW - матрица обновленных весов.

- *Данная модель Bert без дополнительного слоя также корректно работает с аппроксимацией*

$$\underset{(d\times k)}{\Delta W} = \underset{(d\times r)}{A} \times \underset{(r\times k)}{B} \tag{3.12}$$

- *Выполняется lemma 1.0.1. (можно считать данную модель состоятельной).*

Тогда можно утверждать, что при (3.14) *заданная модель BERT с дополнительным слоем гарантирует корректную выходную матрицу.*

Доказательство. Докажем, что выход из дополнительного слоя корректен:
По дистрибутивному свойству сложения матриц и ассоциативному свойтсву произведения матриц:

$$\hat{\mathbf{y}} = \text{softmax}\left(W_{upd}^T \mathbf{x}\right) = \text{softmax}\left((W + \Delta W)^T \mathbf{x}\right) =$$

$$= \frac{\exp\left(W^T \mathbf{x} + \Delta W^T \mathbf{x}\right)}{\sum_{i=1}^{k} \exp\left(W^T \mathbf{x} + \Delta W^T \mathbf{x}\right)_i} = \tag{3.13}$$

$$= \frac{\exp\left(W^T \mathbf{x}\right) \exp\left(\Delta W^T \mathbf{x}\right)}{\sum_{i=1}^{k} \exp\left(W^T \mathbf{x}\right)_i \exp\left(\Delta W^T \mathbf{x}\right)_i}$$

где $\mathbf{x}$ - выходная матрица последнего слоя Bert. $\mathbf{x}$ корректна по условию.
В предложенной модели с использованием LoRA:

$$\begin{aligned}\hat{\mathbf{y}} = \text{softmax}\left(W_{upd}\mathbf{x}\right) = \text{softmax}\left(\left(W + AB\right)^T\mathbf{x}\right) = \\ = \frac{\exp\left(W^T\mathbf{x} + (AB)^T\mathbf{x}\right)}{\sum_{i=1}^{k}\exp\left(W^T\mathbf{x} + (AB)^T\mathbf{x}\right)_i} = \\ = \frac{\exp\left(W^T\mathbf{x}\right)\exp\left((AB)^T\mathbf{x}\right)}{\sum_{i=1}^{k}\exp\left(W^T\mathbf{x}\right)_i\exp\left((AB)^T\mathbf{x}\right)_i}\end{aligned} \tag{3.14}$$

где $\mathbf{x}$ также выходная матрица BERT с LoRA и $\mathbf{x}$ корректна по условию.
Так как финальные размерности остались неизменными, как и $W^T\mathbf{x}$, легко заметить:

$$\begin{aligned}\underset{(k\times d)}{\Delta W^T} \times \mathbf{x} = u \\ (\underset{(d\times r)}{A} \times \underset{(r\times k)}{B})^T \times \mathbf{x} = (\underset{(k\times r)}{B^T} \times \underset{(r\times d)}{A^T}) \times \mathbf{x} = u^*\end{aligned} \tag{3.15}$$

Так как (3.14), можно заключить что $u^* = u$ и является корретной матрицей, а следовательно предложенная модель работает корректно. ■

4 Вычислительный эксперимент

4.1 Данные

Открытый исходный датасет для мультиклассовой классификации текстов, написанных человеком и различными языковыми моделями [22]. Изначально было представлено 6 классов (включая человека), но из-за технических ограничений количество классов было сокращено до 4: ChatGPT, Davinci, Cohere, Humans. Всего в датасете 21 000 текстов с разметкой по классам.

4.2 Предобработка

Тексты были токенизированы при помощи AutoTokenizer [24]. Дополнительная обработка не требуется из-за структуры модели [21].

