ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

Национальный исследовательский университет «Высшая школа экономики»

Факультет гуманитарных наук

Образовательная программа «Фундаментальная и компьютерная лингвистика»

КУРСОВАЯ РАБОТА

На тему «Выбор промежуточного языка в диалоговых задачах»

Тема на английском «Intermediate Language Choice in Chat-related Tasks»

Студент 2 курса группы №213

Хайрутдинов Тагир Рамилевич
(Ф.И.О.)

Научный руководитель

Сериков Олег Алексеевич
(Ф.И.О)

Приглашенный преподаватель

Научный сотрудник Института

Искусственного Интеллекта
(должность, звание)

Научный консультант

Протасов Виталий Павлович
(Ф.И.О)

Младший научный сотрудник Института

Искусственного Интеллекта

(должность, звание)

Москва, 2023 г.

Оглавление

1. Введение

1.1. Введение в компьютерную лингвистику и диалоговые задачи

Компьютерная лингвистика является разделом искусственного интеллекта, который изучает и разрабатывает методы и модели для обработки естественного языка с использованием компьютерных технологий. В последние годы диалоговые системы и задачи, связанные с обработкой диалогов, стали особенно важными в компьютерной лингвистике.

Диалоговые задачи охватывают широкий спектр приложений, включая виртуальных ассистентов, чат-боты, системы автоматического перевода и многое другое. Они представляют собой вызов в области обработки естественного языка из-за своей сложности и неоднозначности. Взаимодействие с пользователем через диалог требует понимания контекста, распознавания интентов и генерации естественных ответов, что требует разработки эффективных и точных моделей.

В данной работе мы сосредоточимся на одном из ключевых аспектов диалоговых задач - выборе промежуточного языка. Промежуточный язык - это язык, на который исходный диалоговый набор данных переводится перед обучением моделей машинного обучения. Выбор правильного промежуточного языка может оказать значительное влияние на производительность моделей, поскольку перевод может потерять или изменить смысл исходных диалогов. Это является важным аспектом исследования в области компьютерной лингвистики и требует более подробного рассмотрения.

1.2. Актуальность выбора промежуточного языка в диалоговых задачах

Выбор промежуточного языка в диалоговых задачах является актуальным вопросом, имеющим свою мотивацию и цель. В нашем исследовании мы стремимся расширить доступные корпусы для языков, которые имеют меньше ресурсов, чем, например, английский. Использование промежуточного языка позволяет нам создать переводы на эти языки и изучить, как результаты моделей будут изменяться при переводе на них.

При переводе диалоговых данных на промежуточный язык возникает ряд сложностей, таких как сохранение смысла и учет различий в языковых особенностях и лингвистической структуре. Неправильный выбор промежуточного языка может привести к искажению или потере информации, что неоднозначно сказывается на производительности моделей.

Решение этой проблемы требует внимательного анализа и сравнения различных методов и инструментов для перевода, а также исследования влияния промежуточного языка на различные архитектуры моделей машинного обучения. Такое исследование поможет нам лучше понять, как выбор промежуточного языка влияет на производительность моделей в диалоговых задачах и как можно улучшить этот процесс.

1.3. Цель и задачи исследования

Целью данной работы является исследование влияния выбора промежуточного языка на производительность моделей в диалоговых задачах. Для достижения этой цели мы ставим перед собой следующие задачи:

1. Провести анализ и сравнение различных методов перевода диалоговых данных на промежуточный язык.
2. Изучить влияние промежуточного языка на производительность различных архитектур моделей машинного обучения, таких как fastText, mT5-large, mT5-base, BERT, XLM-RoBERTa и BLOOM.
3. Оценить стабильность мультиязычных моделей на разных переводах и выявить наиболее подходящий промежуточный язык для диалоговых задач.

Результаты данного исследования могут быть полезными для разработчиков диалоговых систем и специалистов в области компьютерной лингвистики, чтобы принимать более обоснованные решения при выборе промежуточного языка и обучении моделей в диалоговых задачах.

2. Обзор литературы

2.1. Обзор существующих методов и подходов в диалоговых задачах

В последние годы диалоговые системы и задачи, связанные с обработкой диалогов, получили значительное внимание исследователей в области компьютерной лингвистики. Различные методы и подходы были разработаны для эффективного решения диалоговых задач.

Одним из наиболее распространенных подходов является использование моделей глубокого обучения, таких как рекуррентные нейронные сети (RNN) и трансформеры. Рекуррентные нейронные сети позволяют моделировать контекст и последовательность диалоговых сообщений, сохраняя информацию о предыдущих взаимодействиях.

Трансформеры, в свою очередь, основаны на механизмах внимания и способны обрабатывать контекстную информацию более эффективно.

Также существует множество работ, посвященных генерации ответов в диалогах с использованием генеративных моделей, таких как генеративные состязательные сети (GAN) и вариационные автоэнкодеры (VAE). Генеративные модели способны генерировать естественные и качественные ответы, учитывая контекст и интенты.

Другой важной темой в диалоговых задачах является извлечение и классификация интентов, то есть определение намерений пользователя на основе его высказывания. Для этого применяются различные алгоритмы машинного обучения, такие как методы на основе правил, статистические методы и глубокое обучение.

2.2. Роль промежуточного языка в диалоговых задачах

Промежуточный язык играет важную роль в диалоговых задачах, связанных с машинным переводом и обработкой естественного языка. Выбор промежуточного языка может существенно влиять на производительность моделей. Перевод диалоговых данных на промежуточный язык может привести к потере или изменению смысла исходных диалогов, что негативно сказывается на точности и качестве моделей.

Для достижения наилучших результатов в диалоговых задачах необходимо тщательно выбирать промежуточный язык, учитывая его лингвистические особенности и схожесть с исходным и целевым языками. Также важно учитывать стабильность мультиязычных моделей на различных переводах и находить оптимальный баланс между точностью и производительностью.

Таким образом, анализ роли промежуточного языка в диалоговых задачах является важной задачей в области компьютерной лингвистики и требует дальнейшего изучения и исследования.

