Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ

Школа профессионального и академического образования

ДОПУСТИТЬ К ЗАЩИТЕ ПЕРЕД ГЭК

Директор ШПиАО
_____ Д.В. Денисов
(подпись) (Ф.И.О.)
«_____»_____2024 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

ДООБУЧЕНИЕ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ ДЛЯ РЕШЕНИЯ СПЕЦИАЛИЗИРОВАННЫХ ЗАДАЧ

Научный руководитель: Долганов Антон Юрьевич
к.т.н., доцент _____ подпись

Нормоконтролер: Огуренко Егор Владимирович _____ подпись

Студент группы: РИМ-220962 Молчанова Татьяна Александровна _____ подпись

Екатеринбург
2024

РЕФЕРАТ

Выпускная квалификационная работа магистра 79 с., 13 рис., 21 табл., 48 источников.

ЯЗЫКОВОЕ МОДЕЛИРОВАНИЕ, БОЛЬШИЕ ЯЗЫКОВЫЕ МОДЕЛИ, ТРАНСФОРМЕРЫ, ДООБУЧЕНИЕ МОДЕЛЕЙ, КВАНТИЗАЦИЯ МОДЕЛЕЙ, МАШИННЫЙ ПЕРЕВОД, МУЛЬТИЯЗЫЧНЫЙ МАШИННЫЙ ПЕРЕВОД

Цель работы: сравнение методов дообучения больших языковых моделей для решения специализированных задач. Специализированная задача: мультиязычный машинный перевод в сфере информационной безопасности.

Объект исследования: большие языковые модели.

Методы исследования: разведочный анализ данных, подтверждающий анализ данных, метод квантизации моделей QLORA, методы дообучения zero-shot, few-shot, и PEFT, метод оценивания BLEU.

Ограничения: одно устройство с одной устройство с одной видеокартой от 12 до 24 ГБ. В связи с этим не рассматривался метод дообучения full fine-tune.

Результаты работы: проведено сравнение методов zero-shot, few-shot и PEFT применительно к дообучению больших языковых моделей Mistral и Llama 2 для решения задачи мультиязычного машинного перевода в сфере информационной безопасности.

Выпускная квалификационная работа выполнена в текстовом редакторе Microsoft Word и представлена в твердой копии.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

В последние годы для решения задач в области обработки естественного языка (NLP), обработки сигналов и компьютерного зрения активное распространение получили модели нейронных сетей. Исследования показывают, что чем больший размер имеют такие модели, тем более точные результаты они демонстрируют [1].

Особенно широкое распространение в последнее время получили большие языковые модели, предварительно обученные на больших корпусах текстов [2]. Такие языковые модели стали популярным инструментом благодаря своей способности без дообучения решать общие задачи из области NLP. Тем не менее, для решения специализированных задач большие языковые модели без дополнительного обучения не совсем подходят и показывают низкие результаты [3].

Таким образом, **актуальность** данного исследования обусловлена возрастающим вниманием со стороны исследователей искусственного интеллекта к большим языковым моделям и одновременно недостаточной способностью таких моделей решать специализированные задачи без дообучения.

Объектом нашего исследования являются большие языковые модели.

Предметом изучения выступают методы дообучения больших языковых моделей для решения специализированных задач.

Целью исследования является сравнение методов дообучения больших языковых моделей для решения специализированных задач. Достижение поставленной цели предполагает решение следующих **задач**:

1) провести обзор источников литературы по языковому моделированию;

2) выбрать большие языковые модели для дообучения, выбрать методы дообучения моделей, сформировать датасет для дообучения моделей;

3) провести сравнение методов дообучения и дообученных языковых моделей в решении специализированной задачи.

При написании данной работы использовались следующие **методы**:

1) методы анализа: разведочный анализ данных (EDA) и подтверждающий анализ данных (CDA);

2) метод квантизации моделей QLORA,

3) методы дообучения моделей: zero-shot, few-shot, и PEFT;

4) методы оценки моделей: семейство метрик BLEU (BLEU-1, BLEU-2, BLEU-3, BLEU-4, BLEU-S).

Исследование проводилось локально на одном устройстве с одной видеокартой (24 ГБ). В связи с эти не рассматривался метод full fine-tune.

Научная новизна исследования заключается в следующем:

1. Впервые проведено исследование на способность больших языковых моделей решать задачу машинного перевода в сфере информационной безопасности.

2. Впервые проведена оценка способностей больших языковых моделей Mistral и Llama 2 осуществлять перевод одновременно с и на английский, русский и испанский языки.

Проведённое диссертационное исследование имеет следующую **практическую значимость**:

1. С помощью предложенных методов обучены большие языковые модели решать задачу мультиязычного машинного перевода с и на три языка: русский, английский и испанский.

2. Предложенные методы позволяют дообучить модели на решение специализированных задач.

3. Обученные модели выложены в открытый доступ и могут быть использованы в решении переводческих задач.

4. Собранный датасет выложен в открытый доступ и может быть использован для дообучения и/или оценки языковых моделей.

Глава 1. Теоретические основы языкового моделирования для решения специализированных задач

1.1. Языковое моделирование

Язык — это отличительная способность человека выражать свои мысли, способ общения с другими. С начала XX века и по сегодняшний день математики, информатики, статистики и исследователи искусственного интеллекта пытаются смоделировать язык с помощью научного аппарата своих дисциплин [4].

В общем смысле под языковым моделированием понимается «способность вычислять распределения вероятностей по последовательностям слов» [5].

Математик А.А. Марков считается первым ученым, чьи работы посвящены языковому моделированию, хотя термина «языковая модель» в то время еще не существовало. В 1913 году на примере поэзии А.С. Пушкина, учёный разработал языковую модель, которую в дальнейшем будут называть «марковской моделью» [6]. Её можно представить в виде формулы:

$$p(w_1, w_2, \dots, w_N) = \prod_{i=1}^{N} p(w_i | w_1, w_2, \dots, w_{i-1}), \quad (1)$$

где N – количество слов в последовательности;

$w_1, w_2, \dots, w_N$ – последовательность слов от 1 до N;

$p(w_1, w_2, \dots, w_N)$ – вероятностное распределение последовательности слов от 1 до N;

$p(w_i | w_1, w_2, \dots, w_{i-1})$ – вероятностное распределение при условии всех прочих членов последовательности, кроме i-го.

Основатель теории информатики К. Шеннон в 1948 опубликовал свой фундаментальный труд «Математическая теория связи», где ввёл понятия энтропии, кросс-энтропии, а также изучил свойства n-граммных моделей (т.е. моделей состоящих из последовательностей n-слов, так, например, последовательность из двух слов будет называться «биграммой», из трёх – «триграммой» и т.д.). Если представить язык как данные, генерируемые случайным процессом, то энтропия распределения вероятностей n-грамм определяется следующим образом:

$$H_N(p) = -\sum_{w_1,w_2,\dots,w_N} p(w_1, w_2, \dots, w_N) \times \log_2 p(w_1, w_2, \dots, w_N), \qquad (2)$$

где N – количество n-грамм в последовательности;

$w_1, w_2, \dots, w_N$ – последовательность n-грамм от 1 до N;

$p(w_1, w_2, \dots, w_N)$ – вероятностное распределение последовательности n-грамм от 1 до N.

В таком случае, кросс-энтропию распределения вероятностей n-грамм относительно «истинного» распределения вероятностей данных можно представить как:

$$H_N(p, q) = -\sum_{w_1,w_2,\dots,w_N} p(w_1, w_2, \dots, w_N) \times \log_2 q(w_1, w_2, \dots, w_N), \qquad (3)$$

где N – количество n-грамм в последовательности;

$w_1, w_2, \dots, w_N$ – последовательность n-грамм от 1 до N;

$q(w_1, w_2, \dots, w_N)$ – вероятностное распределение последовательности n-грамм от 1 до N;

$p(w_1, w_2, \ldots, w_N)$ – истинное вероятностное распределение последовательности n-грамм от 1 до *N*.

Основной смысл формулы кросс-энтропии состоит в том, что если одна языковая модель может более точно предсказать последовательность слов, чем другая, то у нее должна быть более низкая кросс-энтропия. Таким образом, работа Шеннона представляет собой важный инструмент оценки языкового моделирования, который используется и в настоящее время [7].

Однако модель n-грамм сильно ограничена в своей способности к обучению. Так, n-граммы в таких моделях кодируются как разреженный вектор, а количество параметров модели увеличивается экспоненциально в зависимости от размера словаря (т.н. «проклятие размерности»). Математически сложность такого алгоритма можно представить как:

$$O = V^2, \tag{4}$$

где V – объём словаря.

То есть, если в нашем словаре будет содержаться 15 000 n-грамм, то у языковой модели на основе такого словаря будет 225 000 000 признаков, большая часть из которых будет заполнена нулями. Иными словами, такие языковые модели будут плохо работать с большими контекстами.

Чтобы снизить вычислительную сложность статистического языкового моделирования, в 2001 году была выдвинута гипотеза о кодировании слов с помощью эмбеддингов, т.е. плотных векторных представлений, которые получаются в результате дополнительных преобразований. Количество признаков в таких моделях (например, 50, 100 или 300) гораздо меньше размера самого словаря (например, 15 000) [8].

Использование эмбеддингов получило широкое распространение в нейросетевых архитектурах. Одной из первых нейросетевых моделей стала

Word2Vec, предложенная в 2013 году. Модель была представлена в двух модификациях: CBOW (Continuous Bag-of-Words) и Skip-gram [9].

В основе CBOW лежит нейронная сеть прямого распространения, где нелинейный скрытый слой удаляется, а слой проекции используется для всех слов (а не только для матрицы проекции); таким образом, все слова проецируются в одну и ту же позицию (их векторы усредняются). Стоит отметить, что порядок слов при такой архитектуре не имеет значения.

Вычислительная сложность CBOW составляет:

$$O = E \times T \times (N \times D + D \times \log_2(V)), \quad (5)$$

где E – количество эпох обучения;

T – количество слов в тренировочном датасете;

N – количество слов в левом (предыдущем) контексте;

D – размер проекции;

V – размер выходного слоя.

Вторая архитектура похожа на CBOW, но основное отличие состоит в том, что CBOW предсказывает текущее слово на основе контекста, а Skip-gram предсказывает окружающие слова по текущему слову.

Вычислительная сложность Skip-gram составляет:

$$O = E \times T \times C \times (D + D \times \log_2(V)), \quad (6)$$

где E – количество эпох обучения;

T – количество слов в тренировочном датасете;

C – максимальное расстояние между словами;

D – размер проекции;

V – размер выходного слоя.

Таким образом, применение эмбеддингов значительно сокращает вычислительную сложность языковых моделей, позволяя задействовать в обучении больше документов.

Использование эмбеддингового слоя стало распространено и в других «классических» архитектурах нейронных сетей: в многослойный перцептронах, свёрточных и рекуррентных нейронных сетях, включая его подтипы LSTM и GRU. Однако данные архитектуры имею ряд недостатков:

– Многослойный перцептрон игнорирует информацию о последовательности, рассматривая все входные токены как независимые. Кроме того, данная архитектура с трудом улавливает семантическую информацию во входной последовательности.

– Свёрточные и «ванильные» рекуррентные нейронные сети плохо работают с длинными контекстами ввиду угасания градиентов.

– Семейство рекуррентных сетей последовательно обрабатывают последовательности токенов, хотя параллельные вычисления производить гораздо быстрее [10].

Данные причины послужили предпосылками к созданию в 2017 году новой архитектуры нейронных сетей – трансформерной. Ключевым фактором, благодаря которому трансформеры получили широкое распространение, является механизм внимания. Функцию внимания можно описать как сопоставление запроса и пар ключ-значение с выходными данными, где запрос, ключи, значения и выходные данные являются векторами. Выход модели рассчитывается как взвешенная сумма значений, где вес, присвоенный каждому значению, вычисляется с помощью сопоставления запроса с соответствующим ключом [11].

Модели трансформеров можно разделить на три типа:

1. Энкодер-трансформеры: на вход таким моделям поступает текст, а на выходе получаются закодированные эмбеддинги. Данный тип трансформеров обычно используется для задач классификации текстов или

токенов. Примерами таких нейросетей можно назвать BERT-подобные модели.

2. Декодер-трансформеры: на вход таким моделям поступает текст, а на выходе модели предсказывается следующее слово, которое затем снова подаётся на вход модели. В связи с этим, декодер-трансформеры, в основном, используются для решения задач генерации текста. Примерами таких моделей является семейство GPT моделей.

3. Энкодер-декодер трансформеры (или полные трансформеры): на вход таким моделям поступает текст, который сначала обрабатывается энкодером для создания промежуточного пространства, которое затем переводится в желаемый формат с помощью декодера. Данная архитектура получила своё применение в задачах машинного перевода, а яркими представителями таких трансформеров являются модели типа T5 [12].

Ещё одной важной особенностью трансформерных моделей стала их способность к дообучению. Веса трансформерных моделей, полученные благодаря предварительному обучению на больших корпусах текстов, можно относительно быстро донастроить для решения специализированных задач: для этого потребуется лишь небольшой корпус текстов, размеченных под данную задачу.

К 2020 году исследователи из компании по разработке искусственного интеллекта OpenAI приходят к выводу, что увеличение размеров языковых моделей приводит к улучшению их лингвистических способностей в целом [1] и появлению у моделей так называемых «возникающих способностей» (emergent abilities), т.е. способностей решать новые задачи без дополнительного дообучения [13]. Так, например, модель GPT-3 имеет 175 миллиарда параметров и, по заявлению её создателей, способна решать любые задачи на английском языке [4].

Языковые модели трансформерной архитектуры с большим количеством параметров обучения получили название «больших языковых моделей».

Учёные расходятся во мнениях, когда языковую модель можно назвать «большой». Одни считают, что первые трансформеры (BERT и GPT-2) уже имели достаточное количество параметров, чтобы считаться «большими моделями» (340 миллионов и 1,5 миллиарда соответственно) [2]. Другие придерживаются мнения, что большие языковые модели – это GPT-3 и похожие модели, появившиеся после неё [10]. В нашем исследовании под большой языковой моделью будет пониматься трансформерная нейросеть, обученная для решения задачи языкового моделирования и имеющая несколько миллиардов параметров.

Основные компании, которые выпускают большие языковые модели – это Google и OpenAI. Первой принадлежат GLaM, PaLM, PaLM2, LaMDA, Bard и Gemini. Второй – GPT-3, GPT-4, InstructGPT, ChatGPT и Codex. К сожалению, все перечисленные модели имеют закрытый исходный код [10].

Безусловно, существуют и открытые языковые модели, например Llama, Llama-2, NLLB, BLOOM, Mistral, Mixtral и другие. Поскольку среди открытых моделей достаточно высокие позиции занимают модели Llama-2 и Mistral [14], то в нашем исследовании мы будем дообучать именно их. Стоит также отметить, что обе модели имеют 7 миллиардов параметров, поэтому их удобно сравнивать между собой. Кроме того, выбор данных моделей обусловлен тем, что они могут быть дообучены локально лишь на одном графическом ускорителе.

Среди преимуществ больших языковых моделей как класса можно выделить:

1. Широкий спектр применений. Из-за обучения на большом корпусе данных и «возникающих способностей» большие языковые модели способны выполнять множество различных задач. Так, например, GPT-3 и GPT-3.5. способны понимать и генерировать как естественный язык, так и код, а GPT-4 может обрабатывать задачи, связанные с обработкой текстов и изображений [10].

2. Адаптивность. Поскольку большие языковые модели уже обучены на огромных объемах данных, для обучения новым задачам им не требуются большие наборы новых данных [15].

Тем не менее, большие языковые модели обладают и рядом недостатков:

1. Неэффективное решение специализированных задач. Поскольку большие языковые модели обучались на текстах общей тематики, им может быть недостаточно знаний для работы со специализированной областью или задачей. Например, языковая модель, обученная на общих статьях из интернета, может столкнуться с проблемами при создании медицинского отчета, который включает в себя множество медицинских профессионализмов [4].

2. Проблема забывания. После дообучения под какую-либо специализированную задачу языковые модели имеют тенденцию к снижению способностей выполнять прочие задачи [16].

3. Предвзятости в обучающей выборке. Как правило, тексты, на которых обучались большие языковые модели, отражают мнения и стереотипы, распространённые в обществе. Обучаясь на такой выборке, языковые модели наследуют предвзятые мнения, касательно пола, национальной, расовой и религиозной принадлежности [17].

4. Галлюцинации. Важной проблемой работы языковых моделей является создание ими текстов с ложной информацией. Сгенерированный текст может противоречить существующему источнику информации (т.е. модель врёт), либо не может быть подтверждён доступным источником (т.е. модель попросту выдумала ответ, несоответствующий реальности) [4].

5. Ресурсоёмкость. Как обучение, так и использование больших языковых моделей требует значительных вычислительных ресурсов, включая процессорное время и объем памяти [18].

Именно используя преимущества больших языковых моделей, мы будем исследовать, как бороться с первой упомянутой проблемой: неэффективным

решением специализированных задач. В следующем подразделе мы изучим, какие специализированные задачи можно решать с помощью данных языковых моделей.

1.2. Специализированные задачи, решаемые с помощью больших языковых моделей

В данном подразделе мы рассмотрим основные задачи, решаемые при помощи больших языковых моделей. Самые распространённые из них генеративные задачи, а именно: продолжение текста, условная генерация текста, генерация кода.

Задачи продолжения текста предполагают простое предсказание следующего токена в тексте. Как правило, такие задачи оцениваются при помощи перплексии (perplexity), или меры того, насколько распределение вероятностей хорошо предсказывает выборку. Исследования показывают, что увеличение размера языковой модели приводит к снижению перплексии [1].