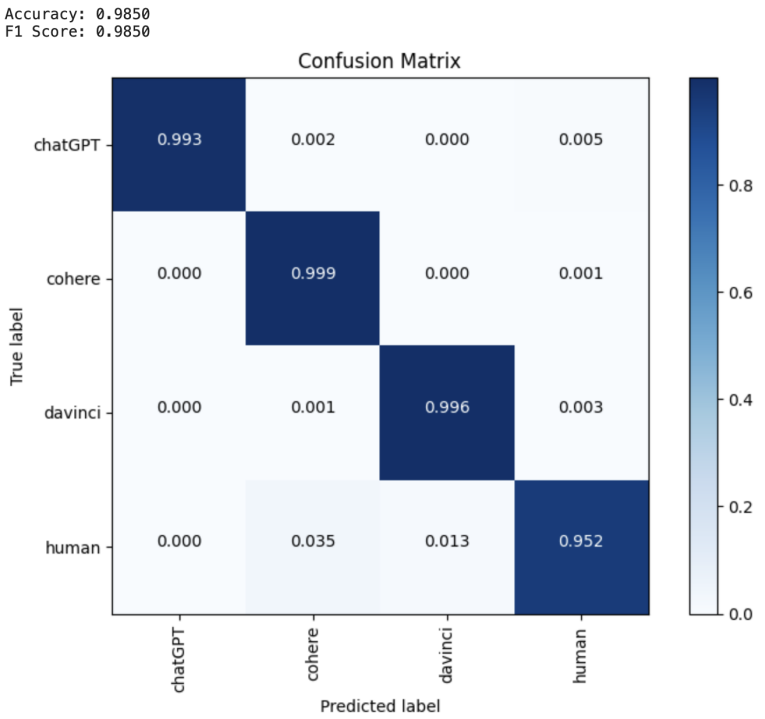
4.3 Эксперименты

Для всех экспериментов использовалась предобученная модель DistilRoBERTa base (далее в тексте - DRoBERTa-base) [14]. Модель использовалась со следующими гиперпараметрами: доля тренировочного/тестового набора данных - 0.9/0.1; 3 эпохи обучения. Для экспериментов с использованием алгоритма LoRA были использованы все вышеуказанные параметры, а также ранг матриц аппроксимации r = 5.

4.4 Эксперимент (1)

Предобученная модель DRoBERTa-base обучилась на всем обучающем датасете.

После обучения для оценки использовались матрица ошибок и метрики точности, полноты и F1-меры:
train_runtime: 4041.3188



Classification report:

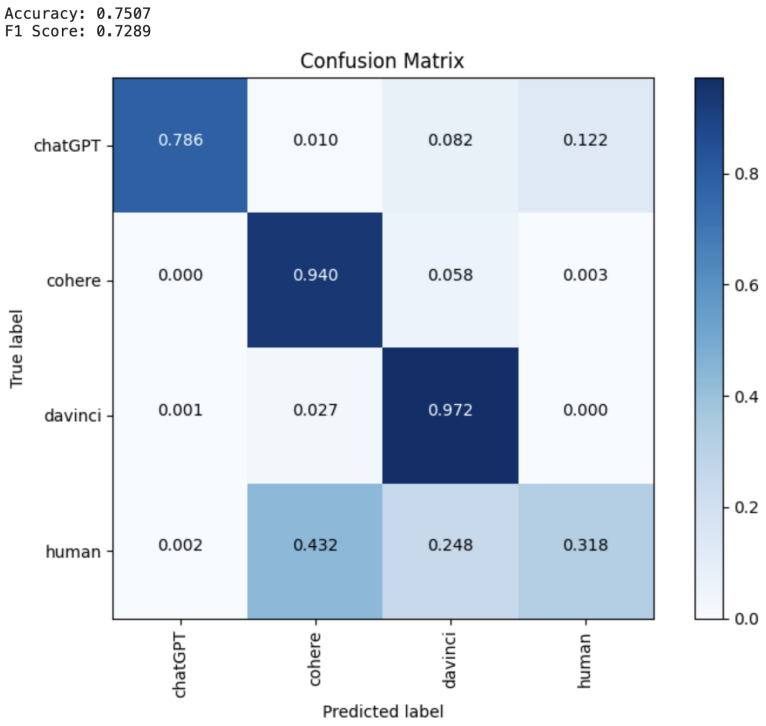
	precision	recall	f1-score	support
chatGPT	1.0000	0.9933	0.9967	1200
cohere	0.9638	0.9992	0.9812	1199
davinci	0.9860	0.9965	0.9912	1134
human	0.9913	0.9517	0.9711	1200
accuracy			0.9850	4733
macro avg	0.9853	0.9852	0.9850	4733
weighted avg	0.9853	0.9850	0.9849	4733