3. Основные концепции и методы

3.1. Трансформеры в машинном обучении

3.1.1. Введение в трансформеры и описание их преимуществ

Трансформеры являются одной из ключевых архитектур в области обработки естественного языка. В отличие от рекуррентных нейронных сетей, трансформеры не

используют рекуррентные связи и могут эффективно обрабатывать контекстную информацию. Они позволяют моделировать длинные зависимости между словами и достигают высокой производительности в различных задачах, таких как машинный перевод и генерация текста.

3.1.2. Архитектура трансформера

Основой архитектуры трансформера является механизм внимания, который позволяет модели фокусироваться на различных частях входной последовательности при выполнении операций. Это позволяет эффективно учитывать контекст и устанавливать важность каждого элемента в процессе обработки.

Трансформер состоит из нескольких слоев, где каждый слой имеет два основных подкомпонента: многозаголовочное внимание и полносвязная нейронная сеть. Многозаголовочное внимание позволяет модели учиться сосредотачиваться на разных аспектах входных данных и строить более глубокие представления. Полносвязная нейронная сеть предоставляет нелинейные преобразования и добавляет дополнительные слои глубины к модели.

Трансформеры также широко используют механизм самообучения, известный как «Attention is all you need» (Vaswani et. al. 2017). Это позволяет моделям обучаться эффективно на больших объемах данных без необходимости предварительного обучения на других задачах.

3.1.3. Применение трансформеров в диалоговых задачах

Трансформеры успешно применяются в различных диалоговых задачах, таких как диалоговые системы, генерация ответов и классификация интентов. В диалоговых системах трансформеры позволяют моделировать контекст и последовательность диалоговых сообщений, учитывая предыдущие взаимодействия и интенты пользователя. Это помогает создавать более качественные и естественные ответы.

Для генерации ответов трансформеры могут быть обучены в условной генеративной модели, где они генерируют последовательности текста на основе контекста диалога. Это позволяет моделям генерировать ответы, учитывая семантику и структуру предыдущих сообщений.

Также трансформеры могут быть использованы для классификации интентов в диалоговых задачах. Модель может классифицировать пользовательское высказывание на основе его смысла и цели.

3.2. Эмбеддинги в машинном обучении

3.2.1. Роль эмбеддингов в представлении слов и текстовых данных

Эмбеддинги представляют собой векторные представления слов и текстовых данных, которые позволяют моделям машинного обучения работать с текстом эффективно. Векторные представления слов помогают моделям улавливать семантическую близость между словами и выражать их значимость в контексте.

3.2.2. Описание методов эмбеддингов и их особенности

Существует несколько методов для создания эмбеддингов, включая Word2Vec (Mikolov et. al. 2013), GloVe (Pennington et. al. 2014) и FastText (Bojanowski et. al. 2016). Word2Vec основан на обучении нейронной сети на больших текстовых корпусах и позволяет получить плотные векторные представления слов. GloVe использует матрицу совместной встречаемости слов для получения векторных представлений, учитывая глобальную статистику. FastText является расширением Word2Vec и учитывает также морфологическую информацию слов.

Каждый из этих методов имеет свои особенности и применяется в различных сценариях в зависимости от размера корпуса, доступной информации и задачи обработки текста.

3.2.3. Применение эмбеддингов в диалоговых задачах

Эмбеддинги широко используются в диалоговых задачах для представления слов и текстовых данных. В диалоговых системах, эмбеддинги позволяют моделям обрабатывать текстовые входы и создавать более качественные ответы.

Также эмбеддинги могут быть использованы для сравнения семантической близости между предложениями и классификации интентов в диалогах. Модели машинного обучения могут использовать эмбеддинги для выявления схожих или соответствующих высказываний и принятия решений на основе этой информации.

3.3. Модели, используемые в работе

3.3.1. Описание архитектуры fastText

FastText - это метод для эффективного обучения эмбеддингов и классификации текста. Он основан на идее использования n-грамм для представления слов и текстовых данных. FastText позволяет создавать более гибкие и детализированные векторные представления, учитывая морфологическую информацию.

Согласно статье Enriching Word Vectors with Subword Information (Bojanowski et. al. 2017), модель представляет собой простую нейронную сеть с единственным скрытым слоем. Входной текст, представленный в виде мешка слов (Bag-of-Words, BOW), проходит через первый слой, где он подвергается преобразованию в векторы слов. Затем полученные векторы усредняются по всему тексту и сводятся к единственному вектору. В скрытом слое оперируются *n_words * dim* параметрами, где *dim* представляет размерность вектора слов, а *n_words* - это размер используемого словаря для текста. После процесса усреднения получается один вектор, который подвергается классификации при помощи популярного метода - применения функции softmax для линейного отображения входных данных первого слоя на последний. Для этого линейного отображения используется матрица размерности *dim * n_output*, где *n_output* представляет собой количество результирующих классов, а *dim* - размерность пространства.

3.3.2. Описание архитектуры mT5

Прошлая модель T5 (Text-to-Text Transfer Transformer) использовала унифицированный формат преобразования текста в текст и масштабирование для достижения самых современных результатов в широком спектре англоязычных задач NLP. (Xue et. al. 2020)

mT5 (Multilingual Transformer 5) - это мультиязычная модель на основе трансформера, которая обучается на параллельных корпусах текстов на разных языках. Модель является мощным инструментом для мультиязычных диалоговых систем и позволяет обрабатывать тексты на разных языках с высокой точностью. В данной работе используется 2 варианта модели - mT5-base и mT5-large. Они имеют разное количество параметров, что сказывается на результатах, полученных с их помощью.

3.3.3. Описание архитектуры BERT

BERT (Bidirectional Encoder Representations from Transformers) - это модель на основе трансформера, которая предобучается с использованием комбинации цели моделирования замаскированного языка и предсказания следующего предложения на большом корпусе, включающем Toronto Book Corpus (Книжный корпус Торонто) и Википедию (Devlin et. al. 2018). Она позволяет создавать контекстуальные эмбеддинги слов и текстовых данных, учитывая их контекст и семантику.

В отличие от прошлых моделей языкового представления, BERT разработан для предварительного обучения глубоких двунаправленных представлений на основе неразмеченного текста, позволяя учитывать как левый, так и правый контекст на всех уровнях модели. В результате чего предварительно обученную модель BERT можно легко дообучать, добавляя только один дополнительный выходной слой, чтобы создавать передовые модели для широкого спектра задач, таких как вопросно-ответная система и языковой вывод, без существенных модификаций архитектуры, специфичных для задачи.