Условная генерация текста подразумевает создание текстов, удовлетворяющих конкретным требованиям задачи на основе заданных условий. Частными случаями условной генерации является машинный перевод, суммаризация текста и вопросно-ответные системы [4].

Помимо задач генерации текста на естественном языке, широкое распространение получили и задачи генерации программного кода. Для проверки корректности сгенерированного кода используются метрика pass@k, заключающаяся в подсчёте корректно пройденных тестов сгенерированным кодом [19].

Также встречаются и исследования, направленные на решения математических задач с помощью больших языковых моделей. Однако на данный момент языковые модели справляются с данными задачами с переменным успехом, поскольку решение математических задач предполагает

не только понимание языка математики, но и построение логических цепочек решения задач [20].

Можно также встретить исследования, посвящённые дообучению больших языковых моделей для решения специализированных задач в определённой области знаний. Наиболее распространено дополнительное обучение на медицинских, юридических и финансовых данных [4].

Что касается сферы информатики, то видится недостаток работ, посвящённых дообучению больших языковых моделей для решения задач в данной области [3]. Особенно мало разработанной является перевод в сфере информационной безопасности: были проведены исследования лишь для моделей простых рекуррентных нейронных сетей [21].

Поскольку для нашего исследования, в качестве специализированной задачи для дообучения больших языковых моделей мы выбрали мультиязычный машинный перевод в сфере информационной безопасности, то рассмотрим задачи машинного перевода поподробнее.

Машинный перевод заключается в поиске наиболее вероятной последовательности токенов перевода на целевой язык при условии заданной последовательности токенов исходного языка [5]. Формализовать данное определение можно с помощью следующей формулы:

$$P(y_1, y_2, \dots, y_m | x_1, x_2, \dots, x_n), \tag{7}$$

где $x_1, x_2, \dots, x_n$– предложение на исходном языке;

$y_1, y_2, \dots, y_m$– предложение на целевом языке.

Языковые модели, предназначенные для решения задач машинного перевода, эволюционировали как и обычные языковые модели: сначала были статистическими (основанными на пословных переводах и n-граммах), затем стали нейросетевыми, а с появления механизма внимания – траснформерными [22].

В последние годы, большие языковые модели также стали активно применяться для решения переводческих задач. Тем не менее, без дополнительного обучения на выполнение переводов, такие модели демонстрируют посредственное качество [23].

Исследователи, занимающиеся проблематикой машинного перевода, предлагают для повышения качества работы языковых моделей задействовать в обучении и/или дообучении не одну языковую пару, а несколько. Такие мультиязыковые модели показывают лучшие языковые способности, нежели двуязычные [24].

Таким образом, наше исследование, предполагающее дообучение больших языковых моделей на параллельном корпусе трёх языков (английском, русском и испанском) для решения задачи мультиязыкового перевода в сфере информационной безопасности, демонстрирует свою актуальность.

1.3. Способы дообучения больших языковых моделей

Одним из самых распространённых способов дообучения нейронных сетей и, в частности, языковых моделей является перенос обучения, или transfer learning. Данная техника предполагает использование предварительно обученной модели в качестве основы с её последующим дообучением на новых данных. Метод переноса обучения позволяет сэкономить вычислительные ресурсы и время, так как модель уже обучена на крупном объеме данных [25].

Подвидом переноса обучения является тонкая настройка, или fine-tuning. Суть данного метода заключается в обучении только верхнего слоя нейронной сети для адаптации весов модели новому набору данных для решения определённой задачи. Данная техника часто применятся к трансформерным моделям семейств BERT [26] и GPT [27].

Что касается трансформерных моделей, имеющих одновременно кодирующий и декодирующие слои (например, T5), то к таким моделям, как правило, применяют одновременное обучение на решение нескольких задач (multi-task learning) [25].

Противоположной технике тонкой настройки является zero-shot learning (обучение без примеров), когда модели предлагают решать какую-либо задачу без дополнительного обучения на новом корпусе данных. Стоит также отметить методы few-shot и one-shot learning, предполагающие дообучение моделей искусственного интеллекта на ограниченном количестве примеров или на одном лишь примере соответственно [15].

С появлением больших языковых моделей, содержащих большое количество параметров, возникла потребность в адаптации не всех весов модели под решение специализированных задач, а лишь части из них. Данный подход получил название PEFT (parameter-efficient fine-tuning), т.е. метод тонкой настройки с эффективным использованием параметров [4]. На текущий момент можно выделить четыре вида PEFT:

1. Настройка адаптеров (adapter tuning), заключающаяся в добавлении дополнительных слоёв (адаптеров) в нейронную сеть. В ходе тонкой настройки модули адаптера оптимизируются под конкретные задачи, при этом параметры исходной языковой модели замораживаются [28].

2. Настройка префиксов (prefix tuning), представляющая собой добавление обучаемых непрерывных векторов (префиксов) к каждому слою нейронной сети для обучения модели новой задаче [29].

3. Настройка подсказок (prompt tuning), направленная на включение обучаемых векторов с подсказками во входной слой [30].

4. Низкоранговая адаптация (Low-Rank Adaptation, LoRA) – метод уменьшения количества параметров нейронной сети путем замены её полносвязных слоев на низкоранговые [18]. Стоит отдельно выделить и квантизованные низкоранговые адаптации (QLORA), которые предполагают

квантизацию предварительно обученных моделей до 4-битных версий, к которым затем добавляется небольшой набор обучаемых весов низкорангового адаптера [31].

Основываясь на гипотезе, что большие языковые модели хорошо обучаются на небольшом количестве примеров [15], для нашего исследования мы выбрали методы zero-shot и few-shot обучения. Кроме того, ввиду трудностей с проведением полного дообучения больших языковых моделей (непосредственно из-за их размеров), будем проводить PEFT дообучение, а именно квантизованную низкоранговую адаптацию (QLORA) отдельных параметров моделей.

1.4. Методы оценки языковых моделей

Оценка качества языковой модели является важным аспектом ее разработки и применения. Существует множество методов оценки работы языковых моделей, каждый из которых имеет свои преимущества и недостатки. В данном разделе мы рассмотрим некоторые из наиболее распространенных методов оценки.

Самыми базовыми метриками, которыми можно оценивать языковые модели, являются стандартные метрики классификации: precision, recall, accuracy и F1-мера. Использование данных метрик распространено в задачах классификации текстов, токенов, в закрытых вопросно-ответных системах и так далее [32]. Данные метрики высчитываются по следующим формулам:

$$Precision = \frac{TP}{TP + FP}, \tag{8}$$

$$Recall = \frac{TP}{TP + FN}, \tag{9}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (10)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \quad (11)$$

где TP – количество верных предсказаний положительной метки;

FP – количество неверных предсказаний положительной метки;

TN – количество верных предсказаний отрицательной метки;

FN – количество неверных предсказаний отрицательной метки.

Несмотря на довольно широкую распространённость данных метрик, они не могут оптимально оценить результаты работы моделей в задачах генерации текстов. Базовой метрикой для оценки данного рода задач является перплексия (perplexity). Данная метрика рассчитывает, насколько неожиданной для языковой модели является тестовая последовательность слов:

$$Perplexity = 2^{-\frac{\log_2 P(w_1, w_2, \ldots, w_N)}{N}}, \quad (12)$$

где N – количество слов в последовательности;

$w_1, w_2, \ldots, w_N$ – последовательность слов от 1 до N;

$P(w_1, w_2, \ldots, w_N)$ – вероятностное распределение последовательности слов от 1 до N.

Тем не менее, перплексия как метрика оценки слабо подходит для решения задач генерации текстов для специализированных задач [32]. В связи с этим, стали появляться более специфические методы оценки языковых моделей.

В 2001 году компанией IBM была предложена метрика BLEU (Bilingual Evaluation Understudy) для оценки качества машинного перевода. Данная

метрика высчитывает долю n-грамм переведенного текста, совпадающих с n-граммами эталонного текста, а также штрафует переводы, где длина итоговой последовательности значительно короче эталонной [33]:

$$BLEU = BP * e^{\left(\sum_{n=1}^{N}(w_n \times \log_2 P_n)\right)}, \quad (13)$$

$$BP = \begin{cases} 1, & c > r \\ e^{\left(1-\frac{r}{c}\right)}, & c \leq r \end{cases}, \quad (14)$$

где n – порядковый номер текущей n-граммы в последовательности;

N – количество n-грамм в последовательности;

c – длина переведённой n-граммы;

r – длина n-граммы эталонного перевода.

Для оценки задач, связанных с суммаризацей текстов, в 2004 году было разработано семейство метрик ROUGE (Recall-Oriented Understudy for Gisting Evaluation). Подобно BLUE, данные метрики также измеряют долю n-грамм сгенерированного текста, совпадающих с n-граммами эталонного ответа. Основное отличие ROUGE от BLUE заключается в том, что BLUE основана на метрике precision, в то время как ROUGE – на recall [34]. Самыми распространёнными метриками в семействе ROUGE являются ROUGE-N и ROUGE-L, которые рассчитываются по следующим формулам:

$$ROUGE\ N = \frac{CN}{TN}, \quad (15)$$

$$ROUGE\ S = \frac{LCS}{TN}, \quad (16)$$

где CN – количество сгенерированных n-грамм, совпадающих с эталонным текстом;

TN – общее количество n-грамм в эталонном тексте;

LCS – количество сгенерированных n-грамм, совпадающих с эталонным текстом, с учетом их порядка.

Ещё одной метрикой, предназначенной для оценки машинного перевода и генерации текста в целом, является METEOR (Metric for Evaluation of Translation with Explicit Ordering). Данная метрика основывается на precision, recall, но при этом также учитывает однокоренные слова и синонимы [35]. Формула для расчёта данной метрики следующая:

$$METEOR = \left(\frac{10P \times R}{9P + R}\right) \times \left(1 - (0.5 \times \frac{n\_chunks}{n\_unigrams\_matched})^3\right), \qquad (17)$$

где P – оценка precision;

R – оценка recall;

n_chunks – наименьшее число пар униграмм сгенерированного и эталонного текстов;

n_unigrams_matched – число униграмм сгенерированного текста, совпадающих с униграммами эталонного ответа.

Помимо прочего, оценивать языковые модели можно с помощью обратной связи от пользователей. Как правило, в анкетах обратной связи спрашивают об удовлетворённости пользователей ответами модели. Данный способ оценки распространён для оценки работы диалоговых систем [32].

С развитием искусственного интеллекта, исследователи стали адаптировать языковые модели для решения всё большего спектра задач, например, для извлечения именованных сущностей из биомедицинских данных, решения задач ЕГЭ, переводов в узконаправленных сферах и так далее. Для оценки работы моделей при решении таких задач были придуманы бенчмарки. Бенчмарки представляют собой открытые или доступные по

запросу наборы данных, составленные для решения специализированных задач [36].

Один из самых популярных бенчмарков для проверки работы языковых моделей – GLUE (General Language Understanding Evaluation). Он включает в себя датасеты для проверки англоязычных моделей в решении 9 различных задач:

1. CoLA: оценивает способности модели находить грамматические верные английские предложения. В качестве метрики оценки предлагается использовать коэффициент корреляции Мэтьюса.
2. SST-2: оценивает модель в решении задачи бинарной классификации тональности отзывов на фильмы.
3. MRPC: оценивает способности модели предсказывать, являются ли пары предложений эквивалентными семантически. Из-за несбалансированности классов в выборке, модели оцениваются по метрикам accuracy и F1-меры.
4. QQP: состоит из пар вопросов с сайта Quora, и, аналогично MRPC, проверяет модели на способность определять семантическую эквивалентность данных вопросов. Датасет также не сбалансирован, поэтому проверяется accuracy и F1-мера.
5. STS-B: состоит предложений на естественном языке, соотнесённых либо с заголовками новостей, либо с подписями к видео и изображениям. Каждая такая пара аннотирована оценкой схожести от 1 до 5. Задача модели состоит в том, чтобы предсказать данные оценки. Работа модели оценивается, используя коэффициенты корреляции Пирсона и Спирмена.
6. MNLI: оценивает способности языковой модели предсказывать по паре предложений, противоречит ли одно высказывание другому, вытекает ли второе высказывание из первого, либо оба предложения никак не соотносятся друг с другом.

7. QNLI: оценивает модель как вопросно-ответную систему на способность извлекать из абзаца с вопросом предложение с ответом на вопрос.

8. RTE: проверят модели на способность установления причинно-следственной связи между двумя предложениями.

9. WNLI: датасет на задачу поиска референта. Модели предлагается выбрать из списка референт для местоимения [37].

Бенчмарк GLUE был создан в 2018 году. Спустя год появилась его обновлённая версия – SuperGLUE [38]. Тем не менее, данные бенчмарки не подходили для оценки языковых моделей, обученных для решения задач на русском языке. В свзи с этим, в 2020 году появился бенчмарк RussianSuperGLUE [39].

Из минусов бенчмарков семейства GLUE стоит отметить отсутствие проверки работы моделей на задачах, связанных с генерацией текста. Поскольку в нашей работе мы исследуем дообучение моделей для решения подзадачи генерации текста – машинного перевода, рассмотрим, какие бенчмарки используются для оценки моделей в решении данной задачи.

Одним из таковых является FLORES-200. Он состоит из 3001 предложения, переведённого профессиональными переводчиками на 204 языка [24]. Безусловным плюсом данного бенчмарка является большое количество доступных для оценки языков. Тем не менее, для обучения модели какому-либо языку с нуля, такой датасет вряд ли подойдёт.

Создателями FLORES-200 затем была предпринята попытка увеличить размер выборки. Так появился NLLB Seed с 6 тысячами предложений из Википедии, переведённый на 39 языков. Авторы предлагают использовать данный датасет не сколько для оценки языковых моделей, сколько для обучения моделей распознавать языки или даже обучать модели языку [24]. Однако в данном датасете отсутствуют данные по двум языкам, которые мы определили как целевые: по русскому и испанскому.

В качестве метрик оценки качества перевода авторы данных бенчмарков предлагают использовать spBLEU и chrF++, поскольку они учитывают особенности языков агглютинативного и изолирующего типа языков [24].

Таким образом, изучив способы и метрики оценки работы языковых моделей для решения задач машинного перевода, а также учитывая тот факт, что целевые языки нашего исследования являются флективными, то было принято решение использовать метрику BLUE.

1.5. Выводы по Главе 1

В первой главе мы рассмотрели основные понятия языкового моделирования: языковая модель, n-граммы, кросс-энтропия, эмбеддинги, трансформеры. Также мы перечислили основные характеристики больших языковых моделей и обозначили их основные проблемы, одной из которых является слабая способность таких моделей решать специализированные задачи.

Отдельно мы провели обзор специализированных задач, которые можно решать с применением больших языковых моделей. Основываясь на актуальности задачи мультиязычного машинного перевода и отсутствием исследований, посвященных изучению способностей больших языковых моделей осуществлять перевод в сфере информационной безопасности, была выбрана специализированная задача для нашего исследования: мультиязычный машинный перевод для трёх языков (русского, английского и испанского) в сфере информационной безопасности.

В третьем подразделе данной главы были исследованы способы дообучения больших языковых моделей для решения специализированных задач. Основываясь на гипотезе, что большие языковые модели хорошо обучаются на небольшом количестве примеров, для нашего диссертационного исследования мы выбрали методы zero-shot и few-shot обучения. Кроме того,

ввиду трудностей с проведением полного дообучения больших языковых моделей (непосредственно из-за их размеров), будем проводить PEFT дообучение, а именно квантизованную низкоранговую адаптацию (QLORA) отдельных параметров моделей.

Помимо прочего, были рассмотрены основные способы оценки работы больших языковых моделей: от стандартных метрик до бенчмарков. Поскольку в нашей диссертационной работе в качестве специализированной задачи был выбран машинный перевод, а работать мы будем только с флективными языками, то для оценки способности переводить на русский, английский и испанский языки будет использоваться метрика BLUE.

В следующей главе мы приведём методологию исследования, а именно опишем набор данных, на котором будем проводить дообучение больших языковых моделей, сами модели, которые будем дообучать, а также непосредственно опишем алгоритм дообучения моделей.

Глава 2. Методология дообучения больших языковых моделей для решения специализированных задач

2.1. Описание набора данных для дообучения больших языковых моделей

Как уже отмечалось ранее, использование больших языковых моделей для перевода текстов информационной безопасности является мало разработанной темой, поэтому в качестве специализированной задачи, для которой мы будем осуществлять дообучение большой языковой модели, был выбран перевод в сфере информационной безопасности для шести языковых пар:

- английский-испанский,
- английский-русский,
- испанский-английский,
- испанский-русский,
- русский-английский,
- русский-испанский.

Выбор данных языковых пар обусловлен уверенным владением автора данными языками, а также плохой способностью больших языковых моделей переводить на русский язык без дополнительного обучения [40].

В ходе исследования был собран параллельный корпус из 1001 предложения на трёх языках: русском, английском и испанском (кастильском). Источниками для корпуса послужили сайты с официальной документацией 4 компаний, ведущих свою деятельность в сфере информационной безопасности:

1. Trellix (образованный в результате слияния McAfee Enterprise и FireEye): 250 параллельных предложений [41].

2. Dr. Web: 250 параллельных предложений [42].

3. Kaspersky: 250 параллельных предложений [43].

4. IBM: 251 параллельное предложение [44].

Данные компании были выбраны, так как на их сайтах представлена документация на целевых языках. Стоит также отметить, что в собранном корпусе намеренно представлены 2 компании, базирующихся в России (Dr. Web и Kaspersky), а значит вероятнее всего перевод документации осуществлялся с русского языка на прочие; и 2 американские компании (Trellix и IBM), где первым языком выступает английский. Испанских компаний из сферы информационной безопасности с доступной документацией одновременно на русском, английском и испанских языках обнаружено не было.