Рис. 2: результат эксперимента 1

4.5 Эксперимент (2)

Предобученная модель DRoBERTa-base с использованием LoRA обучилась на всем обучающем датасете.

trainable params: 685828 all params: 82807304 || trainable%: 0.8282
Только 0.828% параметров обучаются при использовании LoRA
train_runtime: 3210.977



Classification report:

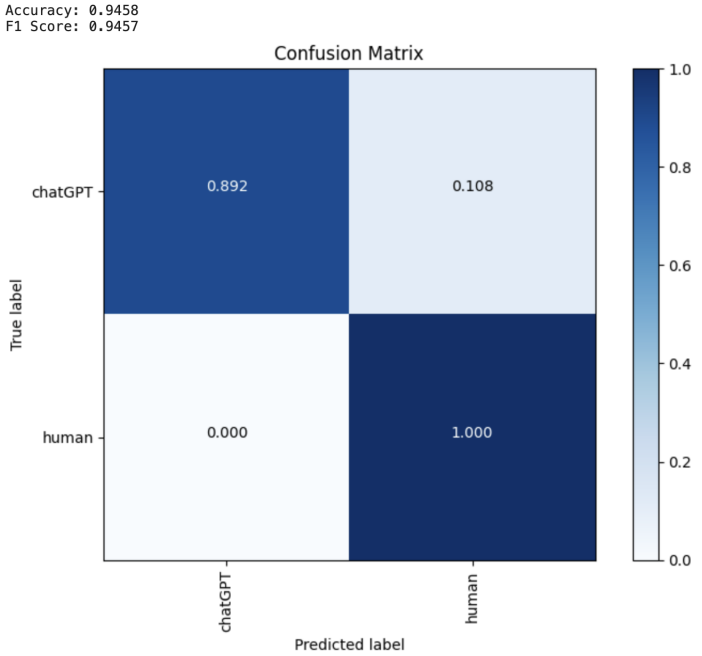
	precision	recall	f1-score	support
chatGPT	0.9968	0.7858	0.8788	1200
cohere	0.6673	0.9399	0.7805	1199
davinci	0.7033	0.9718	0.8160	1134
human	0.7175	0.3175	0.4402	1200
accuracy			0.7507	4733
macro avg	0.7712	0.7538	0.7289	4733
weighted avg	0.7722	0.7507	0.7277	4733

Рис. 3: результат эксперимента 2

4.6 Эксперимент (3)

Три независимые модели DRoBERTa-base с исп. LoRA обучались на парах классов: GPT vs Human, Davinci vs Human, Cohere vs Human.