3.3.4. Описание архитектуры XLM-RoBERTa

XLM-RoBERTa является многоязычной версией модели RoBERTa. Она предварительно обучается на 2,5 терабайтах отфильтрованных данных CommonCrawl, содержащих информацию на 100 языках (Conneau et. al. 2019).

RoBERTa - это модель трансформера, предварительно обученная на большом корпусе в автономном режиме (Liu et. al. 2019). Это означает, что она была предварительно обучена только на необработанных текстах без какой-либо разметки со стороны людей с автоматическим процессом создания входных данных и меток из этих текстов, поэтому она может использовать большое количество общедоступных данных.

Модель была предварительно обучена с использованием метода маскированной языковой модели (Masked language modeling, MLM): берется предложение, в котором случайным образом маскируется 15% слов во входных данных, после чего это маскированное предложение проходит через всю модель, а модель должна предсказать маскированные слова. Это отличается от традиционных рекуррентных нейронных сетей (RNN), которые обычно видят слова одно за другим, или от авторегрессионных моделей, таких как GPT, которые внутренне маскируют будущие токены. Это позволяет модели изучать двунаправленное представление предложения.

3.3.5. Описание архитектуры BLOOM

BLOOM (BigScience Large Open-science Multilingual Language Model) - это масштабная мультиязычная модель обработки естественного языка, разработанная в рамках проекта BigScience (Scao et. al. 2022). Она основана на архитектуре трансформера и обучена на огромных наборах данных для достижения высокой производительности на разных языках. В результате модель способна выводить связный текст на 46 языках, включая 13 языков программирования, который едва отличим от текста, написанного человеком. Кроме того, BLOOM может быть настроена на выполнение текстовых задач, для которых она не была явно обучена, путем представления их в виде задач генерации текста.

4. Методология

4.1. Описание задачи и датасета Dialogue Breakdown Detection Challenge

Установление доверия пользователей к диалоговым агентам требует безупречного и последовательного обмена диалогами, однако агенты могут легко утратить контекст беседы и порождать неподходящие высказывания. В таких случаях возникают разрывы в диалоге, когда высказывания агента препятствуют продолжению разговора пользователями. Создание систем для обнаружения разрывов в диалоге позволяет агентам адекватно восстановиться или полностью избежать таких разрывов. В данной работе мы исследуем применение различных моделей машинного обучения для улучшения обнаружения разрывов в диалоге. Мы демонстрируем эффективность этих методов на задаче обнаружения разрывов в диалоге в рамках соревнования Dialogue Breakdown Detection Challenge 4 (DBDC) (Higashinaka et. al. 2016).

Для задачи обнаружения разрывов в диалоге существует незначительное количество общедоступных данных, поэтому все эксперименты мы проводим на наборе данных, распространяемом в рамках задания по обнаружению разрывов в диалоге на английском языке. Этот набор данных состоит из 411 диалогов на английском языке между пользователями и разговорными агентами, каждый из которых содержит от 10 до 20 высказываний. Поскольку мы предсказываем разрывы только в диалогах агентов, у нас есть примерно 2000 обучающих примеров и примерно такое же количество примеров для

валидации. Для всех экспериментов мы используем предоставленное разделение на обучающую, валидационную и тестовую выборки.

Для того, чтобы продемонстрировать разрыв в диалоге, обратимся к примеру от организаторов соревнования:

S1: *It's nice to go shopping alone.*

U1: *I agree. It's nice.*

S2: *Shopping takes time.*

U2: *Window shopping is also fun.*

S3: *It's fun to go shopping with someone else.* (Higashinaka et. al. 2016)

Здесь системные высказывания имеют префикс S, а пользовательские - U. Контекст диалога - от S1 до U2, а целевое высказывание для обнаружения разрыва диалога - S3 (подчеркнуто). В этом примере S3, скорее всего, приведет к прерыванию диалога, поскольку S3 противоречит S1. Следовательно, детектор, который классифицирует это как нарушение диалога, будет, *вероятно*, считаться точным. Мы говорим "вероятно", потому что решение человека относительно разрыва диалога очень субъективно, именно поэтому с точностью классифицировать высказывания не представляется возможным. Организаторы соревнования использовали несколько аннотаторов для аннотации диалога. При формирования датасета происходило голосование большинством голосов и распределение их вероятностей.

4.2. Выбор промежуточного языка

При выборе языков для перевода оригинального набора данных и дальнейшей оценки производительности моделей на этих языках учитывается несколько факторов. Одним из факторов является баланс между обученностью моделей на представленных языках и разнообразием выбранных языков.

Выбрав 5 языков: русский, африкаанс, немецкий, итальянский и испанский, мы старались достичь этого баланса. Включение русского языка, как одного из самых распространенных языков мира, имеет важное значение, учитывая его широкую аудиторию и значительное количество ресурсов для его обработки в области обработки естественного языка.

Добавление африкаанса, языка, происходящего из нидерландского и используемого в Южной Африке, дает возможность рассмотреть язык с уникальными лингвистическими

особенностями и культурным контекстом. Это также позволяет оценить производительность моделей на языках, которые имеют относительно небольшую лингвистическую исследовательскую базу.

Немецкий язык выбран как язык с большим количеством носителей и значительным влиянием в научных и технических областях. Он представляет собой сложный язык с богатыми грамматическими правилами и особенностями, что позволяет оценить способность моделей работать с более сложными лингвистическими структурами.

Выбор итальянского языка позволяет охватить романскую языковую группу, которая имеет сходство с другими романскими языками, такими как испанский и французский. Это может быть полезно для определения общих тенденций и понимания, как модели проявляют себя на языках одной языковой семьи.

Испанский язык, один из самых распространенных языков в мире, включен для оценки производительности моделей на языке с богатой историей и разнообразием диалектов. Он также предоставляет возможность сравнить результаты с другими романскими языками и оценить универсальность моделей на языках с общими чертами.

Таким образом, выбор этих пяти языков обусловлен стремлением найти баланс между широко используемыми языками, представителями различных языковых групп и языками с уникальными лингвистическими особенностями. Это помогает в обобщении результатов, понимании производительности моделей на разных языках и исследовании их универсальности и применимости.