Над собранным корпусом предложений была произведена предварительная обработка данных: удалены дубликаты и произведена нормализация символов Юникода с помощью регулярных выражений. В очищенном датасете оказалось 988 троек предложений.

Далее, датасет был разбит на обучающую и тестовую часть (790 и 198 троек предложений соответственно).

Затем, для каждого предложения из тройки была сформирована инструкция «*Translate the following text from {source_language} to {target_language}\n\n### {source_language}: {text_in_source_language}\n\n### {target_language}:* » и ответ в виде эталонного перевода «*{text_in_target_language}*». Таким образом, для каждой тройки предложения было составлено 6 пар инструкция-перевод на каждой языковой паре. Всего в обучающей выборке оказалось 4740 предложений, а в тестовой – 1188.

Для подбора оптимальных гиперпараметров обучения дополнительно был проведён разведочный анализ данных. Важно отметить распределение количества слов в предложениях тренировочной (рисунок 1) и тестовой выборки (рисунок 2):

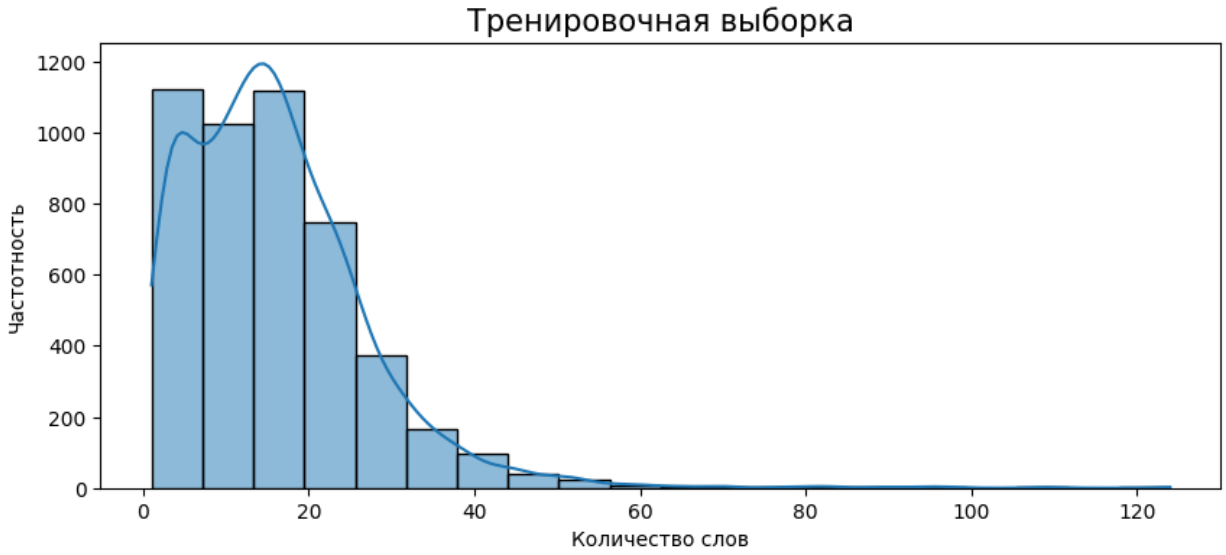

Рисунок 1 – Количество слов в тренировочной выборке

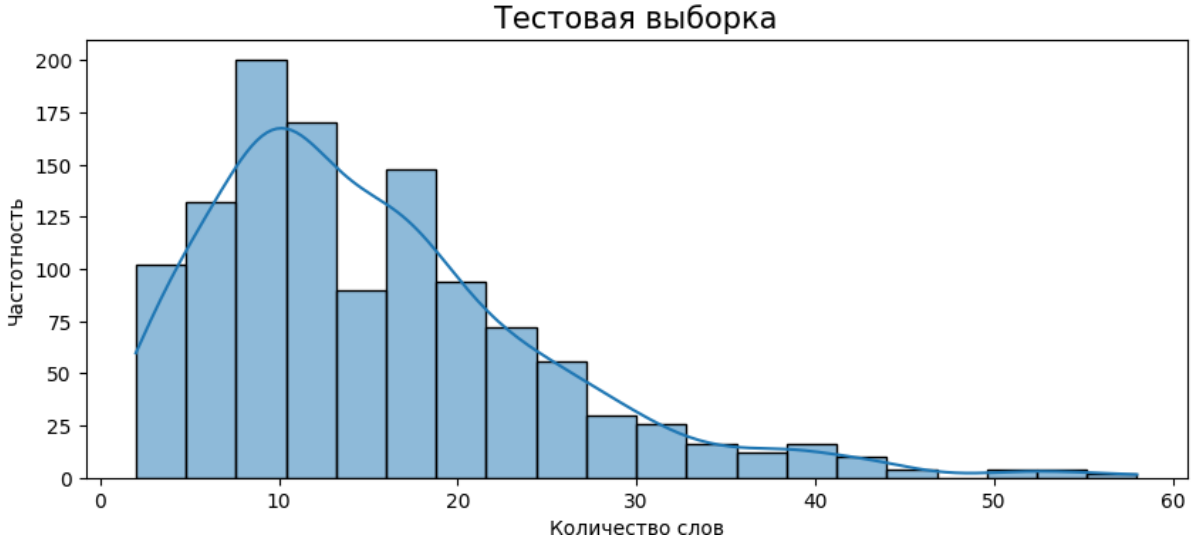

Рисунок 2 – Количество слов в тестовой выборке

Исходя из графика, можно сделать вывод, что в тренировочной выборке редко, но встречаются предложения длиной чуть больше 120 слов, в то время как максимальное количество слов в тестовой выборке составляет 60. Поскольку параметр длины генерируемой последовательности является довольно важным, то было принято решение сократить длинные предложения тестовой выборки до 60 слов. Данное преобразование претерпело 24 предложения из 4740. Распределение очищенной тренировочной выборки представлено на рисунке 3.

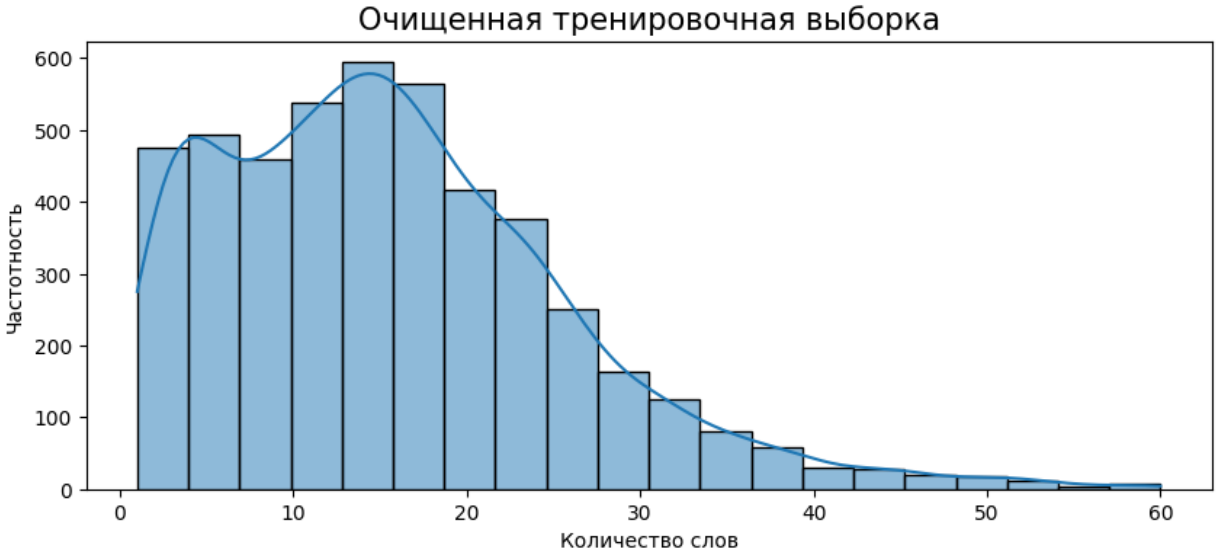

Рисунок 3 – Количество слов в очищенной тренировочной выборке

Таким образом, в данном подразделе мы описали процесс сбора и предварительной обработки датасета, который в дальнейшем будет использоваться при дообучении больших языковых моделей для решения специализированной задами мультиязычного машинного перевода.

2.2. Описание больших языковых моделей для дообучения

2.2.1. Mistral

Mistral – семейство больших языковых моделей, обладающих следующими архитектурными особенностями:

– механизм внимания скользящего окна (sliding window attention), ограничивающий количество токенов, которые «видит» токен с предыдущего слоя;

– скользящий буферный кэш (rolling buffer cache), позволяющий перезаписывать значения в кэше через определенное количество времени;

– предзаписанные значения в кэше, которые могут заполнять значениями из промпта;

– разделение на блоки (chunking), позволяющее применять механизм внимания к каждому кэшу и каждому блоку [14].

На момент проведения исследования, Mistral AI выпустила следующие модели:

1. Стандартные модели Mistral и Mixtral, предназначенные для решения задач генерации последующих токенов:

а) Mistral 7B (с 7 миллиардами параметров);

б) Mistral Small (исходный код закрыт, количество параметров не известно);

в) Mistral Medium (исходный код закрыт, количество параметров не известно);

г) Mistral Large (исходный код закрыт, количество параметров не известно);

д) Mixtral 8x7B (8 моделей-экспертов с 7 миллиардами параметров);

е) Mixtral 8x22B (8 моделей-экспертов с 22 миллиардами параметров).

2. Instruct-модели для генерации ответов по инструкциям пользователей:

а) Mistral 7B Instruct v0.1 (с 7 миллиардами параметров);

б) Mistral 7B Instruct v0.2 (с 7 миллиардами параметров);

в) Mixtral 8x7B Instruct (8 моделей-экспертов с 7 миллиардами параметров);

г) Mixtral 8x22B Instruct (8 моделей-экспертов с 22 миллиардами параметров).

Модели типа Mistral обучены преимущественно на английском языке, в то время как данные для обучения Mixtral были более сбалансированными для английского, французского, немецкого, испанского и итальянского языков [45].

Для нашего исследования, предполагающего выполнение инструкции о переводе с одного языка на другой, была выбрана модель Mistral 7B Instruct v0.2. Выбор v0.2, а не v0.1 обусловлен тем, что v0.2 является более новой и усовершенствованной версией v0.1. Основные параметры данной модели:

- размер слоя (dim): 4 096,
- количество слоёв (n_layers): 32,
- размер головы (head_dim): 128,
- размер скрытого состояния (hidden_dim): 14 336,
- количество голов внимания (n_heads): 32,
- количество голов ключ-значение (n_kv_heads): 8,
- размер скользящего окна (window_size): 4 096,
- размер контекста (context_len): 32 000,
- размер словаря (vocab_size): 32 000,
- базовый порядок для RoPE-эмбеддингов (rope_theta): 1 000 000.

Таким образом, в данном подпункте были описаны модели семейства Mistral и для проведения экспериментов с дообучением была выбрана Mistral 7B Instruct v0.2.

2.2.2. Llama 2

Llama 2 – семейство больших языковых моделей, главными архитектурными особенностями которых является:

– предварительная нормализация слоёв сети с помощью среднеквадратического значения (RMSNorm);

– использование функции SwiGLU для активации;

– добавление позиционных эмбеддингов вращения (Rotary Position Embeddings, RoPE) [46].

Модели Llama 2 представлены в следующих вариациях:

1. Стандартные Llama 2, предназначенные для решения задач генерации последующих токенов:
   а) Llama 2 7B (с 7 миллиардами параметров);
   б) Llama 2 13B (с 13 миллиардами параметров);
   в) Llama 2 34B (с 34 миллиардами параметров, в публичный доступ не выложена);
   г) Llama 2 70B (с 70 миллиардами параметров).
2. Llama 2-Chat, чьими основными задачами является предоставление ответов на запросы пользователей:
   а) Llama 2-Chat 7B (с 7 миллиардами параметров);
   б) Llama 2-Chat 13B (с 13 миллиардами параметров);
   в) Llama 2-Chat 70B (с 70 миллиардами параметров).

Что касается данных, на которых производилось обучение модели, то упоминается лишь то, что обучение производилось «на новом наборе общедоступных данных». Размер такого набора данных не уточняется, но

указывается, что данные представлены преимущественно на английском языке [15].

Поскольку для нашего исследования необходима способность модели выполнять запрос пользователя на перевод, а модели Llama 2-Chat 13B и Llama 2-Chat 70B имеют большой размер и долго дообучаются, то было принято решение остановиться на версии Llama 2-Chat 7B. Основные параметры данной модели:

- размер слоя (dim): 4 096,
- количество слоёв (n_layers): 32,
- количество голов внимания (n_heads): 32,
- размер словаря (vocab_size): 32 000,
- умножение на число после применения SwiGLU (multiple_of): 256,
- значение эпсилон для нормализации (norm_eps): 0,00001,
- базовый порядок для RoPE-эмбеддингов (rope_theta): 10 000.

Таким образом, в данном подпункте были описаны модели семейства Mistral и для проведения экспериментов с дообучением была выбрана Llama 2-Chat 70B.

2.3. Описание процесса дообучения больших языковых моделей для решения специализированных задач

2.3.1. Окружение развёртывания программного обеспечения

Запуск дообучения моделей проводился локально на компьютере, обладающим следующими характеристиками:

– центральный процессор: Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50GHz,

– оперативная память центрального процессора: 64 ГБ,

– количество ядер: 24,

– графический процессор: NVIDIA GeForce RTX 3090,

– оперативная память графического процессора: 24 ГБ,

– операционная система: Windows 11 Pro (22H2).

Для разработки программного кода использовался язык Python версии 3.11.1. Для проведения экспериментов были использованы следующие библиотеки данного языка программирования:

– основная библиотека для дообучения моделей: transformers (4.41.0.dev0),

– вспомогательная библиотека для дообучения моделей: torch (2.1.0+cu121),

– для ускорения обучения моделей: accelerate (0.26.1), peft (0.8.2), и flash_attn (2.4.1),

– для квантизации моделей: bitsandbytes (0.43.1),

– для низкоранговой адаптации моделей: loralib (0.1.2),

– для оценки работы моделей: evaluate (0.4.1),

– для сериализации данных: pickle (4.0),

– для проведения математических операций: numpy (1.26.4),

– для работы с табличными данными: pandas (2.2.1),

– для пред- и постобработки тестовых данных: re (2.2.1),

– для разбиения выборок: scikit-learn (1.4.0),

– для визуализации графиков: matplotlib (3.8.2) и seaborn (0.13.2),

– для визуализации прогресса: tqdm (4.66.1).

Далее опишем сам процесс дообучения, в зависимости от режима дообучения.

2.3.2. Загрузка датасета для дообучения

На данном этапе загружался ранее собранный датасет для решения специализированной задачи: перевод в сфере информационной безопасности для шести языковых пар.

В зависимости от выбранного режима загружаются либо обе выборки (обучающая и тренировочная), либо только тестовая. Так, для дообучения в режимах zero-shot и few-shot загружалась лишь тестовая выборка, а для PEFT – обе.

2.3.3. Предобработка датасета для дообучения

Для каждого режима дообучения использовался свой вариант для предобработки:

– для zero-shot-режима предобработка ограничивалась в написании инструкции для модели, например: «*Translate the following text from {source_language} to {target_language}\n\n### {source_language}: {text_in_source_language}\n\n### {target_language}:* »;

– для few-shot-режимов в инструкцию также включались примеры эталонного ответа (один для One-shot, два для Two-shot и три для Three-shot). Пример для One-shot: «*Translate the following text from {source_language} to {target_language}\n\n### {source_language}: {text1_in_source_language}\n\n### {target_language}: {translation1}\n\n### {source_language}: {text2_in_source_language}\n\n### {target_language}:* »;

– для PEFT режимов предобработка проводилась только над обучающей выборкой и включала в себя формирование промпта, состоящего из инструкции, предложения на исходном языке и эталонного варианта перевода, например: «*Translate the following text from {source_language} to {target_language}\n\n### {source_language}: {text_in_source_language}\n\n### {target_language}: {translation}*».

2.3.4. Загрузка и квантизация базовой модели

Загрузка модели Mistral-7B-Instruct-v0.2 как объекта класса AutoModelForCausalLM из библиотеки transformers происходит без затруднений при наличии доступа к сети Интернет и 15 ГБ свободного места на диске.

Что касается моделей семейства Llama, то для их использования с помощью библиотеки transformers необходимо предварительно запросить доступ к данным моделям. Для этого нужно:

– заполнить анкету о себе и согласиться с условиями использования модели на её официальном сайте;

– получить разрешение об использовании модели в виде электронного письма;

– сгенерировать токен доступа к своему аккаунту на Hugging Face (адрес электронной почты для авторизации на Hugging Face должен совпадать с тем, на который запрашивался доступ к модели);

– заполнить анкету о себе и согласиться с условиями использования модели на её страничке на Hugging Face;

– авторизоваться на Hugging Face через командную строку, используя токен доступа.

Квантизация моделей до 4 бит так же производилась при помощи библиотеки transformers, в которую интегрированы (но требуют отдельной установки) классы и методы библиотеки bitsandbytes.