ChatGPT vs Human
train_runtime: 1633.8114

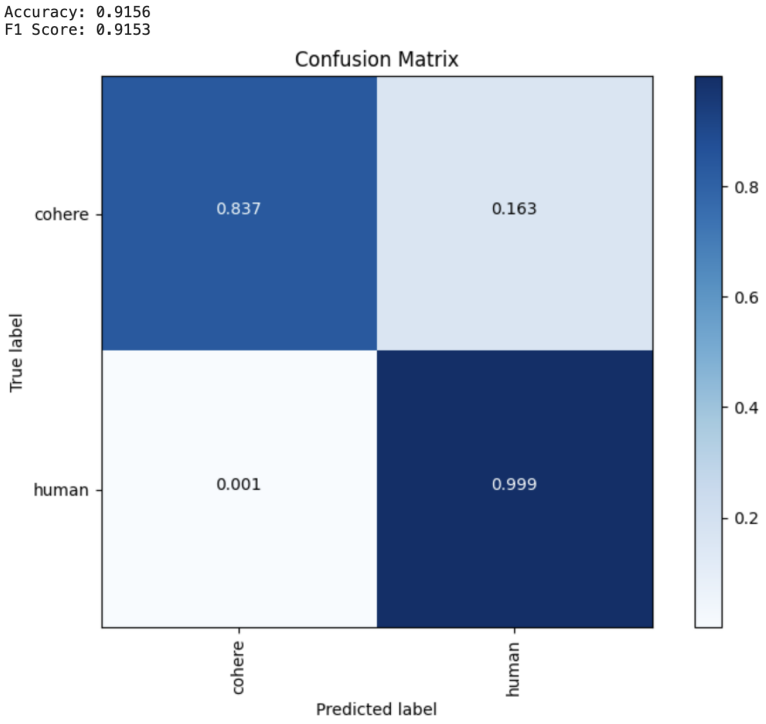


Classification report:

	precision	recall	f1-score	support
chatGPT	1.0000	0.8917	0.9427	1200
human	0.9023	1.0000	0.9486	1200
accuracy			0.9458	2400
macro avg	0.9511	0.9458	0.9457	2400
weighted avg	0.9511	0.9458	0.9457	2400

Рис. 4: результат эксперимента 3.1

Cohere vs Human
train_runtime: 1583.556

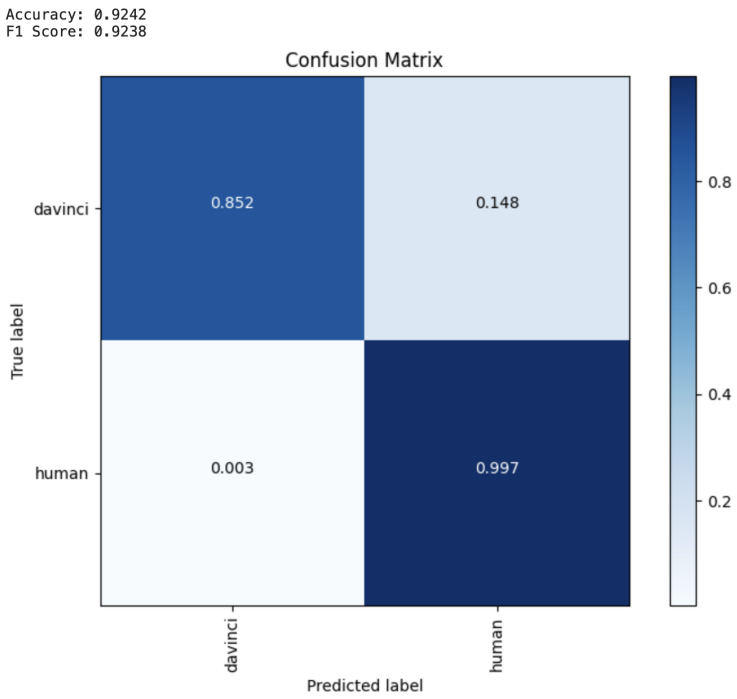


Classification report:

	precision	recall	f1-score	support
cohere	0.9990	0.8367	0.9107	1200
human	0.8525	0.9991	0.9200	1134
accuracy			0.9156	2334
macro avg	0.9258	0.9179	0.9153	2334
weighted avg	0.9278	0.9156	0.9152	2334

Рис. 5: результат эксперимента 3.2

Davinci vs Human
train_runtime: 1632.395



Classification report:

	precision	recall	f1-score	support
davinci	0.9961	0.8517	0.9182	1200
human	0.8705	0.9967	0.9293	1200
accuracy			0.9242	2400
macro avg	0.9333	0.9242	0.9238	2400
weighted avg	0.9333	0.9242	0.9238	2400

Рис. 6: результат эксперимента 3.3

5 Заключение

В работе рассматривался метод LoRA снижения размерности пространства обучаемых параметров в задаче классификации текстов, написанных большими языковыми моделями. Была сформулирована и доказана теорема о конструктивности предложенного метода.