4.3. Выбор способов перевода

Для достижения более объективных результатов в обучении моделей на переведенных данных, мы использовали два различных метода перевода: Google Translate API[1]и EasyNMT. Этот подход позволяет нам сравнить и оценить производительность и качество моделей, основанных на разных методах перевода.

Google Translate - это широко известный онлайн-сервис машинного перевода, предоставляемый компанией Google. Он использует разнообразные языковые модели и статистические методы для перевода текста между различными языками. Google Translate отличается высокой производительностью и широким охватом языков, что делает его подходящим для сравнительного анализа и оценки качества перевода.

[1] Google Translate API URL: https://console.cloud.google.com/tos?id=translate (дата обращения: 24.05.2023).

EasyNMT, как уже упоминалось ранее, представляет собой библиотеку машинного перевода, разработанную с открытым исходным кодом. Она предоставляет предобученные модели для различных языковых пар. Преимущество данного инструмента заключается в том, что это решение является открытым и воспроизводимым, в то время как API Google Translate является закрытым с точки зрения разработки решением. Использование разных по своей структуре решений помогает учесть влияние выбора метода перевода на производительность моделей и делает исследование более объективным и надежным.

4.4. Метрики и методы оценки производительности моделей

Для всесторонней оценки производительности моделей следует рассмотреть как точность (precision), так и полноту (recall). Метрика F1-score позволяет учесть оба этих аспекта. Взвешенный F1-score вычисляется путем нахождения среднего значения F1-score для каждого класса, учитывая баланс каждого класса. При взвешенном усреднении выходное среднее значение учитывает вклад каждого класса, взвешенный по числу примеров для данного класса.

5. Результаты и анализ

5.1. Анализ перевода оригинального набора данных

В данной главе мы проводим анализ перевода оригинального набора данных на различные языки, используя два разных метода перевода: Google Translate и EasyNMT. Нашей целью является оценка производительности моделей на переведенных данных и сравнение их с оригинальным набором данных.

Сначала рассмотрим общие средние результаты, полученные при переводе оригинального набора данных. В таблице 1 представлены средние значения метрики F1-score по типам перевода.

Таблица 1. Средние результаты по типам перевода

Тип перевода	Среднее значение F1-Weighted
EasyNMT	0.582085

Google	0.609079
Оригинальный датасет	0.639991

Из таблицы 1 видно, что в среднем переведенные датасеты показывают результаты хуже, чем оригинальный набор данных.

Обратившись к таблице 2, при сравнении двух моделей машинного перевода, EasyNMT и Google Translate, можно заметить, что результаты F1-Weighted находятся примерно на одном уровне для большинства языков. Некоторые языки, такие как немецкий (German) и итальянский (Italian), показывают схожие результаты для обеих моделей. Однако, разница в производительности между моделями есть, например, в случае испанского языка (Spanish), где модель Google Translate продемонстрировала немного лучший результат.

При ближайшем рассмотрении результатов для отдельных языков можно заметить некоторые различия в производительности моделей. Например, для русского языка (Russian) модель EasyNMT показала ниже средних результатов по сравнению с другими языками и моделью Google Translate. Это может быть связано с разницей в объеме и качестве данных обучения для русского языка в сравнении с другими языками, что может повлиять на способность модели к переводу данного языка.

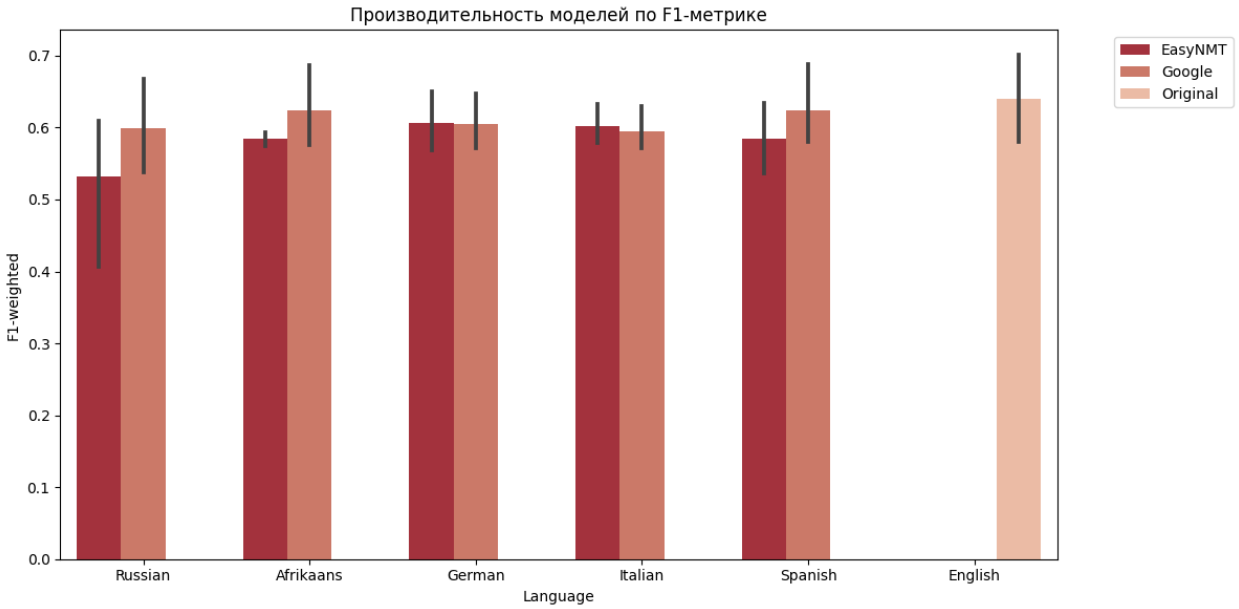
Различия в производительности моделей на разных языках могут быть связаны с доступным объемом данных обучения для каждого языка. Если при обучении использовалось меньше данных для определенного языка, это может отразиться на качестве перевода и привести к более низким результатам.

Таблица 2. Средние результаты обучения по типам перевода по каждому языку

	EasyNMT	Google Translate
Afrikaans	0.583933	0.623506

German	0.606151	0.604511
Italian	0.602684	0.594049
Russian	0.532478	0.598846
Spanish	0.585181	0.624485

Диаграмма 1. Средние результаты обучения по типам перевода по каждому языку



Рассмотрим таблицу 3, в которой представлены результаты нескольких моделей машинного обучения в разрезе различных типов перевода. Базовый датасет, который мы переводили, обозначается как Baseline, так как сравнение происходит непосредственно с этим набором данных на английском языке.