2.3.5. Токенизация текстов

Токенизация используемых текстов проводилась при помощи объекта класса AutoTokenizer библиотеки transformers с указанием следующих атрибутов:

– ссылка или путь на базовую модель (pretrained_model_name_or_path);

– сторона, с которой происходит заполнение вспомогательными токенами (padding_side): левая (left), поскольку в исследовании используются только декодерные вариации трансформерных моделей;

– добавление токена начала последовательности (add_bos_token): верно (True), чтобы избежать дополнительных пред- и постобработок результатов генерации;

– добавление токена конца последовательности (add_eos_token): верно (True) по аналогичной причине, указанной в предыдущем атрибуте;

– максимальная длина последовательности токенов, обрабатываемая моделью (model_max_length): 256.

Последний параметр был выведен на основе анализа распределения токенов в предложениях, который показал для решения нашей специализированной генеративной задачи модели достаточно 256 токенов. С распределением можно ознакомиться на рисунке 4.

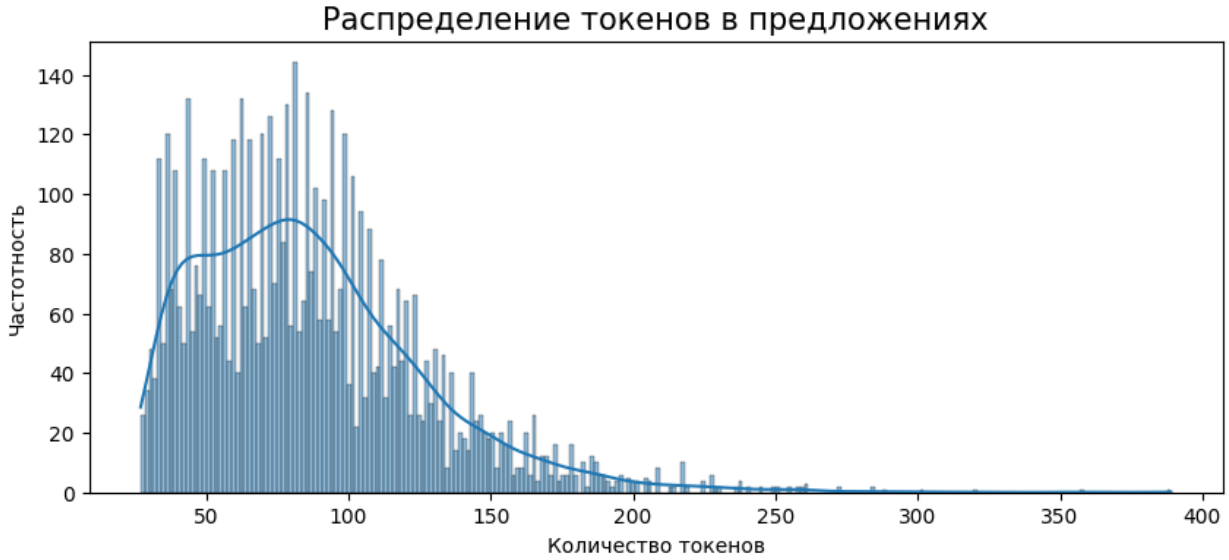

Рисунок 4 – Распределение токенов в предложениях

2.3.6. Настройка низкоранговой адаптации

Через данный этап проходили лишь те модели, для которых проводилось дообучение в формате PEFT. На этом этапе задавались параметры низкоранговой адаптации для квантизованных моделей, как то:

– ранг дообученной матрицы (r): 32;

– коэффициент масштабирования (lora_alpha): 64;

– смещение (bias): нет (None);

– коэффициент прореживания (lora_dropout): 0,05;

– тип задания (task_type): условное языковое моделирование (CAUSAL_LM);

– список линейных слоёв, к которым будут прикрепляться низкоранговые матрицы (target_modules):

а) матрица проекции запросов (q_proj),

б) матрица проекции ключей (k_proj),

в) матрица проекции значений (v_proj),

г) матрица проекции выхода многоголового внимания (o_proj),

д) матрица проекции первого слоя многослойного перцептрона (gate_proj),

е) матрица проекции второго слоя многослойного перцептрона (up_proj),

ж) матрица проекции третьего слоя многослойного перцептрона (down_proj),

з) матрица проекции последнего слоя, головы (lm_head).

2.3.7. Настройка ускорителя

Аналогично предыдущему шагу, через данный этап проходят только модели, дообучаемые в режиме PEFT. На этом этапе импортируется

библиотека accelerate, которая позволяет совершать оптимизированные высокопроизводительные вычисления. В данной библиотеке стоит отметить объект FullyShardedDataParallelPlugin, который для используется как обёртка над базовыми моделями для оптимизации их вычислений. В целом, данный этап дообучения является опциональным, но в нашем исследовании важен для ускорения вычислений, проводимых при дообучении больших языковых моделей.

2.3.8. Обучение модели на тренировочной выборке.

Так же как этапы настройки низкоранговой адаптации и настройки ускорителя, данный этап дообучения актуален только для метода PEFT.

Поскольку в нашем исследовании дообучение моделей производилось преимущественно при помощи библиотеки transformers, то важно отметить атрибуты, используемые при создании экземпляра класса TrainingArguments:

- название директории, куда будут сохраняться контрольные точки моделей (output_dir);
- количество примеров тренировочной выборки в партии (per_device_train_batch_size): 2;
- количество шагов, после которых осуществлять обратный проход (gradient_accumulation_steps): 1;
- использовать контрольные точки при расчёте градиента (gradient_checkpointing): верно (True);
- максимальное количество шагов (max_steps): 500;
- скорость обучения модели (learning_rate): 2.5e-5;
- использовать точность в 16-битных числах (bf16): верно (True), для ускорения вычислений;

– оптимизатор (optim): paged_adamw_8bit, для избежания утечек памяти,

– логирование результатов функции потери через n-шагов (logging_steps): 25;

– название директории для логирования (logging_dir);

– как делать контрольную точку (save_strategy): через каждый шаг (steps);

– когда делать контрольную точку (save_steps): 25;

– стратегия валидации модели (evaluation_strategy): через каждый шаг (steps);

– интервал для проведения валидации (eval_steps): 25;

– проводить валидацию (do_eval): верно (True).

2.3.9. Генерация ответов на тестовой выборке

Данный этап применялся ко всем моделям, вне зависимости от выбранного режима обучения. Для генерации ответов использовались встроенные методы инициализированных ранее объектов модели и токенизатора.

На генерацию ответов накладывались следующие ограничения:

– максимальное количество новых токенов (max_new_tokens): 60, поскольку именно данным числом мы ограничили тренировочную и тестовую выборку при создании датасета;

– штраф за повторы (repetition_penalty): 1,15, чтобы модель не уходила в бесконечный цикл генерации ответов;

– пропуск специальных токенов (skip_special_tokens): верно (True), так как при декодировании в них нет необходимости.

2.3.10. Постобработка результатов генерации модели

При анализе результатов работы дообученных моделей, стал очевиден тот факт, данные языковые модели (Mistral-7B-Instruct-v0.2 и Llama-2-7b-chat-hf) при генерации ответа вставляют в начало переданный им промпт. Для проведения адекватной оценки метрикой BLEU, данные предложения удалялись из выборки. Также, модель Mistral в некоторых случаях добавляла в начало ответа символ ⁂, который было решено удалить при помощи библиотеки для работы с регулярными выражениями re.

2.3.11. Оценка работы модели

Как уже упоминалось ранее, в качестве метрики оценивания работы дообученных моделей была выбрана метрика BLUE, поскольку она используется в качестве универсальной при оценке переводов европейских языков.

Метрика BLEU, реализованная в библиотеке transformers, представляет из себя совокупность из пяти оценок:

1) BLEU-1 высчитывает долю совпадений униграмм переведенного и эталонного текста, с учётом штрафа за короткие тексты;

2) BLEU-2 поступает аналогичным образом, только с биграммами;

3) BLEU-3 – с триграммами;

4) BLEU-4 – с четыреграммами;

5) BLEU-score (или BLEU-S) – высчитывает одновременно совпадения для уни-, би-, три- и четырёхграмм.

Стоит отметить, что значение штрафа за длину в каждой разновидности высчитывается индивидуально. Более подробно с методикой расчёта данной оценки можно ознакомиться в Главе 1.

Метрики BLEU, основанные на метрике precision, принимают значения от 0 до 1, где 1 означает полное совпадение с эталонным переводом. Однако на практике, значения BLEU равные единице маловероятны, поскольку в процессе перевода может отличаться порядок слов или использоваться синонимы. Так, в оригинальной статье, где была предложена данная метрика, её значения при совместном оценивании уни-, би-, три- и четырёхграмм (т.е. BLEU-S) варьировались от 0,05 до 0,26 [33], в то же время трансформерным моделям при таком же оценивании n-грамм удалось добиться значений 0,28 при переводе с английского на немецкий язык и 0,41 при переводе с английского на французский [11].

Таким образом, при оценке работы наших моделей на решение задач машинного перевода, мы также будем использовать уни-, би-, три- и четырёхграммы, а как оценку хорошего перевода будем принимать значения BLEU около 0,25.

Резюмируя основные положения пункта, то предложенный нами алгоритм дообучения больших языковых состоит из следующих шагов: загрузка датасета для дообучения, его предобработка, подготовка модели для обучения, токенизация текстов, генерация ответов, постобработка сгенерированных текстов и оценка работы модели в целом. Однако, для каждого режима дообучения имеются свои особенности, и для дообучения при помощи PEFT дополнительно добавляются этапы настройки низкоранговой адаптации, ускорителя обучения и непосредственного обучения на тренировочной выборке.

2.4. Выводы по Главе 2

В качестве специализированной задачи, для решения которой осуществлялось дообучение больших языковых моделей, был выбран машинный мультиязычный перевод в сфере информационной безопасности. В

качестве языков, с и на которые производился перевод, были выбраны английский, испанский и русский.

Чтобы осуществить дообучение моделей для решения данной задачи, был создан специализированный датасет. Датасет был собран из официальной документации четырёх компаний, занимающихся информационной безопасностью. Датасет содержит 1001 тройку (или 3003 пар) параллельных предложений на русском, английском и испанском языках.

В качестве моделей, которые обучались для решения специализированной задачи мультиязычного машинного перевода, были выбраны Mistral и Llama 2 в модификациях Mistral-7B-Instruct-v0.2 и Llama-2-7b-chat-hf соответственно. Данные модели были выбраны поскольку они:

– выложены в открытый доступ;

– имеют одинаковое количество параметров;

– демонстрируют SOTA-результаты в решении общих задач из области обработки естественного языка;

– могут быть запущены локально на одном устройстве с одной видеокартой.

Дообучение выбранных моделей происходило в нескольких режимах: zero-shot, few-shot (one-shot, two-shot и three-shot), а также PEFT.

Резюмировать алгоритм дообучения, в зависимости от режима дообучения, можно в виде схемы, представленной на рисунке 5.

В следующей главе будут представлены результаты дообучения больших языковых моделей для решения специализированной задачи мультиязычного машинного перевода.

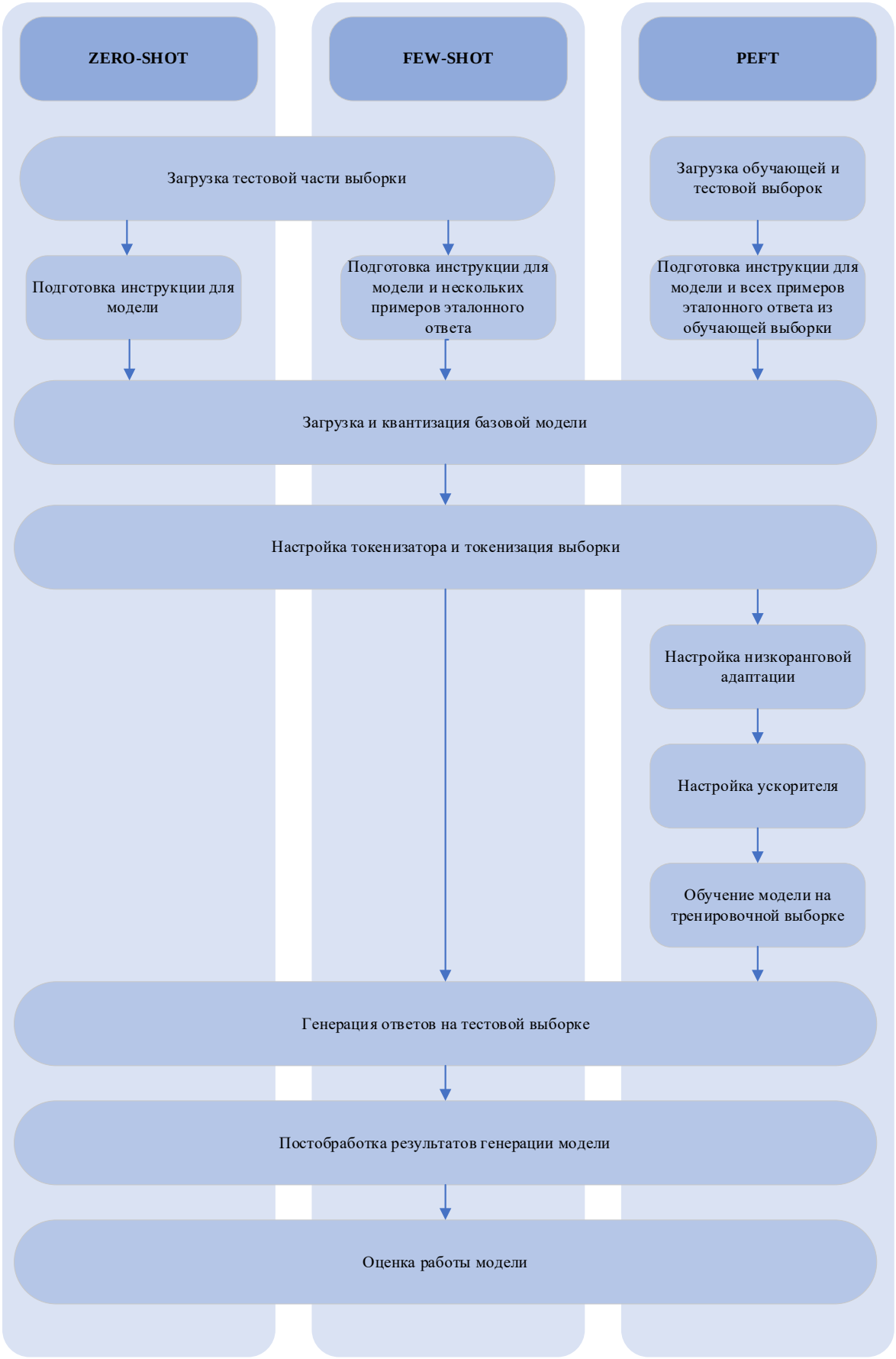

Рисунок 5 – Алгоритм дообучения больших языковых моделей

Глава 3. Результаты дообучения больших языковых моделей

3.1. Zero-shot обучение

Напомним, что для дообучения в режиме zero-shot мы передавали модели лишь инструкцию с просьбой перевести с одного языка на другой для шести языковых пар.

Пример для пары переводов с английского на испанский: «*Translate the following text from English to Spanish\n\n### English:\nIn addition, Trojans can also change the start page in a web browser or delete certain files.\n\n### Spanish:\n*».

Результаты дообучения модели Mistral в таком режиме в доверительных интервалах 95% представлены в таблице 1, а результаты для Llama 2 – в таблице 2.

Таблица 1 – Оценки BLEU при zero-shot дообучении Mistral Instruct 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,39 ± 0,03	0,21 ± 0,03	0,41 ± 0,04	0,18 ± 0,02	0,36 ± 0,03	0,28 ± 0,03	0,31 ± 0,01
BLEU-2	0,22 ± 0,03	0,09 ± 0,02	0,25 ± 0,03	0,07 ± 0,02	0,19 ± 0,02	0,14 ± 0,02	0,16 ± 0,01
BLEU-3	0,15 ± 0,02	0,06 ± 0,02	0,18 ± 0,03	0,04 ± 0,01	0,12 ± 0,02	0,09 ± 0,02	0,11 ± 0,01
BLEU-4	0,10 ± 0,02	0,04 ± 0,02	0,13 ± 0,03	0,03 ± 0,01	0,08 ± 0,02	0,05 ± 0,01	0,07 ± 0,01
BLEU-S	0,15 ± 0,03	0,05 ± 0,02	0,18 ± 0,03	0,04 ± 0,01	0,13 ± 0,02	0,08 ± 0,02	0,10 ± 0,01

Таблица 2 – Оценки BLEU при zero-shot дообучении Llama 2 Chat 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,30 ± 0,03	0,11 ± 0,02	0,32 ± 0,03	0,12 ± 0,02	0,22 ± 0,02	0,21 ± 0,02	0,22 ± 0,01
BLEU-2	0,17 ± 0,02	0,04 ± 0,01	0,17 ± 0,03	0,04 ± 0,01	0,10 ± 0,02	0,10 ± 0,02	0,11 ± 0,01
BLEU-3	0,11 ± 0,02	0,02 ± 0,01	0,12 ± 0,02	0,03 ± 0,01	0,06 ± 0,01	0,07 ± 0,01	0,07 ± 0,01
BLEU-4	0,07 ± 0,02	0,01 ± 0,01	0,08 ± 0,02	0,02 ± 0,01	0,04 ± 0,01	0,04 ± 0,01	0,04 ± 0,01
BLEU-S	0,11 ± 0,02	0,02 ± 0,01	0,13 ± 0,02	0,03 ± 0,01	0,06 ± 0,01	0,07 ± 0,02	0,07 ± 0,01

Сравнивая оценки BLEU, полученные моделями при дообучении в zero-shot режиме, можно отметить, что Mistral показала более высокие результаты в решении специализированной задачи мультиязычного перевода в сфере информационной безопасности. Также стоит отметить, что обеим моделям с трудом удаётся выполнять переводы на русский язык. Примечательно, что обе модели достигают наибольшего качество при переводах с испанского на английский язык. Данные факты подтверждаются тем, что именно английский язык преобладал в исходном датасете для претрейна обеих моделей [15, 45].