В ходе эксперимента на датасете из текстов, написанных как языковыми моделями, так и человеком, была доказана эффективность предложенного метода. При решении задачи мультиклассовой классификации предложенная модель BERT & LoRA тратит меньше ресурсов, чем модель без использования LoRA, но метрики качества падают. Однако, при решении тремя одинаковыми независимыми моделями задачи бинарной классификации с последующим усреднением метрики качетва растут, а использование ресурсов — нет.

Список литературы

[1] Sara Abdali, Richard Anarfi, CJ Barberan, and Jia He. Decoding the ai pen: Techniques and challenges in detecting ai-generated text. *arXiv preprint arXiv:2403.05750*, 2024.

[2] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020.

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[4] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36, 2024.

[5] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.

[6] Valeria Fonti and Eduard Belitser. Feature selection using lasso. *VU Amsterdam research paper in business analytics*, 30:1–25, 2017.

[7] Quanquan Gu, Zhenhui Li, and Jiawei Han. Generalized fisher score for feature selection. *arXiv preprint arXiv:1202.3725*, 2012.

[8] Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. Mgtbench: Benchmarking machine-generated text detection. *arXiv preprint arXiv:2303.14822*, 2023.

[9] G Hinton, O Vinyals, and J Dean. Nips deep learning and representation learning workshop, 2015.

[10] Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee,

and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*, 2023.

[11] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[12] Minhyeok Lee. A mathematical investigation of hallucination and creativity in gpt models. *Mathematics*, 11(10):2320, 2023.

[13] Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. Gpt detectors are biased against non-native english writers. *Patterns*, 4(7), 2023.

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[15] Ivan V Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[16] Rukhma Qasim, Waqas Haider Bangyal, Mohammed A Alqarni, and Abdulwahab Ali Almazroi. A fine-tuned bert-based transfer learning approach for text classification. *Journal of healthcare engineering*, 2022, 2022.

[17] Zubair Qazi, William Shiao, and Evangelos E Papalexakis. Gpt-generated text detection: Benchmark dataset and tensor-based detection method. *arXiv preprint arXiv:2403.07321*, 2024.

[18] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune bert for text classification? In *Chinese computational linguistics: 18th China national conference, CCL 2019, Kunming, China, October 18–20, 2019, proceedings 18*, pages 194–206. Springer, 2019.

[19] John Thickstun. The transformer model in equations. *University of Washington: Seattle, WA, USA*, 2021.

[20] Vladimir Vapnik, Rauf Izmailov, et al. Learning using privileged information: similarity control and knowledge transfer. *J. Mach. Learn. Res.*, 16(1):2023–2049, 2015.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[22] Yuxia Wang, Jonibek Mansurov, Petar Ivanov, jinyan su, Artem Shelmanov, Akim Tsvigun, Osama Mohammed Afzal, Tarek Mahmoud, Giovanni Puccetti, Thomas Arnold, Chenxi Whitehouse, Alham Fikri Aji, Nizar Habash, Iryna Gurevych, and Preslav Nakov. Semeval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 2041–2063, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[23] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[24] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[25] Xianjun Yang, Wei Cheng, Linda Petzold, William Yang Wang, and Haifeng Chen. Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text. *arXiv preprint arXiv:2305.17359*, 2023.

[26] Xiao Yu, Yuang Qi, Kejiang Chen, Guoqiang Chen, Xi Yang, Pengyuan Zhu, Weiming Zhang, and Nenghai Yu. Gpt paternity test: Gpt generated text detection with gpt genetic inheritance. *arXiv preprint arXiv:2305.12519*, 2023.

[27] Ali Zare, Alp Ozdemir, Mark A Iwen, and Selin Aviyente. Extension of pca to higher order data structures: An introduction to tensors, tensor decompositions, and tensor pca. *Proceedings of the IEEE*, 106(8):1341–1358, 2018.

[28] Yujia Zhai, Wei Song, Xianjun Liu, Lizhen Liu, and Xinlei Zhao. A chi-square statistics based feature selection method in text classification. In *2018 IEEE 9th International conference on software engineering and service science (ICSESS)*, pages 160–163. IEEE, 2018.

[29] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.