Из таблицы видно, что различные модели показывают разную производительность на данных датасетах. Например, модели BERT и XLM-RoBERTa достигают высоких результатов на большинстве датасетов, в то время как модель fastText демонстрирует более

низкие показатели. Это указывает на важность выбора оптимальной модели для достижения высоких результатов перевода.

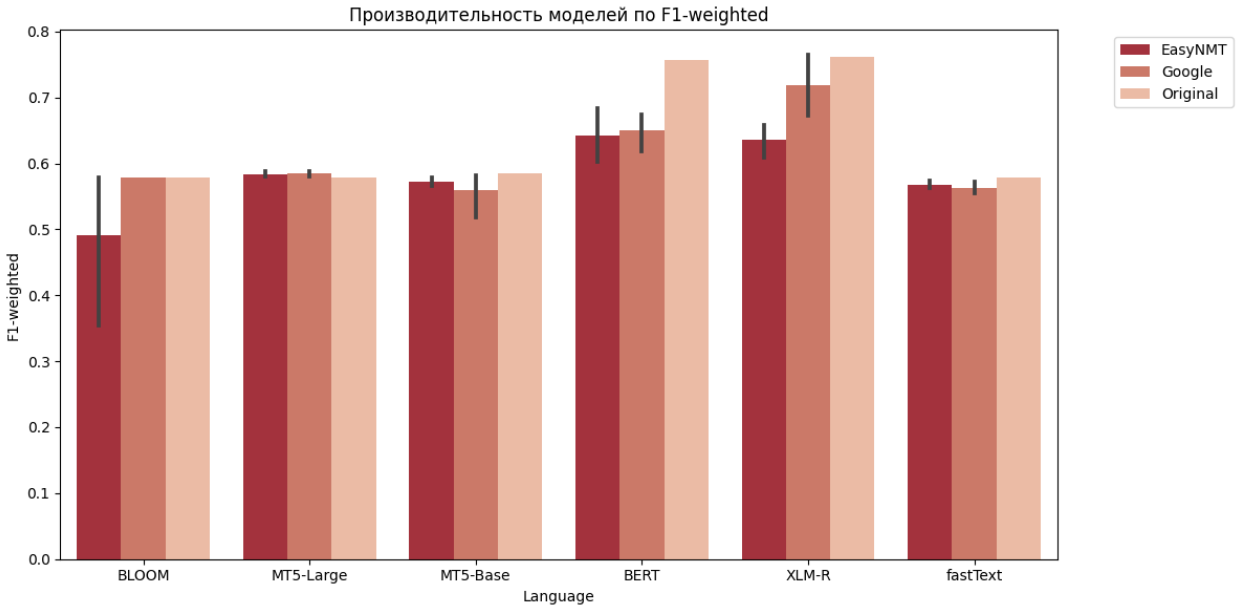
Также в таблице отражены результаты моделей, обученных на данных, полученных через различные способы перевода, такие как EasyNMT и Google Translate. Результаты показывают, что производительность моделей может различаться в зависимости от способа перевода. Например, модель BERT достигает лучших результатов при использовании данных, полученных через способ перевода EasyNMT, в то время как модель XLM-RoBERTa демонстрирует лучшие показатели на данных, полученных через способ перевода Google Translate.

Сравнивая переведенные датасеты с оригинальным и анализируя результаты моделей, можно обратить внимание на то, что большинство моделей показывают идентичные результаты, в то время как модели BERT и XLM-RoBERTa значительно превосходят их по качеству.

Таблица 3. Средние результаты обучения по типам перевода по каждой модели

	BERT	BLOOM	mT5-Base	mT5-Large	XLM-R	fastText
EasyNMT	0.642125	0.491234	0.572520	0.583369	0.635084	0.568180
Google Translate	0.649394	0.578648	0.559703	0.584575	0.719237	0.562919
Baseline	0.757161	0.578832	0.585417	0.578832	0.760987	0.578717

Диаграмма 2. Средние результаты обучения по типам перевода по каждой модели



В таблице 3 были приведены усредненные результаты по всем языкам, но для более детального анализа необходимо рассмотреть конкретные пары, представленные способом перевода, языком и моделью машинного обучения. Для этого обратимся к диаграмме 3.