Проверим статистическую значимость различия полученных оценок BLEU для Mistral и Llama 2. Для выбора статистического теста, нам необходимо провести проверку на нормальность распределения получившихся величин. На рисунке 6 приведено распределение для Mistral Instruct 7B, а на рисунке 7 – для Llama 2.

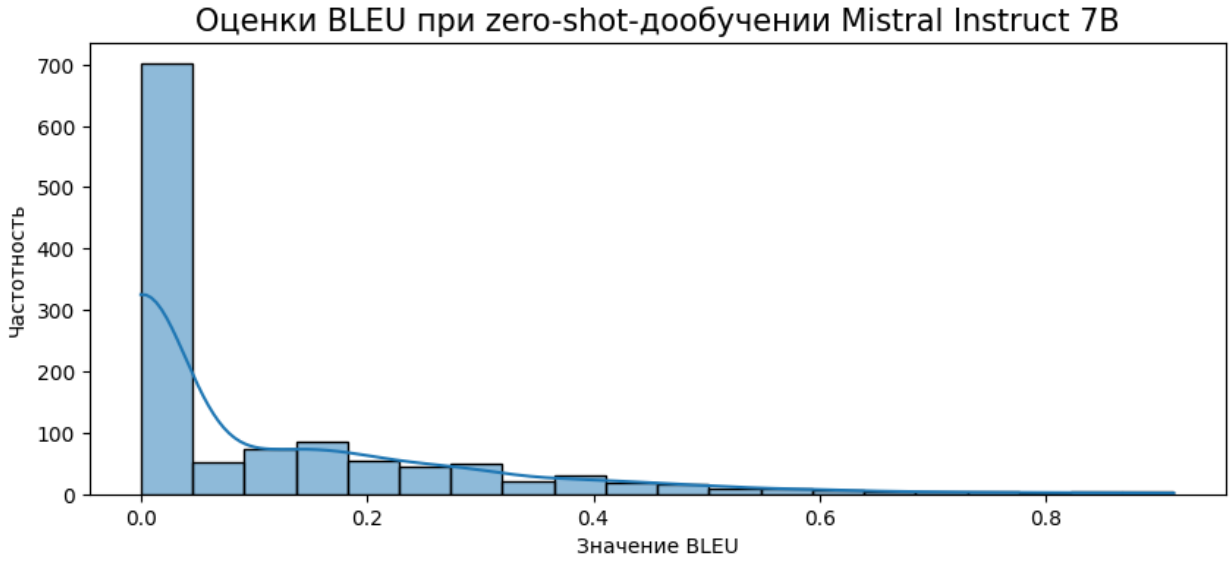

Рисунок 6 – Распределение оценок BLEU при zero-shot дообучении Mistral Instruct 7B

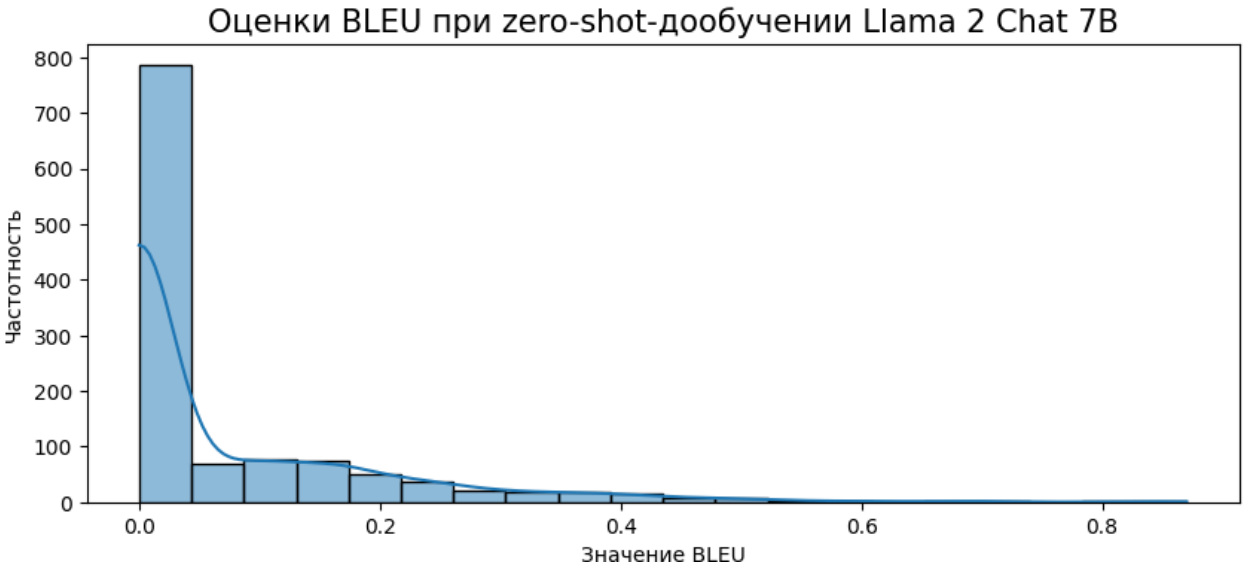

Рисунок 7 – Распределение оценок BLEU при zero-shot дообучении Llama 2 Chat 7B

Для проверки на нормальность распределения полученных результатов, использовался критерий согласия Пирсона при уровне значимости 0,05. Результаты тестов представлены в таблице 3.

Таблица 3 – Критерий согласия Пирсона для оценок моделей, дообученных в zero-shot режиме

Модель	Статистика	p-value
Mistral Instruct 7B	442,84	0,0
Llama 2 Chat 7B	574,17	0,0

Результаты обоих тестов показали, что на уровне значимости 0,05 можно отклонить гипотезу о том, что распределения оценок BLEU при zero-shot дообучении как для Mistral Instruct 7B, так и для Llama 2 Chat 7B подчиняются нормальному распределению, а следовательно, для проведения тестов на значимость различия результатов дообучения будем использовать непараметрический U-тест Манна-Уитни.

U-тест Манна-Уитни на уровне значимости 0,05 показал, что можно отклонить гипотезу равенстве распределений оценок BLEU при zero-shot дообучении двух моделей: статистика равна 7 265, а p-value – 0,00007.

Также отметим, что значения BLEU, полученные в данном режиме дообучения мы будем использовать в качестве baseline для проведения дальнейших экспериментов по улучшению качества мультиязычных переводов в сфере информационной безопасности.

3.2. Few-shot обучение

Few-shot обучение предполагает добавление в промпт модели помимо самой инструкции-задания ещё и несколько эталонных примеров выполнения такого задания. Количество таких примеров может разниться. Так, например, в нашей задаче машинного перевода при режиме one-shot модели передаётся один пример эталонного перевода, при режиме two-shot – два и при three-shot – три. Далее рассмотрим, как обучались модели в рамках этих трёх сценариев.

3.2.1. One-shot обучение

В режиме one-shot моделям передавалась инструкция с просьбой перевести с одного языка на другой, её эталонный перевод и ещё одна инструкция-задание, ответ на которую модель и должна сделать перевод. Для каждой из шести языковых пар, составлялись соответствующие инструкции.

Пример для пары переводов с испанского на русский: «*Translate the following text from Spanish to Russian\n\n### Spanish:\nAdemás, puede pulsar el botón Acerca de para ver la información de su licencia.\n\n### Russian:\nКроме этого, нажав О программе на этой же странице, вы также можете просмотреть информацию о вашей лицензии.\n\n### Spanish:\nPara guardar los ajustes, seleccione el Perfil estándar y en el menú que aparece seleccione la opción Guardar.\n\n### Russian:\n*».

Результаты дообучения моделей Mistral Instruct 7B и Llama 2 Chat 7B в one-shot режиме в доверительных интервалах 95% представлены в таблицах 4 и 5 соответственно.

Таблица 4 – Оценки BLEU при one-shot дообучении Mistral Instruct 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,58 ± 0,03	0,40 ± 0,03	0,65 ± 0,03	0,43 ± 0,03	0,53 ± 0,03	0,52 ± 0,03	0,52 ± 0,01
BLEU-2	0,37 ± 0,03	0,20 ± 0,03	0,42 ± 0,04	0,22 ± 0,03	0,31 ± 0,03	0,29 ± 0,03	0,30 ± 0,01
BLEU-3	0,26 ± 0,03	0,12 ± 0,02	0,29 ± 0,04	0,13 ± 0,02	0,20 ± 0,03	0,19 ± 0,03	0,20 ± 0,01
BLEU-4	0,18 ± 0,03	0,08 ± 0,02	0,22 ± 0,04	0,08 ± 0,02	0,13 ± 0,02	0,13 ± 0,02	0,13 ± 0,01
BLEU-S	0,24 ± 0,03	0,11 ± 0,03	0,30 ± 0,04	0,12 ± 0,03	0,20 ± 0,03	0,19 ± 0,03	0,19 ± 0,01

Таблица 5 – Оценки BLEU при one-shot дообучении Llama 2 Chat 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,34 ± 0,03	0,22 ± 0,03	0,40 ± 0,04	0,16 ± 0,02	0,34 ± 0,04	0,27 ± 0,03	0,29 ± 0,01
BLEU-2	0,20 ± 0,03	0,09 ± 0,02	0,24 ± 0,03	0,06 ± 0,02	0,17 ± 0,03	0,12 ± 0,02	0,15 ± 0,01
BLEU-3	0,13 ± 0,02	0,06 ± 0,02	0,17 ± 0,03	0,04 ± 0,01	0,10 ± 0,02	0,07 ± 0,02	0,09 ± 0,01
BLEU-4	0,09 ± 0,02	0,03 ± 0,01	0,12 ± 0,03	0,02 ± 0,01	0,07 ± 0,02	0,04 ± 0,01	0,06 ± 0,01
BLEU-S	0,13 ± 0,02	0,05 ± 0,02	0,17 ± 0,03	0,04 ± 0,01	0,11 ± 0,02	0,07 ± 0,02	0,09 ± 0,01

При анализе результатов дообучения моделей в one-shot режиме, можно сделать аналогичные zero-shot дообучению выводы:

1) в среднем более высокие результаты у Mistral, нежели чем у Llama 2;

2) хуже всего обе модели справляются с переводами на русский язык;

3) обе модели достигают наилучших результатов при переводах с испанского на английский язык.

Распределения оценок для обеих моделей в режиме one-shot несколько отличаются от zero-shot. Распределение для Mistral представлено на рисунке 8, а для Llama 2 – на рисунке 9.

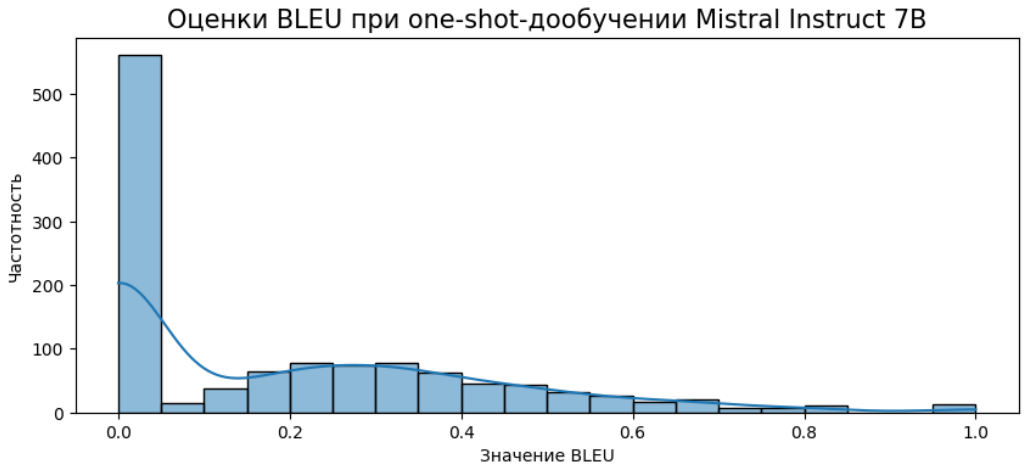

Рисунок 8 – Распределение оценок BLEU при one-shot дообучении Mistral Instruct 7B

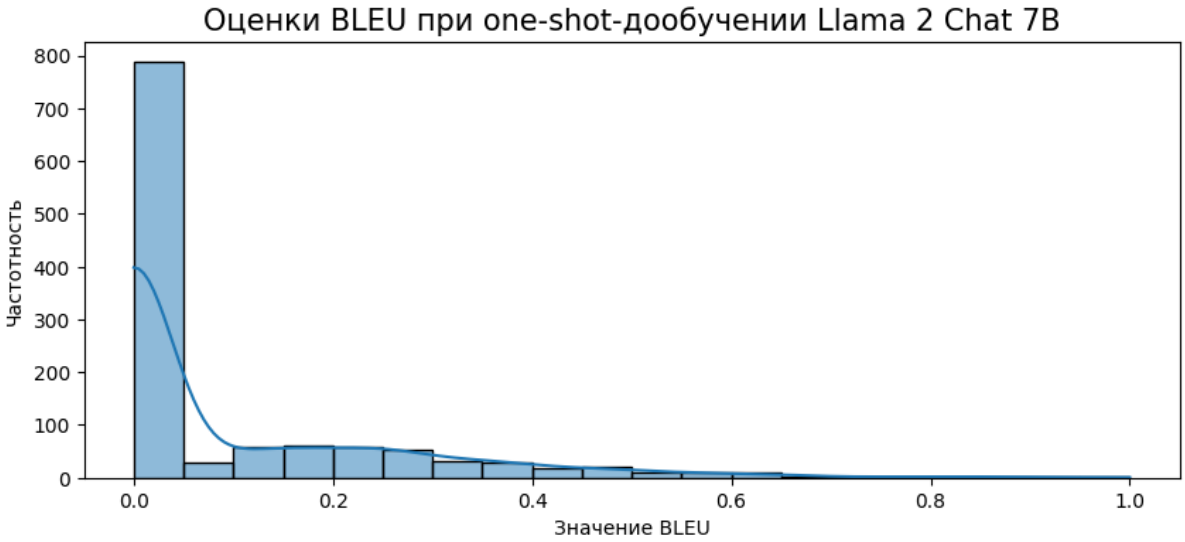

Рисунок 9 – Распределение оценок BLEU при one-shot дообучении Llama 2 Chat 7B

Для проверки гипотезы о соответствии распределения полученных оценок нормальному распределению использовался критерий согласия Пирсона. Результаты данного статистического теста представлены в таблице 6.

Таблица 6 – Критерий согласия Пирсона для оценок моделей, дообученных в one-shot режиме

Модель	Статистика	p-value
Mistral Instruct 7B	176,22	0,0
Llama 2 Chat 7B	464,04	0,0

Таким образом, критерий согласия Пирсона на уровне значимости 0,05 показал, что можно отклонить гипотезу о соответствии полученных распределений нормальному. Поэтому, для проверки следующих гипотез о статистически значимых различиях распределений, будет использоваться критерий Манна-Уитни. Результаты данных тестов представлены в таблице 7.

Таблица 7 – U-критерий Манна-Уитни для оценок моделей, дообученных в one-shot режиме

Сравниваемые модели	Статистика	p-value
Mistral Instruct 7B (one-shot) и Llama 2 Chat 7B (one-shot)	8 649,00	0,000
Mistral Instruct 7B (one-shot) и Mistral Instruct 7B (zero-shot)	8 348,50	0,000
Llama 2 Chat 7B (one-shot) и Llama 2 Chat 7B (zero-shot)	6 827,00	0,003

Интерпретируя результаты таблицы 7, можно сказать, что три U-теста Манна-Уитни на уровне значимости 0,05 показали, что можно отклонить гипотезу о равенстве распределений оценок BLEU и принять альтернативную гипотезу о статистически значимой разнице между распределениями оценок всех трёх пар.

Таким образом, доказывается эффективность передачи в промпт модели, помимо самой инструкции на выполнение специализированной задачи, эталонного примера выполнения специализированной задачи. В следующем подпункте мы проверим гипотезу о передачи двух эталонных примеров в промпт.

3.2.2. Two-shot обучение

В режиме two-shot для каждой языковой пары были сформированы инструкции с заданием на перевод, а также в промпт были дополнительно добавлены по два примера с эталонными переводами, советующими языковой паре.

Пример сформированной инструкции для перевода с русского языка на английский: «*Translate the following text from Russian to English\n\n### Russian:\nУбедитесь, что все конечные точки соответствуют минимальным требованиям (KB82761).\n\n### English:\nVerify that all endpoints meet the minimum requirements (KB82761).\n\n### Russian:\nТроянские программы подменяют какую-либо из часто запускаемых программ и выполняют ее*

функции (или имитируют исполнение этих функций), одновременно производя какие-либо вредоносные действия (повреждение и удаление данных, пересылка конфиденциальной информации и т.д.), либо делая возможным несанкционированное использование компьютера злоумышленником, например, для нанесения вреда третьим лицам.\n\n### English:\nTrojans substitute a frequently-used program and perform its functions (or imitate its operation). Meanwhile, they perform some malicious actions in the system (damages or deletes data, sends confidential information, etc.) or make it possible for hackers to access the computer without permission, for example, to harm the computer of a third party.\n\n### Russian:\nПродукт Informix поддерживает базы данных ANSI и совместим с промышленными стандартами для языка SQL.\n\n### English:\n».

Ниже, в таблице 8, представлены результаты дообучения модели Mistral в two-shot режиме в 95% доверительном интервале. Аналогичные результаты для Llama 2 можно увидеть в таблице 9.

Таблица 8 – Оценки BLEU при two-shot дообучении Mistral Instruct 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,59 ± 0,03	0,41 ± 0,03	0,67 ± 0,03	0,44 ± 0,03	0,55 ± 0,03	0,53 ± 0,03	0,53 ± 0,01
BLEU-2	0,38 ± 0,03	0,21 ± 0,03	0,44 ± 0,04	0,22 ± 0,03	0,32 ± 0,03	0,30 ± 0,03	0,32 ± 0,01
BLEU-3	0,27 ± 0,03	0,13 ± 0,03	0,32 ± 0,04	0,14 ± 0,03	0,21 ± 0,03	0,20 ± 0,03	0,21 ± 0,01
BLEU-4	0,18 ± 0,03	0,09 ± 0,02	0,24 ± 0,04	0,09 ± 0,02	0,13 ± 0,02	0,14 ± 0,02	0,14 ± 0,01
BLEU-S	0,25 ± 0,03	0,12 ± 0,03	0,32 ± 0,04	0,13 ± 0,03	0,21 ± 0,03	0,20 ± 0,03	0,21 ± 0,01

Таблица 9 – Оценки BLEU при two-shot дообучении Llama 2 Chat 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,52 ± 0,03	0,30 ± 0,03	0,51 ± 0,03	0,24 ± 0,03	0,41 ± 0,03	0,33 ± 0,03	0,38 ± 0,01
BLEU-2	0,31 ± 0,03	0,14 ± 0,03	0,32 ± 0,03	0,10 ± 0,02	0,22 ± 0,03	0,14 ± 0,03	0,20 ± 0,01
BLEU-3	0,21 ± 0,03	0,08 ± 0,02	0,22 ± 0,03	0,06 ± 0,02	0,14 ± 0,02	0,08 ± 0,02	0,13 ± 0,01
BLEU-4	0,14 ± 0,03	0,05 ± 0,02	0,16 ± 0,03	0,04 ± 0,01	0,09 ± 0,02	0,05 ± 0,01	0,09 ± 0,01
BLEU-S	0,20 ± 0,03	0,07 ± 0,02	0,22 ± 0,03	0,06 ± 0,02	0,14 ± 0,03	0,08 ± 0,02	0,13 ± 0,01

Первое, на что стоит обратить внимание при сравнении результатов Mistral Instruct 7B и Llama 2 Chat 7B в режиме two-shot дообучения, – это более высокие значения BLEU у Mistral в целом, нежели у Llama 2.

Тем не менее, стоит также отметить уменьшение разрыва в оценках качества моделей для переводов с английского на испанский языки. Произошло это преимущественно благодаря значительному росту метрик Llama 2, что говорит об эффективности стратегии увеличения количества эталонных примеров в промпте модели.

Что касается результатов Mistral при дообучении в данном режиме, то стоит заметить, что добавление второго примера привело к увеличению оценок BLEU лишь на 0,1-0,2 пункта по сравнению с one-shot дообучением. Далее проведём статистические тесты на проверку статистической значимости различий распределений шести полученных моделей (Mistral и Llama 2 в режимах zero-shot, one-shot и two-shot).

С помощью критерия согласия Пирсона были проведены тесты на статистическую значимость различий полученных распределений оценок BLEU с нормальным распределением, которые показали, что на уровне значимости 0,05 гипотезы о соответствии данных распределений нормальному следует отклонить. Результаты тестов представлены в таблице 10.

Таблица 10 – Критерий согласия Пирсона для оценок моделей, дообученных в two-shot режиме

Модель	Статистика	p-value
Mistral Instruct 7B	178,83	0,0
Llama 2 Chat 7B	291,90	0,0

Сами распределения оценок BLEU для модели Mistral Instruct 7B представлен на рисунке 10, а для Llama 2 Chat 7B – на рисунке 11.

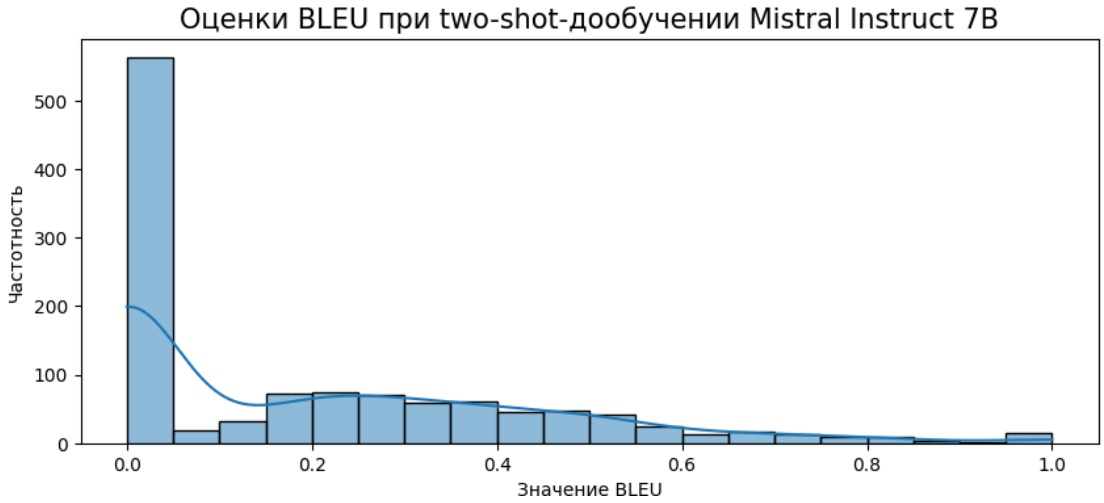

Рисунок 10 – Распределение оценок BLEU при two-shot дообучении Mistral Instruct 7B

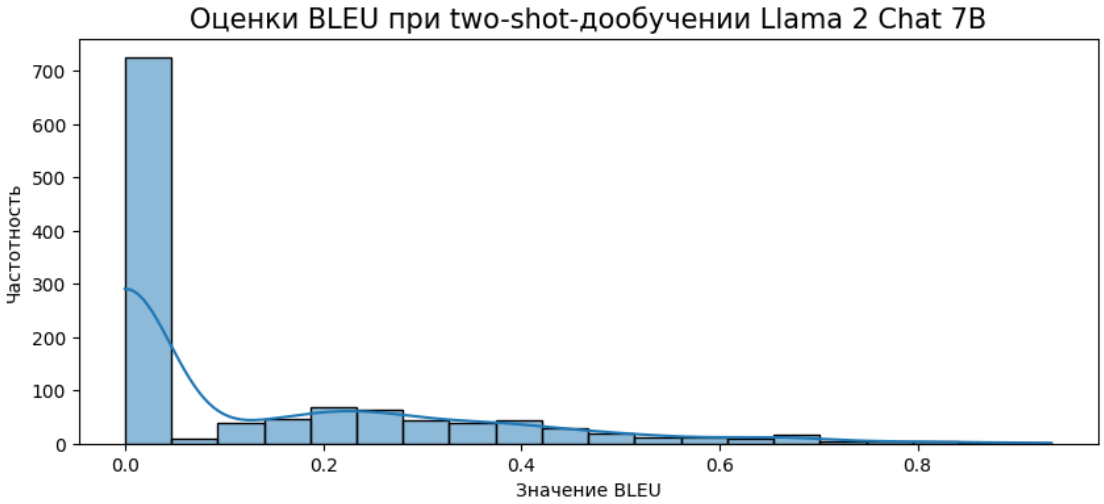

Рисунок 11 – Распределение оценок BLEU при two-shot дообучении Llama 2 Chat 7B

Поскольку на уровне значимости 0,05 мы отклонили гипотезы о соответствии распределения оценок BLEU для моделей Mistral и Llama нормальному распределению, то для сравнения непосредственных различий между распределениями моделей, дообученных в two-shot режиме, использовался критерий Манна-Уитни. Результаты представлены в таблице 11.

Таблица 11 – U-критерий Манна-Уитни для оценок моделей, дообученных в two-shot режиме

Сравниваемые модели	Статистика	p-value
Mistral Instruct 7B (two-shot) и Llama 2 Chat 7B (two-shot)	7 663,00	0,000
Mistral Instruct 7B (two-shot) и Mistral Instruct 7B (zero-shot)	8 562,00	0,000
Mistral Instruct 7B (two-shot) и Mistral Instruct 7B (one-shot)	5 796,50	0,520
Llama 2 Chat 7B (two-shot) и Llama 2 Chat 7B (zero-shot)	8 053,00	0,000
Llama 2 Chat 7B (two-shot) и Llama 2 Chat 7B (one-shot)	6 827,00	0,003

Что касается U-тестов Манна-Уитни, то на уровне значимости 0,05 они показали, что имеются статистически значимые различия между:

1) Mistral, дообученной в режиме zero-shot, и Mistral, дообученной в режиме two-shot;

2) Llama 2, дообученной в режиме zero-shot, и Llama 2, дообученной в режиме two-shot;

3) Llama 2, дообученной в режиме one-shot, и Llama 2, дообученной в режиме two-shot;

4) Mistral, дообученной в режиме two-shot, и Llama 2, дообученной в режиме two-shot.

Однако на уровне значимости 0,05 мы не можем отклонить гипотезу о равенстве распределений Mistral, дообученной в режиме one-shot, и Mistral, дообученной в режиме two-shot. Данный факт, возможно, говорит нам о том, что для дообучения Mistral в режиме few-shot модели не важно, сколько примеров эталонных ответов ей передаётся в промпт. Тем не менее, в

следующем подпункте, мы попытается выяснить, насколько эффективно при дообучении моделей как Mistral, так и Llama добавление третьего эталонного примера.

3.2.3. Three-shot обучение

Обучение модели в режиме three-shot, являясь подтипом few-shot обучения, предполагает передачу модели в промпт трёх эталонных примеров переводов, а также непосредственно инструкцию на перевод в зависимости от языковой пары.

Так, один из промптов с задачей перевода с русского языка на испанский выглядел следующим образом: «*Translate the following text from Russian to Spanish\n\n### Russian:\nНастоящий документ носит информационный и справочный характер в отношении указанного в нем программного обеспечения семейства Dr.Web.\n\n### Spanish:\nEl presente documento tiene carácter informativo y de consulta respecto al software de la familia Dr.Web que en él aparece.\n\n### Russian:\nДругие пользователи данной установки смогут по-прежнему запускать выполнение файла powershell.exe.\n\n### Spanish:\nOtros usuarios de la instalación podrán ejecutar powershell.exe.\n\n### Russian:\nФайлы хранилища электронной почты кэшируются для отдельных пользователей.\n\n### Spanish:\nLos archivos de almacenamiento de correo electrónico se almacenan en caché de forma individual para cada usuario.\n\n### Russian:\nСоберите информацию о сети и настройте виртуальную сеть.\n\n### Spanish:\n*».

Оценки переводов моделей Mistral и Llama 2, дообученных в three-shot режиме, представлены в таблицах 12 и 13 соответственно. Для оценок моделей использовался 95% доверительный интервал.

Таблица 12 – Оценки BLEU при three-shot дообучении Mistral Instruct 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,61 ± 0,03	0,43 ± 0,03	0,69 ± 0,03	0,45 ± 0,03	0,56 ± 0,03	0,54 ± 0,03	0,55 ± 0,01
BLEU-2	0,40 ± 0,04	0,23 ± 0,03	0,45 ± 0,04	0,24 ± 0,03	0,34 ± 0,03	0,31 ± 0,03	0,34 ± 0,01
BLEU-3	0,28 ± 0,04	0,15 ± 0,02	0,33 ± 0,04	0,15 ± 0,03	0,24 ± 0,03	0,21 ± 0,03	0,22 ± 0,01
BLEU-4	0,20 ± 0,03	0,11 ± 0,02	0,25 ± 0,04	0,10 ± 0,02	0,15 ± 0,03	0,15 ± 0,02	0,15 ± 0,01
BLEU-S	0,26 ± 0,04	0,14 ± 0,02	0,35 ± 0,04	0,14 ± 0,03	0,23 ± 0,03	0,21 ± 0,03	0,25 ± 0,01

Таблица 13 – Оценки BLEU при three-shot дообучении Llama 2 Chat 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,52 ± 0,03	0,31 ± 0,03	0,55 ± 0,04	0,27 ± 0,03	0,47 ± 0,03	0,39 ± 0,03	0,42 ± 0,01
BLEU-2	0,30 ± 0,03	0,14 ± 0,03	0,34 ± 0,04	0,12 ± 0,02	0,26 ± 0,03	0,19 ± 0,03	0,23 ± 0,01
BLEU-3	0,20 ± 0,03	0,09 ± 0,02	0,24 ± 0,03	0,07 ± 0,02	0,17 ± 0,03	0,11 ± 0,02	0,15 ± 0,01
BLEU-4	0,14 ± 0,03	0,06 ± 0,02	0,18 ± 0,03	0,05 ± 0,02	0,11 ± 0,02	0,06 ± 0,01	0,10 ± 0,01
BLEU-S	0,19 ± 0,03	0,08 ± 0,02	0,25 ± 0,04	0,07 ± 0,02	0,17 ± 0,03	0,10 ± 0,02	0,14 ± 0,01

Сравнивая результаты в таблицах 8-9 и 12-13, можно сделать следующие выводы:

1. По прежнему, в режимах few-shot дообучения модель Mistral показывает себя лучше, чем Llama 2.

2. У модели Mistral на оценке BLEU-S нам удалось добиться значения 0,25, говорящего о хорошем качестве модели в целом;

3. Лучше всего модели справляются с переводами с испанского на английский язык (и чуть хуже с английского на испанский).

4. При three-shot дообучении удалось добиться неплохих результатов для решения подзадачи перевода с русского на английский языки.

5. Если сравнивать two-shot и three-shot обучение, то в последнем режиме прирост по метрикам BLEU составил 0,1-0,2 пункта.

Проверим, является ли замеченная нами разница распределений оценок BLEU статистически значимой. Для этого необходимо сделать тесты на проверку на подчинение результатов моделей нормальному распределению при помощи критерия согласия Пирсона. Результаты данных тестов представлены в таблице 14.

Таблица 14 – Критерий согласия Пирсона для оценок моделей, дообученных в three-shot режиме

Модель	Статистика	p-value
Mistral Instruct 7B	163,05	0,0
Llama 2 Chat 7B	266,33	0,0

Поскольку критерий согласия Пирсона показал, что можно отклонить гипотезу о соответствии распределений оценок BLEU, полученных в результате three-shot дообучения, нормальному распределению, то для проверки гипотез о статистически значимых различиях между распределениями оценок данных моделей, использовались U-тесты Манна-Уитни, с результатами которых можно ознакомиться в таблице 15.

Таблица 15 – U-критерий Манна-Уитни для оценок моделей, дообученных в three-shot режиме

Сравниваемые модели	Статистика	p-value
1	2	3
Mistral Instruct 7B (three-shot) и Llama 2 Chat 7B (three-shot)	7 594,00	0,000
Mistral Instruct 7B (three-shot) и Mistral Instruct 7B (zero-shot)	8 885,50	0,000
Mistral Instruct 7B (three-shot) и Mistral Instruct 7B (one-shot)	6 199,50	0,119
Mistral Instruct 7B (three-shot) и Mistral Instruct 7B (two-shot)	5 926,00	0,348
Llama 2 Chat 7B (three-shot) и Llama 2 Chat 7B (zero-shot)	8 511,50	0,000

1	2	3
Llama 2 Chat 7B (three-shot) и Llama 2 Chat 7B (one-shot)	7 364,50	0,000
Llama 2 Chat 7B (three-shot) и Llama 2 Chat 7B (two-shot)	6 013,50	0,255

Таким образом, статистические U-тесты Манна-Уитни показали, что мы не можем отклонить гипотезу о равенстве распределений между моделями:

– Mistral, дообученной в режиме three-shot, и Mistral, дообученной в режиме one-shot;

– Mistral, дообученной в режиме three-shot, и Mistral, дообученной в режиме two-shot;

– Llama 2, дообученной в режиме three-shot, и Llama 2, дообученной в режиме two-shot.

Резюмируя основные положения пункта о few-shot обучении в целом, можно сделать вывод, что неплохих результатов при решении специализированных задач с помощью модели Mistral можно добиться с помощью передачи модели в промпт как минимум одного примера эталонного варианта ответа. В нашем случае, передача в промпт трёх примеров помогла добиться модели метрики BLEU-S на уровне $0{,}25 \pm 0{,}01$, что говорит о хорошем качестве генерируемых моделью переводов для всех языковых пар. Тем не менее, U-тест Манна-Уитни показал, что разница между оценками переводов моделей Mitral, дообученных в режимах one-, two- и three-shot, на уровне значимости 0,05 может быть статистически незначимой.

Что касается моделей Llama, то для них передача примеров в промпт также имеет большое значение. Более качественных результатов генерации можно добиться, передав модели от двух примеров. U-тест Манна-Уитни также показал, что разница между оценками переводов модели Llama, дообученной в режиме two-shot, и аналогичной моделью, но обученной с помощью three-shot дообучения, на уровне значимости 0,05 может быть статистически незначимой.

3.3. PEFT обучение

В рамках PEFT подхода дообучения больших языковых моделей был выбран метод низкоранговой адаптации квантизованных моделей, предполагающий не полное обновление весов моделей, а лишь некоторой части. С количеством исходных и обучаемых параметров модели можно ознакомиться в таблице 16.

Таблица 16 – Параметры языковых моделей

Модель	Всего параметров (шт.)	Обучаемые параметры (шт.)	Обучаемые параметры (%)
Mistral Instruct 7B	3 837 112 320	85 041 152	2,22
Llama 2 Chat 7B	3 581 521 920	81 108 992	2,27

При дообучении моделей PEFT способом, в обучающую выборку в качестве промпта передавалась инструкцию с заданием на перевод, а также непосредственно сам перевод.