Диаграмма 3. Результаты обучения по языкам по каждой модели

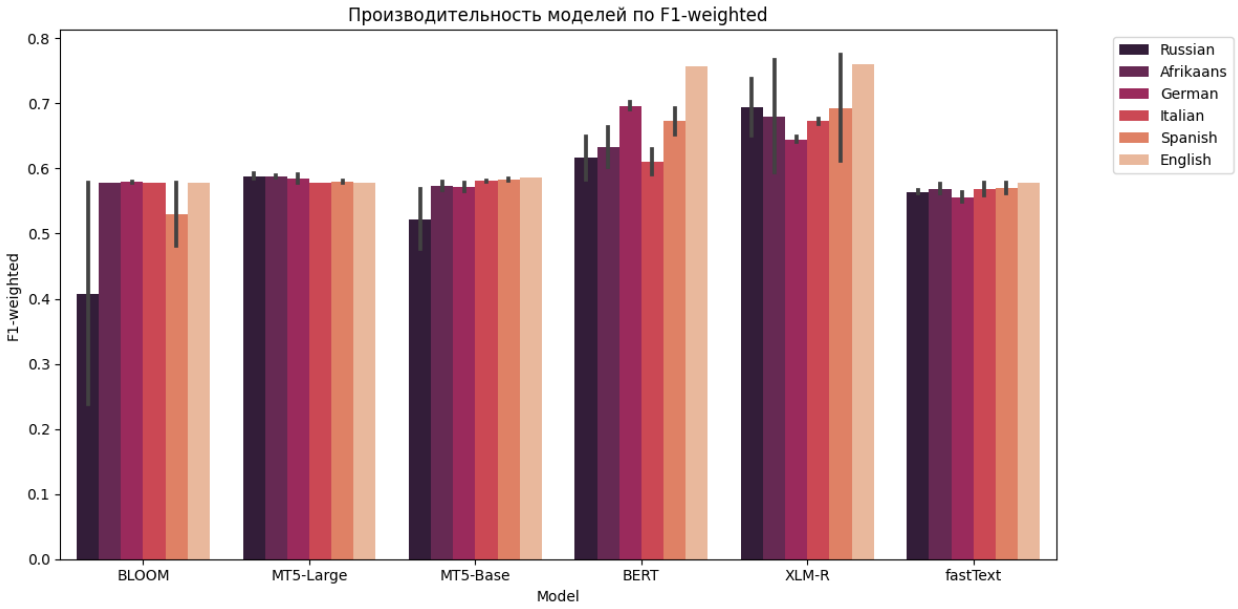


Таблица 4. Результаты обучения по языкам по каждой модели

	BERT	BLOOM	MT5-Base	MT5-Large	XLM-R	fastText
English	0.757161	0.578832	0.585417	0.578832	0.760987	0.578717
Afrikaans	0.633300	0.578871	0.573262	0.587571	0.680442	0.568871
German	0.696186	0.578941	0.571520	0.584867	0.644686	0.555786
Italian	0.610525	0.578871	0.580796	0.578871	0.672672	0.568464
Russian	0.616309	0.408017	0.522392	0.588131	0.694791	0.564331
Spanish	0.672477	0.530004	0.582588	0.580420	0.693212	0.570295

Большинство моделей ведут себя относительно стабильно на всех языках, исключением являются такие модели, как BLOOM, BERT, XLM-RoBERTa. Одна из причин нестабильности моделей в разрезе языков заключается в том, что не все модели во время предобучения имели равномерный объем данных на разных языках. Далее мы рассмотрим особенности каждой модели.

Как упоминалось ранее, модель BLOOM обучалась на 46 естественных языках и 13 языках программирования. В рамках нашего исследования мы использовали версию, которая имеет наименьшее число параметров - 559,214,592. Версия получила название BLOOM-560m, а из представленных в исследовании языков обучение происходило на английском (31.3% выборки) и испанском (11.1%), а 13.4% датасета были представлены кодом, который имеет преимущественно английский синтаксис.

Ожидалось, что языки, которые не представлены в обучающей выборке, будут иметь низкие результаты, в то время как модели, обученные на испанском и английском будут иметь высокие результаты.

Наши результаты подтвердили гипотезу относительно русского языка, поскольку результаты для него оказались достаточно низкими. Неожиданно было то, что испанский язык, в отличие от ожиданий, не проявил выдающуюся производительность. Напротив, по сравнению с другими языками, которые не использовались при обучении, испанский язык показал плохие результаты. Для иллюстрации результатов обратимся к таблице 5.

Таблица 5. Результаты обучения модели BLOOM по языкам и типам перевода

Язык	Тип перевода	F1-Weighted
Russian	EasyNMT	0.238124
Russian	Google	0.577910
Afrikaans	EasyNMT	0.578909
Afrikaans	Google	0.578832
German	EasyNMT	0.579051
German	Google	0.578832
Italian	EasyNMT	0.578909
Italian	Google	0.578832
Spanish	EasyNMT	0.481176
Spanish	Google	0.578832
English	Original	0.578832

Аналогично модели BLOOM, модель mT5 имеет несколько вариантов, отличающихся количеством параметров. В нашем исследовании мы использовали две версии этой модели - mT5-large и mT5-base, с 1.2 миллиарда и 580 миллионов параметров соответственно. Ожидаемо, вариант mT5-large показал более высокие результаты по сравнению с менее обученной альтернативой.

Интересным наблюдением, которое мы сделали при работе с данной моделью, является то, что результаты обучения на англоязычном датасете для mT5-large оказались менее удовлетворительными по сравнению с другими языками. Это отличается от результатов, полученных при использовании mT5-base, где исходный датасет показал наилучшую производительность. Для иллюстрации результатов обучения моделей mT5-base и mT5-large обратимся к таблицам 6 и 7 соответственно.

Таблица 6. Результаты обучения модели mT5-base по языкам и типам перевода

Язык	Тип перевода	F1-Weighted
Russian	EasyNMT	0.568323
Russian	Google	0.476461
Afrikaans	EasyNMT	0.566815
Afrikaans	Google	0.579708
German	EasyNMT	0.564994
German	Google	0.578046
Italian	EasyNMT	0.581883
Italian	Google	0.579708
Spanish	EasyNMT	0.580582
Spanish	Google	0.584594
English	Original	0.585417

Таблица 7. Результаты обучения модели mT5-large по языкам и типам перевода

Язык	Тип перевода	F1-Weighted
Russian	EasyNMT	0.591857
Russian	Google	0.584404
Afrikaans	EasyNMT	0.585484
Afrikaans	Google	0.589658
German	EasyNMT	0.578587
German	Google	0.591146
Italian	EasyNMT	0.578909

Italian	Google	0.578832
Spanish	EasyNMT	0.582008
Spanish	Google	0.578832
English	Original	0.578832

При работе с моделью BERT мы использовали менее обученную версию BERT-base, которая имеет 110 миллионов параметров, вместо более мощной вариации BERT-large с 340 миллионами параметров. Однако несмотря на это, мы получили результаты обучения, которые превышают большинство других моделей из нашей выборки. Здесь наблюдается явный дисбаланс в результатах для разных языков, особенно в пользу немецкого, испанского и английского языков. Для иллюстрации результатов обратимся к таблице 8.

Таблица 8. Результаты обучения модели BERT по языкам и типам перевода

Язык	Тип перевода	F1-Weighted
Russian	EasyNMT	0.583120
Russian	Google	0.649499
Afrikaans	EasyNMT	0.602816
Afrikaans	Google	0.663784
German	EasyNMT	0.702133
German	Google	0.690238
Italian	EasyNMT	0.629538
Italian	Google	0.591512
Spanish	EasyNMT	0.693017
Spanish	Google	0.651938
English	Original	0.757161

Модель XLM-RoBERTa продемонстрировала превосходные результаты по сравнению с другими моделями. Обратимся к таблице 9 для иллюстрации результатов. Аналогично модели BERT, она была предварительно обучена на всех языках, представленных в выборке. В нашем исследовании мы использовали наименьшую версию модели, содержащую 250 миллионов параметров. Одной из отличительных особенностей этой модели является то, что она была предварительно обучена исключительно на необработанных текстах без какой-либо человеческой разметки. Это также означает, что при предварительном обучении использовалось множество доступных общедоступных данных.