Пример промпта для перевода с английского на русский язык из тренировочной выборки: «*Translate the following text from English to Russian\n\n### English:\nYou can streamline signing in to websites and applications, as well as filling out online forms, by keeping all your passwords in a single trusted application.\n\n### Russian:\nУпростите себе вход на сайты и в приложения, а также заполнение онлайн-форм, сохранив все ваши пароли в одном надежном приложении.».*

В промптах тестовой выборке содержались лишь инструкции и непосредственно тексты, которые необходимо перевести.

Пример промпта для перевода с английского на русский язык из тестовой выборки: «*Translate the following text from English to Russian\n\n### English:\nYour personal license is not required for operation in the centralized protection mode.\n\n### Russian:\n».*

Ниже представлены графики дообучения больших языковых моделей Mistral-7B-Instruct-v0.2 (рисунок 12) и Llama-2-7b-chat-hf (рисунок 13).

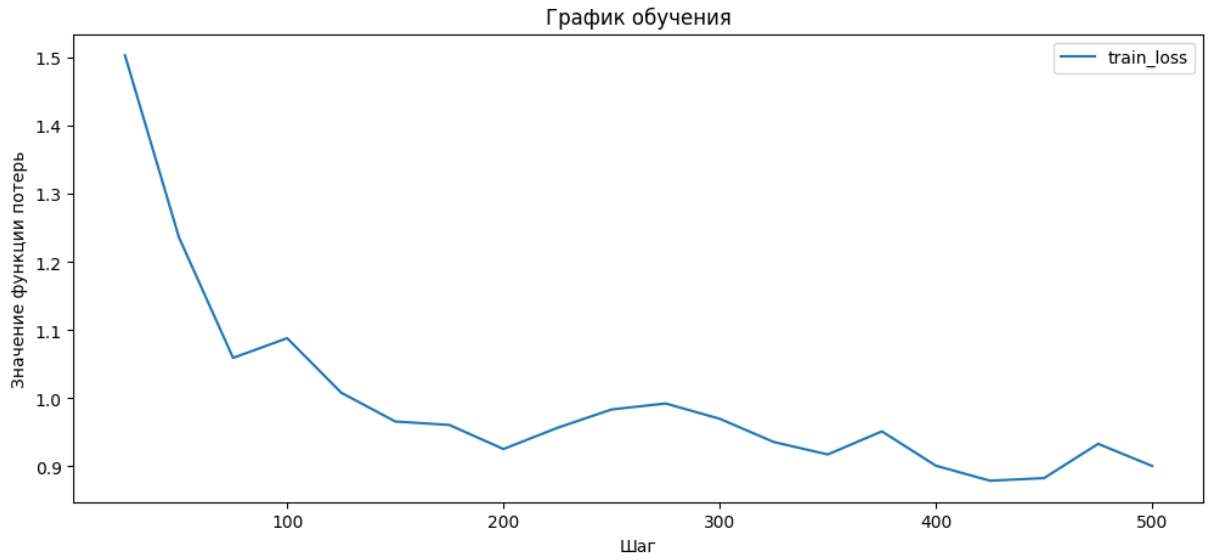

Рисунок 12 – График обучения Mistral

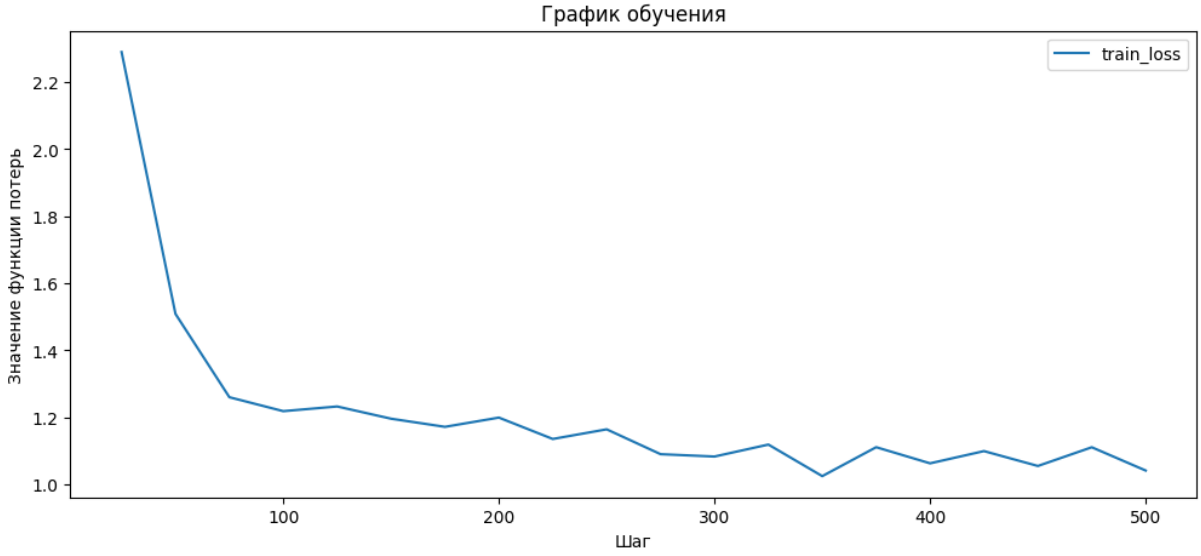

Рисунок 13 – График обучения Llama 2

Если сравнивать графики обучения моделей, то можно заметить, что наименьшее значение функции потерь у Mistral находится 425 контрольной точке, а у Llama 2 – в 350. Также, в целом, при обучении Mistral удалось добиться более низкого значения функции кросс-энтропии, нежели чем Llama 2.

В таблице 17 представлены оценки BLEU на решении задачи мультиязычного перевода для модели Mistral из 425 контрольной точки.

Таблица 17 – Оценки BLEU при PEFT дообучении Mistral Instruct 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,51 ± 0,04	0,45 ± 0,04	0,49 ± 0,04	0,45 ± 0,04	0,46 ± 0,04	0,48 ± 0,04	0,47 ± 0,02
BLEU-2	0,35 ± 0,04	0,26 ± 0,04	0,31 ± 0,04	0,26 ± 0,03	0,27 ± 0,04	0,29 ± 0,03	0,29 ± 0,01
BLEU-3	0,26 ± 0,04	0,17 ± 0,03	0,22 ± 0,03	0,18 ± 0,03	0,19 ± 0,03	0,19 ± 0,03	0,20 ± 0,01
BLEU-4	0,20 ± 0,03	0,11 ± 0,02	0,16 ± 0,03	0,10 ± 0,02	0,12 ± 0,03	0,13 ± 0,02	0,14 ± 0,01
BLEU-S	0,26 ± 0,04	0,15 ± 0,03	0,22 ± 0,04	0,15 ± 0,03	0,18 ± 0,03	0,19 ± 0,03	0,19 ± 0,01

В таблице 18 представлены оценки BLEU на решении аналогичной задачи для модели Llama 2 из 350 контрольной точки.

Таблица 18 – Оценки BLEU при PEFT дообучении Llama 2 Chat 7B

Оценка	Языковая пара						
	EN-ES	EN-RU	ES-EN	ES-RU	RU-EN	RU-ES	Все языки
BLEU-1	0,34 ± 0,03	0,18 ± 0,02	0,41 ± 0,04	0,22 ± 0,02	0,33 ± 0,04	0,39 ± 0,03	0,31 ± 0,01
BLEU-2	0,19 ± 0,03	0,08 ± 0,02	0,23 ± 0,03	0,09 ± 0,02	0,17 ± 0,03	0,21 ± 0,03	0,16 ± 0,01
BLEU-3	0,13 ± 0,02	0,05 ± 0,02	0,15 ± 0,03	0,05 ± 0,01	0,10 ± 0,02	0,14 ± 0,03	0,10 ± 0,01
BLEU-4	0,08 ± 0,02	0,03 ± 0,01	0,09 ± 0,02	0,03 ± 0,01	0,06 ± 0,02	0,09 ± 0,02	0,07 ± 0,01
BLEU-S	0,13 ± 0,02	0,05 ± 0,02	0,14 ± 0,03	0,05 ± 0,01	0,10 ± 0,02	0,13 ± 0,03	0,10 ± 0,01

Аналогично режимам обучения, рассмотренным ранее, модель Mistral Instruct 7B получила более высокие оценки BLEU нежели Llama 2 Chat 7B. Также стоит отметить, что обе модели хуже всего справлялись с задачами перевода на русский язык.

Чтобы сравнить статистическую значимость различий оценок моделей, обученных с помощью PEFT, с обученными ранее моделями (распределение оценок которых отлично от нормального), проведены U-тесты Манна-Уитни. С их результатами можно ознакомиться в таблице 19.

Таблица 19 – U-критерий Манна-Уитни для оценок моделей, дообученных в PEFT режиме

Сравниваемые модели	Статистика	p-value
Mistral Instruct 7B (PEFT) и Llama 2 Chat 7B (PEFT)	8 615,00	0,000
Mistral Instruct 7B (PEFT) и Mistral Instruct 7B (zero-shot)	8 534,00	0,000
Mistral Instruct 7B (PEFT) и Mistral Instruct 7B (one-shot)	5 482,00	0,946
Mistral Instruct 7B (PEFT) и Mistral Instruct 7B (two-shot)	5 179,50	0,445
Mistral Instruct 7B (PEFT) и Mistral Instruct 7B (three-shot)	4 757,00	0,086
Llama 2 Chat 7B (PEFT) и Llama 2 Chat 7B (zero-shot)	7 241,50	0,000
Llama 2 Chat 7B (PEFT) и Llama 2 Chat 7B (one-shot)	5 941,00	0,331
Llama 2 Chat 7B (PEFT) и Llama 2 Chat 7B (two-shot)	4 526,00	0,025
Llama 2 Chat 7B (PEFT) и Llama 2 Chat 7B (three-shot)	4 002,50	0,001

Таким образом, статистические U-тесты Манна-Уитни на уровне значимости 0,05 показали, что мы не можем отклонить гипотезу о равенстве распределений между моделями:

– Mistral, дообученной в PEFT режиме, и Mistral, дообученной в one-shot режиме;

– Mistral, дообученной в PEFT режиме, и Mistral, дообученной в two-shot режиме;

– Mistral, дообученной в PEFT режиме, и Mistral, дообученной в three-shot режиме;

– Llama 2, дообученной в PEFT режиме, и Llama 2, дообученной в one-shot режиме.

Данные результаты позволяют нам сделать следующие выводы:

1. Результаты Mistral, дообученной с помощью PEFT, сравнимы с результатами Mistral во few-shot режимах.

2. Среди few-shot режимов, наиболее близким распределением оценок к модели Mistral, дообученной с помощью PEFT, обладает аналогичная модель, дообученная в one-shot режиме.

3. Что касается модели Llama 2, то результаты данной модели, дообученной с помощью PEFT, сравнимы только с результатами Llama 2 в one-shot режиме.

4. Следовательно, для получения похожих результатов при работе с обеими моделями, можно не проводить долгий PEFT, а ограничиться лишь передачей в промпт одного примера эталонного решения поставленной задачи.

3.4. Сравнительный анализ методов дообучения

В данной главе в качестве методов дообучения больших языковых моделей для использования в специализированных задачах были применены техники zero-shot, few-shot и PEFT.

Zero-shot дообучение предполагает передачу модели только инструкции с заданием на перевод без примеров. За счёт этого, данный метод можно применять без дополнительных временных затрат. Однако, результаты работы моделей, дообученных в режиме zero-shot, демонстрируют более низкие оценки BLEU, по сравнению с few-shot или PEFT моделями.

Метод few-shot представляет собой передачу модели, помимо инструкции на перевод, ещё и нескольких примеров эталонного выполнения задания. Данный метод несколько более затратный по временным ресурсам, чем zero-shot, за счёт необходимости в подборе эталонных вариантов перевода.

Последний рассмотренный нами метод PEFT позволяет проводить обновление весов модели (в нашем случае чуть больше 2 % исходной модели)

с помощь добавления низкоранговых матриц. Однако для проведения такого вида дообучения необходимо провести больше этапов подготовки, нежели чем при zero-shot и few-shot методах. Кроме того, данный вид дообучения требует больше вычислительных ресурсов.

Резюмировать основные отличия упомянутых методов можно в таблице 20.

Таблица 20 – Сравнение методов дообучения

Аспект	Zero-shot	Few-shot	PEFT
Подготовка обучающей выборки	Не требуется. Модели нужна только инструкция на перевод	Минимальна. Необходима инструкция плюс несколько примеров пар «текст на исходном языке-эталонный перевод»	Требуется полноценная обучающая выборка с инструкциями и множеством примеров пар «текст на исходном языке-эталонный перевод»
Временные затраты	На подбор промпта, загрузку и оценку модели	На подбор промпта, подготовку эталонных примеров, загрузку и оценку модели	На подбор промпта, подготовку полноценной выборки, загрузку модели, обновление её весов и оценку модели
Вычислительные затраты	Загрузка модели (до 8 ГБ)	Загрузка модели (до 8 ГБ)	Загрузка модели (до 8 ГБ) + обучение весов (в нашем случае минимально 1,5 ГБ/батч)
Качество перевода	Низкие значения BLEU	Чем больше примеров передано модели в промпт, тем более высокие значения BLEU	Высокие значения BLEU

3.5. Выводы по Главе 3

В третьей главе мы провели сравнение методов дообучения больших языковых моделей Mistral и Llama 2 для решения специализированной задачи мультиязычного перевода в сфере информационной безопасности.

Обучение моделей производилось в режимах zero-shot, few-shot (one-shot, two-shot, three-shot), а также при помощи техники PEFT, а именно низкоранговой адаптации параметров квантизованных языковых моделей.

Сравнение качества перевода дообученных моделей проводилось при помощи метрик BLEU (BLEU-1, BLEU-2, BLEU-3, BLEU-4 и BLEU-S) для шести языковых пар (EN-ES, EN-RU, ES-EN, ES-RU, RU-EN, RU-ES). Результаты по основной метрике BLEU-S в 95% доверительном интервале представлены в таблице 21.

Таблица 21 – Сравнение языковых моделей по BLUE-S

Языковая пара	Mistral Instruct 7B					Llama 2 Chat 7B				
	Zero-shot	Few-shot			PEFT	Zero-shot	Few-shot			PEFT
		One-shot	Two-shot	Three-shot			One-shot	Two-shot	Three-shot	
EN-ES	0,15 ± 0,03	0,24 ± 0,03	0,25 ± 0,03	0,26 ± 0,04	0,26 ± 0,04	0,11 ± 0,02	0,13 ± 0,02	0,20 ± 0,03	0,19 ± 0,03	0,13 ± 0,02
EN-RU	0,05 ± 0,02	0,11 ± 0,03	0,12 ± 0,03	0,14 ± 0,02	0,15 ± 0,03	0,02 ± 0,01	0,05 ± 0,02	0,07 ± 0,02	0,08 ± 0,02	0,05 ± 0,02
ES-EN	0,18 ± 0,03	0,30 ± 0,04	0,32 ± 0,04	0,35 ± 0,04	0,22 ± 0,04	0,13 ± 0,02	0,17 ± 0,03	0,22 ± 0,03	0,25 ± 0,04	0,14 ± 0,03
ES-RU	0,04 ± 0,01	0,12 ± 0,03	0,13 ± 0,03	0,14 ± 0,03	0,15 ± 0,03	0,03 ± 0,01	0,04 ± 0,01	0,06 ± 0,02	0,07 ± 0,02	0,05 ± 0,01
RU-EN	0,13 ± 0,02	0,20 ± 0,03	0,21 ± 0,03	0,23 ± 0,03	0,18 ± 0,03	0,06 ± 0,01	0,11 ± 0,02	0,14 ± 0,03	0,17 ± 0,03	0,10 ± 0,02
RU-ES	0,08 ± 0,02	0,19 ± 0,03	0,20 ± 0,03	0,21 ± 0,03	0,19 ± 0,03	0,07 ± 0,02	0,07 ± 0,02	0,08 ± 0,02	0,10 ± 0,02	0,13 ± 0,03
Все языки	0,10 ± 0,01	0,19 ± 0,01	0,21 ± 0,01	0,25 ± 0,01	0,19 ± 0,01	0,07 ± 0,01	0,09 ± 0,01	0,13 ± 0,01	0,14 ± 0,01	0,10 ± 0,01

Помимо качества перевода, модели также сравнивались по аспектам подготовки обучающей выборки, временным, а также вычислительным затратам (см. таблицу 20).

На основании проведённого анализа можно сделать следующие выводы:

1. Несмотря на одинаковый размер, большую способность к дообучению для решения специализированных задач показала Mistral Instruct 7B, нежели Llama 2 Chat 7B.

2. В целом, наилучших результатов в решении выбранной специализированной задачи добилась модель Mistral при three-shot дообучении.

3. Модели, дообученные во few-shot режимах показали результаты, сравнимые с результатами моделей, обученных с помощью PEFT. При этом, при few-shot режимах затрачивалось меньше временных ресурсов, а также ресурсов памяти.

4. Модели, дообученные при помощи таких подвидов few-shot обучения, как two-shot и three-shot оказались результативнее, нежели PEFT модели.

5. Если сравнивать между собой подвиды few-shot режимов, то наилучшие метрики BLEU-S продемонстрировали three-shot модели, что позволяет нам сделать вывод о том, что чем больше примеров эталонного решения задач передаётся на вход модели, тем лучше получится результат.

6. Самым быстрым способом решения специализированных задач является zero-shot дообучение. Однако, в решении нашей задачи модели, дообученные в таком решении показали самые низкие результаты.

7. Наиболее высокое качество перевода у моделей Mistral и Llama наблюдается при решении подзадачи перевода с испанского на английский язык, а также с английского на испанский. Данный факт говорит нам о том, что указанные модели лучше работают с латинскими токенами, поскольку предобучались на несбалансированной выборке, преимущественно состоящей из текстов на английском языке.

8. Хуже всего модели справились с подзадачей перевода с испанского языка на русский. При решении подобного вида задач лучше отдать предпочтение PEFT-дообучению.

ЗАКЛЮЧЕНИЕ

В первой главе была решена первая задача нашего исследования, а именно проведён обзор источников литературы по языковому моделированию. Были рассмотрены основные понятия языкового моделирования: языковая модель, n-граммы, кросс-энтропия, эмбеддинги, трансформеры. Также были перечислены основные характеристики и проблемы больших языковых моделей. Среди последних выделяется слабая способность таких моделей решать специализированные задачи.

Во второй главе мы определились с большими языковыми моделями для дообучения. Выбор пал на модели Mistral Instruct 7B и Llama 2 Chat 7B поскольку они:

– выложены в открытый доступ;

– имеют одинаковое количество параметров;

– демонстрируют SOTA-результаты в решении общих задач из области обработки естественного языка;

– могут быть запущены локально на одном устройстве с одной видеокартой.

Так же во второй главе мы определись со специализированной задачей для дообучения больших языковых моделей: мультиязычным переводом в сфере информационной безопасности. Для обучения моделей мы собрали датасет, состоящих из переводов документаций компаний, работающих в данной сфере. В датасете содержится 1001 предложение. Для каждого предложения имеется перевод на три языка: русский, английский и испанский.

В этой же главе мы описали алгоритм дообучения больших языковых моделей для выбранных методов дообучения: zero-shot, few-shot и PEFT. Предложенный нами алгоритм дообучения больших языковых состоит из следующих шагов: загрузка датасета для дообучения, его предобработка, подготовка модели для обучения, токенизация текстов, генерация ответов,

постобработка сгенерированных текстов и оценка работы модели в целом. Однако, для каждого режима дообучения имеются свои особенности, и для дообучения при помощи PEFT дополнительно добавляются этапы настройки низкоранговой адаптации, ускорителя обучения и непосредственного обучения на тренировочной выборке.

В третьей главе мы провели сравнение методов дообучения и дообученных языковых моделей в решении специализированной задачи и пришли к следующим выводам:

1. Несмотря на одинаковый размер, большую способность к дообучению для решения специализированных задач показала Mistral Instruct 7B, нежели Llama 2 Chat 7B.

2. В целом, наилучших результатов в решении выбранной специализированной задачи добилась модель Mistral при three-shot дообучении.

3. Модели, дообученные во few-shot режимах показали результаты, сравнимые с результатами моделей, обученных с помощью PEFT. При этом, при few-shot режимах затрачивалось меньше временных ресурсов, а также ресурсов памяти.

4. Модели, дообученные при помощи таких подвидов few-shot обучения, как two-shot и three-shot оказались результативнее, нежели PEFT модели.

5. Если сравнивать между собой подвиды few-shot режимов, то наилучшие метрики BLEU-S продемонстрировали three-shot модели, что позволяет нам сделать вывод о том, что чем больше примеров эталонного решения задач передаётся на вход модели, тем лучше получится результат.

6. Самым быстрым способом решения специализированных задач является zero-shot дообучение. Однако, в решении нашей задачи модели, дообученные в таком решении показали самые низкие результаты.

7. Наиболее высокое качество перевода у моделей Mistral и Llama наблюдается при решении подзадачи перевода с испанского на английский

язык, а также с английского на испанский. Данный факт говорит нам о том, что указанные модели лучше работают с латинскими токенами, поскольку предобучались на несбалансированной выборке, преимущественно состоящей из текстов на английском языке.

8. Хуже всего модели справились с подзадачей перевода с испанского языка на русский. При решении подобного вида задач лучше отдать предпочтение PEFT-дообучению.

Таким образом, цель выпускной квалификационной работы достигнута: было проведено сравнение методов zero-shot, few-shot и PEFT применительно к дообучению больших языковых моделей Mistral и Llama 2 для решения задачи мультиязычного машинного перевода в сфере информационной безопасности с и на три языка: русский, английский и испанский.

В качестве перспективы дальнейшего исследования мы бы хотели наметить использование дополнительных техник, позволяющих повысить качество дообучения больших языковых моделей для решения специализированных задач. В частности, в последние месяцы неплохие результаты демонстрируют модели, при обучении которых задействовалась генерация с дополненной выборкой (RAG) [47].

Кроме того, можно расшить специализированную задачу перевода с трёх языков до большего количества. Особый интерес для исследователей из области обработки естественного языка представляет также добавление малоресурсных языков [24].

Помимо прочего, можно расширить спектр больших языковых моделей для исследования. Так, например, несколько недель назад стала доступна Llama 3 [48].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Scaling Laws for Neural Language Models / J. Kaplan, S. McCandlish, T. Henighan, [et al.] arXiv:2001.08361 [cs, stat]. – 2020. – URL: http://arxiv.org/abs/2001.08361 (date accessed: 28.02.2024). – Text : electronic.

2. Shanahan, M. Talking about Large Language Models / M. Shanahan // Communications of the ACM. – 2024. – Vol. 67. – № 2. – P. 68-79.

3. Fine-tuning Large Language Models for Domain-specific Machine Translation / J. Zheng, H. Hong, X. Wang, [et al.] arXiv:2402.15061 [cs]. – 2024. – URL: http://arxiv.org/abs/2402.15061 (date accessed: 05.03.2024). – Text : electronic.

4. A Survey of Large Language Models / W. X. Zhao, K. Zhou, J. Li, [et al.] arXiv:2303.18223 [cs]. – 2023. – URL: http://arxiv.org/abs/2303.18223 (date accessed: 09.01.2024). – Text : electronic.

5. Jurafsky, D. Speech and Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (3rd ed. draft) / D. Jurafsky, J. H. Martin. – 2023. – URL: https://web.stanford.edu/~jurafsky/slp3/ (date accessed: 09.01.2024). – Text : electronic.

6. Hayes, B. First Links in the Markov Chain / B. Hayes // American Scientist. – 2013. – Vol. 101. – P. 92-97.

7. Li, H. Language models: past, present, and future / H. Li // Communications of the ACM. – 2022. – Vol. 65. – № 7. – P. 56-63.

8. A Neural Probabilistic Language Model / Y. Bengio, R. Ducharme, P. Vincent, C. Jauvin // Journal of Machine Learning Research 3. – 2003. – P. 1137-1155.

9. Efficient Estimation of Word Representations in Vector Space / T. Mikolov, K. Chen, G. Corrado, J. Dean. – Text : electronic // 1st International

Conference on Learning Representations. – Scottsdale, Arizona, USA, 2013. – URL: http://arxiv.org/abs/1301.3781 (date accessed: 13.03.2024).

10. Kalyan, K. S. A survey of GPT-3 family large language models including ChatGPT and GPT-4 / K. S. Kalyan // Natural Language Processing Journal. – 2024. – Vol. 6. – P. 1-48.

11. Attention is All you Need / A. Vaswani, N. Shazeer, N. Parmar [et al.]. – Text : electronic // Advances in Neural Information Processing Systems : 5998-6008. – Curran Associates, Inc., 2017. – Vol. 30. – URL: https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html (date accessed: 13.03.2024).

12. A survey of transformers / T. Lin, Y. Wang, X. Liu, X. Qiu // AI Open. – 2022. – Vol. 3. – P. 111-132.

13. Emergent Abilities of Large Language Models / J. Wei, Y. Tay, R. Bommasani [et al.] // Transactions on Machine Learning Research. – 2022.

14. Mistral 7B / A. Q. Jiang, A. Sablayrolles, A. Mensch, [et al.] arXiv:2310.06825 [cs]. – arXiv, 2023. – URL: http://arxiv.org/abs/2310.06825 (date accessed: 13.03.2024). – Text : electronic.

15. Language Models are Few-Shot Learners / T. Brown, B. Mann, N. Ryder [et al.]. – Text : electronic // Advances in Neural Information Processing Systems 33. – Vancouver, Canada : Curran Associates, Inc., 2020. – P. 1877-1901. – URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html (date accessed: 30.01.2024).

16. Investigating the Catastrophic Forgetting in Multimodal Large Language Model Fine-Tuning / Y. Zhai, S. Tong, X. Li [et al.]. – Text : electronic // Conference on Parsimony and Learning Conference on Parsimony and Learning. – PMLR, 2024. – P. 202-227. – URL: https://proceedings.mlr.press/v234/zhai24a.html (date accessed: 29.02.2024).

17. Gender bias in transformers: A comprehensive review of detection and mitigation strategies / P. Nemani, Y. D. Joel, P. Vijay, F. F. Liza // Natural Language Processing Journal. – 2024. – Vol. 6. – P. 100047.

18. LoRA: Low-Rank Adaptation of Large Language Models / E. J. Hu, Y. Shen, P. Wallis [et al.]. – Text : electronic // International Conference on Learning Representations. – 2022. – URL: https://openreview.net/forum?id=nZeVKeeFYf9 (date accessed: 21.02.2024).

19. Code Llama: Open Foundation Models for Code / B. Rozière, J. Gehring, F. Gloeckle, [et al.] arXiv:2308.12950 [cs]. – 2024. – URL: http://arxiv.org/abs/2308.12950 (date accessed: 29.02.2024). – Text : electronic.

20. Enhancing Automated Scoring of Math Self-Explanation Quality Using LLM-Generated Datasets: A Semi-Supervised Approach / R. Nakamoto, B. Flanagan, T. Yamauchi [et al.] // Computers. – 2023. – Vol. 12. – № 11. – P. 217.

21. Using Deep Neural Networks to Translate Multi-lingual Threat Intelligence / P. Ranade, S. Mittal, A. Joshi, K. Joshi. – Text : electronic // 2018 IEEE International Conference on Intelligence and Security Informatics (ISI). – 2018. – P. 238-243. – URL: https://ieeexplore.ieee.org/abstract/document/8587374 (date accessed: 01.05.2024).

22. Garg, A. Machine Translation: A Literature Review / A. Garg, M. Agarwal arXiv:1901.01122 [cs]. – 2018. – URL: http://arxiv.org/abs/1901.01122 (date accessed: 18.03.2024). – Text : electronic.

23. Koshkin, R. TransLLaMa: LLM-based Simultaneous Translation System / R. Koshkin, K. Sudoh, S. Nakamura arXiv:2402.04636 [cs]. – 2024. – URL: http://arxiv.org/abs/2402.04636 (date accessed: 28.02.2024). – Text : electronic.

24. No Language Left Behind: Scaling Human-Centered Machine Translation / NLLB Team, M. R. Costa-jussà, J. Cross, [et al.] arXiv:2207.04672 [cs]. – 2022. – URL: http://arxiv.org/abs/2207.04672 (date accessed: 18.03.2024). – Text : electronic.

25. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer / C. Raffel, N. Shazeer, A. Roberts [et al.] // Journal of Machine Learning Research. – 2020. – Vol. 21. – № 1. – P. 1-67.

26. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding / J. Devlin, M.-W. Chang, K. Lee, K. Toutanova. – Text : electronic // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. – Minneapolis, Minnesota, USA : Association for Computational Linguistics, 2019. – Vol. 1 (Long and Short Papers). – P. 4171-4186. – URL: https://aclanthology.org/N19-1423 (date accessed: 25.01.2024).

27. Lee, J.-S. Patent claim generation by fine-tuning OpenAI GPT-2 / J.-S. Lee, J. Hsiang // World Patent Information. – 2020. – Vol. 62. – P. 101983.

28. Parameter-Efficient Transfer Learning for NLP / N. Houlsby, A. Giurgiu, S. Jastrzebski [et al.]. – Text : electronic // Proceedings of the 36th International Conference on Machine Learning. – Long Beach, California, USA : PMLR, 2019. – Vol. 97. – P. 2790-2799. – URL: https://proceedings.mlr.press/v97/houlsby19a.html (date accessed: 21.02.2024).

29. Li, X. L. Prefix-Tuning: Optimizing Continuous Prompts for Generation / X. L. Li, P. Liang. – Text : electronic // Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. – Association for Computational Linguistics, 2021. – Vol. 1: Long Papers. – P. 4582-4597. – URL: https://aclanthology.org/2021.acl-long.353 (date accessed: 21.02.2024).

30. P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks / X. Liu, K. Ji, Y. Fu [et al.]. – Text : electronic // Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. – Dublin, Ireland : Association for Computational Linguistics, 2022. – Vol. 2: Short Papers. – P. 61-68. – URL: https://aclanthology.org/2022.acl-short.8 (date accessed: 25.01.2024).

31. QLORA: Efficient Finetuning of Quantized LLMs / T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer. – Text : electronic // 37th Conference on Neural Information Processing Systems (NeurIPS 2023). – 2023. – URL: https://proceedings.neurips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html (date accessed: 21.02.2024).

32. Ray, P. P. Benchmarking, ethical alignment, and evaluation framework for conversational AI: Advancing responsible development of ChatGPT / P. P. Ray // BenchCouncil Transactions on Benchmarks, Standards and Evaluations. – 2023. – Vol. 3. – № 3. – P. 100136.

33. Bleu: a Method for Automatic Evaluation of Machine Translation / K. Papineni, S. Roukos, T. Ward, W.-J. Zhu. – Text : electronic // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. – Philadelphia, Pennsylvania, USA : Association for Computational Linguistics, 2002. – P. 311-318. – URL: https://aclanthology.org/P02-1040 (date accessed: 30.01.2024).

34. Lin, C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries / C.-Y. Lin. – Text : electronic // Text Summarization Branches Out. – Barcelona, Spain : Association for Computational Linguistics, 2004. – P. 74-81. – URL: https://aclanthology.org/W04-1013 (date accessed: 30.01.2024).

35. Banerjee, S. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments / S. Banerjee, A. Lavie. – Text : electronic // Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. – Ann Arbor, Michigan, USA : Association for Computational Linguistics, 2005. – P. 65-72. – URL: https://aclanthology.org/W05-0909 (date accessed: 21.02.2024).

36. Benchmark datasets driving artificial intelligence development fail to capture the needs of medical professionals / K. Blagec, J. Kraiger, W. Frühwirt, M. Samwald // Journal of Biomedical Informatics. – 2023. – Vol. 137. – P. 104274.

37. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding / A. Wang, A. Singh, J. Michael [et al.]. – Text : electronic // International Conference on Learning Representations. – 2018. – URL: https://openreview.net/forum?id=rJ4km2R5t7 (date accessed: 30.01.2024).

38. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems / A. Wang, Y. Pruksachatkun, N. Nangia [et al.]. – Text : electronic // Advances in Neural Information Processing Systems 32. – Vancouver, Canada : Curran Associates, Inc., 2019. – Vol. 5. – P. 3232-3243. – URL: https://proceedings.neurips.cc/paper_files/paper/2019/hash/4496bf24afe7fab6f046bf4923da8de6-Abstract.html (date accessed: 30.01.2024).

39. RussianSuperGLUE: A Russian Language Understanding Evaluation Benchmark / T. Shavrina, A. Fenogenova, E. Anton [et al.]. – Text : electronic // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). – Online : Association for Computational Linguistics, 2020. – P. 4717-4726. – URL: https://aclanthology.org/2020.emnlp-main.381 (date accessed: 30.01.2024).

40. Hoang, H. On-the-Fly Fusion of Large Language Models and Machine Translation / H. Hoang, H. Khayrallah, M. Junczys-Dowmunt arXiv:2311.08306 [cs]. – 2023. – URL: http://arxiv.org/abs/2311.08306 (date accessed: 19.02.2024). – Text : electronic.

41. Trellix Doc Portal. – URL: https://docs.trellix.com/ (date accessed: 01.03.2024). – Text : electronic.

42. Dr.Web user documentation. – URL: https://download.drweb.com/doc/ (date accessed: 01.03.2024). – Text : electronic.

43. Kaspersky Online Help. – URL: https://support.kaspersky.ru/help/ (date accessed: 01.03.2024). – Text : electronic.

44. IBM Documentation. – URL: https://www.ibm.com/docs/ (date accessed: 01.03.2024). – Text : electronic.

45. Mixtral of Experts / A. Q. Jiang, A. Sablayrolles, A. Roux, [et al.] arXiv:2401.04088 [cs]. – 2024. – URL: http://arxiv.org/abs/2401.04088 (date accessed: 04.05.2024). – Text : electronic.

46. Llama 2: Open Foundation and Fine-Tuned Chat Models / H. Touvron, L. Martin, K. Stone, [et al.] arXiv:2307.09288 [cs]. – 2023. – URL: http://arxiv.org/abs/2307.09288 (date accessed: 13.03.2024). – Text : electronic.

47. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks / P. Lewis, E. Perez, A. Piktus [et al.]. – Text : electronic // Advances in Neural Information Processing Systems. – Curran Associates, Inc., 2020. – Vol. 33. – P. 9459-9474. – URL: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html (date accessed: 19.05.2024).

48. Kıyak, Y. S. A Prompt for Generating Script Concordance Test Using ChatGPT, Claude, and Llama Large Language Model Chatbots / Y. S. Kıyak, E. Emekli. – Text : electronic // Revista Española de Educación Médica. – 2024. – Vol. 5. – № 3. – URL: https://revistas.um.es/edumed/article/view/612381 (date accessed: 19.05.2024).