Возвращаясь к необходимости использования разных моделей для перевода, хочется сравнить результаты, полученные на африкаанс и испанском языке. Модели, обученные на датасетах, переведенных с помощью Google Translate, показывают результаты выше, чем модели, обученные на английском языке, в то время как результаты с переводом с помощью EasyNMT демонстрируют результаты ниже среднего. Модель перевода Google Translate, например, постоянно обучается, поэтому важно использовать в работе продвинутые модели.

Таблица 9. Результаты обучения модели XLM-RoBERTa по языкам и типам перевода

Язык	Тип перевода	F1-Weighted
Russian	EasyNMT	0.651340
Russian	Google	0.738242
Afrikaans	EasyNMT	0.593762
Afrikaans	Google	0.767123
German	EasyNMT	0.649060
German	Google	0.640312
Italian	EasyNMT	0.668997
Italian	Google	0.676348
Spanish	EasyNMT	0.612263

Spanish	Google	0.774161
English	Original	0.760987

Модель fastText показала относительно низкие значения метрики F1-Weighted по сравнению с другими рассмотренными моделями. Во-первых, fastText использует метод n-грамм для учета морфологической информации, а при работе с языками с более сложной и разнообразной морфологией, например на немецком и итальянском, эта особенность может быть менее эффективной для достижения высоких результатов.

Данная модель имеет ограниченную сложность с одним скрытым слоем. По сравнению с более сложными моделями, такими как BERT и XLM-RoBERTa, у которых больше параметров и которые обучаются на более разнообразных и объемных данных, fastText может быть менее способной улавливать сложные зависимости и контексты в тексте.

Стоит отметить, что размер используемого словаря в fastText также ограничен. Это может снижать способность модели учесть все вариации слов и контексты в тексте, особенно для языков с большим числом словоформ и лексическим разнообразием. Обратимся к таблице 10 для иллюстрации результатов.

Таким образом, несмотря на преимущества в учете морфологии и создании эффективных эмбеддингов, fastText может оказаться менее мощной по сравнению с более сложными моделями, такими как BERT и XLM-RoBERTa, в контексте классификации текста на разных языках. Представленные в таблице результаты подтверждают эту тенденцию, где fastText демонстрирует относительно низкие значения метрики F1-Weighted.

Таблица 10. Результаты обучения модели fastText по языкам и типам перевода

Язык	Тип перевода	F1-Weighted
Russian	EasyNMT	0.562101
Russian	Google	0.566561
Afrikaans	EasyNMT	0.575810

Afrikaans	Google	0.561931
German	EasyNMT	0.563081
German	Google	0.548491
Italian	EasyNMT	0.577868
Italian	Google	0.559060
Spanish	EasyNMT	0.562038
Spanish	Google	0.578552
English	Original	0.578717

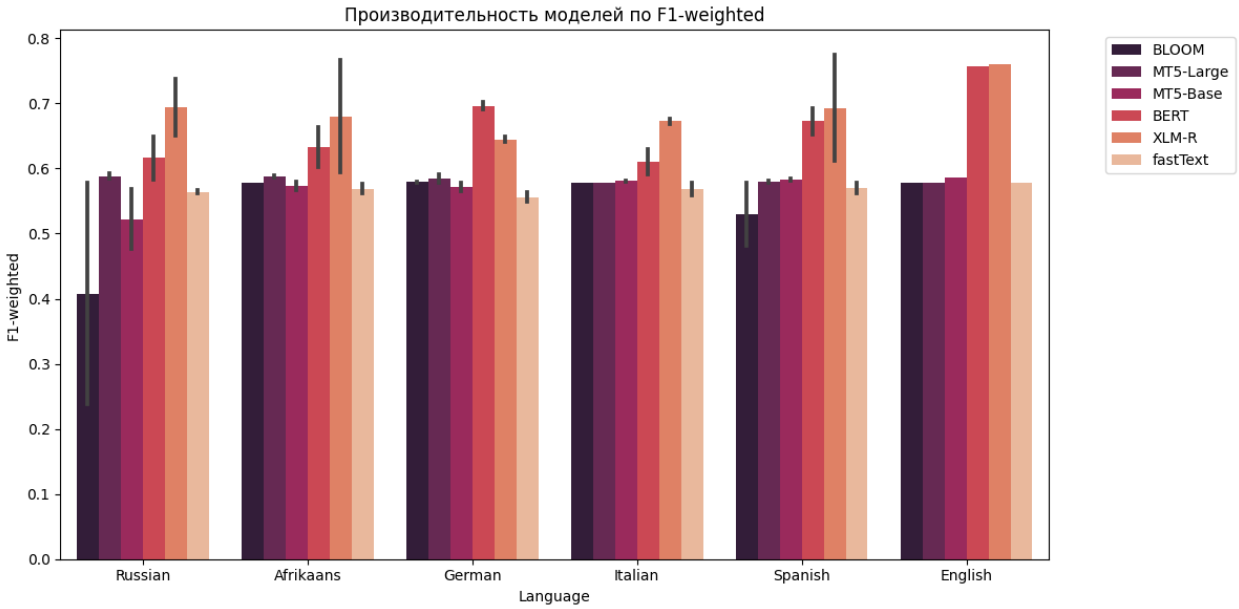
5.2. Исследование стабильности мультиязычных моделей на разных переводах

Взяв несколько языков для перевода датасета, мы могли рассмотреть и сравнить различные языковые группы и проверить, насколько модели эффективно работают на каждом языке. Каждый язык имеет свои уникальные лингвистические особенности, словарь, грамматические правила и семантику, поэтому разные модели могут показывать разную производительность в зависимости от языка.

Такой подход позволяет оценить общую способность моделей обрабатывать разнообразные языки и помогает исследователям и разработчикам понять, насколько модели универсальны и применимы в разных языковых средах. Также это может помочь определить, на каких языках модели показывают наилучшую производительность и где требуется дальнейшее улучшение.

Обратившись к диаграмме 4, можно заметить некоторые тенденции. Все языки имеют отличные результаты на моделях BERT и XLM-RoBERTa, демонстрируя некую стабильность на остальных моделях. Исключением является русский язык, который с его грамматическими и семантическими особенностями является самым нестабильным языком в нашей выборке.

Диаграмма 4: Результаты обучения по языкам по каждой модели



6. Заключение

6.1. Основные выводы исследования

В ходе данного исследования был проведен сравнительный анализ нескольких мультиязычных моделей для классификации текста на разных языках. В рамках этого анализа были рассмотрены модели BLOOM, mT5, BERT, XLM-RoBERTa и fastText, и их производительность была оценена на различных языковых датасетах.

На основе полученных результатов можно сделать следующие основные выводы:

1. Модели mT5-large и XLM-RoBERTa демонстрируют наилучшую производительность по сравнению с другими моделями в данном исследовании. Они достигают высоких значений метрики F1-Weighted на большинстве рассмотренных языков.

2. Модель BERT-base также показывает хорошие результаты и превосходит другие модели для некоторых языков, таких как немецкий, испанский и английский. Однако применение более мощной версии модели BERT-large не привело к существенному улучшению производительности.

3. Модель fastText, несмотря на свои преимущества в учете морфологической информации и создании эффективных эмбеддингов, демонстрирует относительно низкую

производительность по сравнению с другими моделями. Это может быть связано с ее ограничениями в улавливании сложных зависимостей и контекстов в тексте, особенно для языков с более сложной морфологией.

4. Результаты обучения моделей на разных языках показывают различную производительность в зависимости от конкретного языка. Некоторые модели, такие как BERT и XLM-RoBERTa, обладают хорошей универсальностью и демонстрируют стабильную производительность на большинстве языков, в то время как другие модели могут быть более чувствительны к лингвистическим особенностям конкретных языков.

5. При рассмотрении автоматического перевода текста необходимо учитывать различие в качестве перевода между разными методами. Например, в данном исследовании не была проведена специфическая оценка качества перевода с использованием EasyNMT и Google Translate. Однако стоит отметить, что методы машинного перевода, такие как Google Translate, широко применяются и обладают собственными наборами данных и подходами, которые могут влиять на их производительность.

В целом, результаты данного исследования предоставляют ценную информацию о производительности мультиязычных моделей для классификации текста на разных языках. При разработке и применении таких моделей в реальных приложениях необходимо учитывать их особенности, а также различия в качестве перевода, которые могут возникать при использовании различных методов, таких как EasyNMT и Google Translate.

6.2. Практическое применение результатов

Наши исследовательские результаты имеют практическую значимость для различных областей, где требуется классификация текста на разных языках. Некоторые примеры практического применения результатов:

1. Межъязыковая классификация текста: Наши результаты помогают выбрать наиболее эффективную мультиязычную модель для классификации текста на разных языках. Организации и компании, занимающиеся обработкой текстовых данных на множестве языков, могут использовать наши рекомендации для оптимизации своих систем классификации.

2. Мультиязычный поиск и рекомендации: Модели, показавшие высокую производительность на разных языках, могут быть применены в системах мультиязычного

поиска и рекомендаций. Это позволит улучшить релевантность и точность результатов поиска и рекомендаций для пользователей разных языковых групп.

3. Автоматический перевод и обработка текста: Результаты исследования могут быть использованы для выбора подходящей мультиязычной модели при автоматическом переводе и обработке текста на разных языках. Это поможет улучшить качество переводов и обработки текста, особенно для языков с более сложной структурой и особенностями.

4. Развитие и улучшение мультиязычных моделей: Наши результаты могут служить отправной точкой для дальнейшего исследования и улучшения мультиязычных моделей. Изучение причин различной производительности моделей на разных языках может помочь в разработке новых методов и подходов для повышения их универсальности и качества классификации.

В целом, результаты нашего исследования предоставляют ценную информацию о производительности мультиязычных моделей на различных языках и могут быть использованы для улучшения различных приложений обработки текста на множестве языковых групп.

6.3. Ограничения и предложения для дальнейших исследований

В ходе данного исследования мы провели анализ и сравнение результатов работы различных моделей машинного обучения на разных языках с использованием одного исходного датасета. Однако, несмотря на полученные результаты, необходимо отметить ограничения данного подхода. Для дальнейшего развития исследования планируется рассмотреть следующие аспекты:

1. Множественные исходные датасеты: включение в анализ нескольких различных исходных датасетов на сравниваемых языках позволит уменьшить bias и получить более надежные выводы относительно эффективности моделей.

2. Использование параллельных корпусов: для более объективной оценки моделей рекомендуется использование параллельных корпусов, где доступны точные и надежные пары переводов между языками. Это позволит исключить искажения, связанные с качеством исходных переводов, и обеспечит более точное сравнение моделей.

3. Учет различных метрик оценки: в дополнение к использованным метрикам, таким как F1-Weighted, рекомендуется рассмотреть другие метрики, такие как BLEU или

METEOR для более всесторонней оценки качества перевода. Это позволит получить более полную картину эффективности моделей машинного обучения.

4. Расширение набора моделей: для более обширного сравнения моделей машинного обучения стоит рассмотреть использование различных типов и архитектур моделей. Это может включать нейронные сети с долгой краткосрочной памятью (LSTM), трансформеры и другие инновационные модели, которые могут демонстрировать лучшую производительность на конкретных языковых парах или типах перевода.

5. Учет контекста и специфических требований задачи перевода: при выборе наиболее подходящей модели машинного обучения необходимо учитывать контекст и специфические требования конкретной задачи. Иногда модели, которые показывают лучшие результаты на общих метриках, могут быть менее эффективными в конкретной предметной области или для определенного типа текстов.

Внедрение этих аспектов в будущих исследованиях позволит улучшить надежность и обобщаемость результатов, а также продвинуть развитие моделей машинного обучения при использовании промежуточного языка.

7. Литература

1. Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, Veselin Stoyanov Unsupervised Cross-lingual Representation Learning at Scale // arxiv.org . - 2019. - №https://arxiv.org/abs/1911.02116.
2. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin Attention Is All You Need // arxiv.org . - 2017. - №https://arxiv.org/abs/1706.03762.
3. BigScience Workshop: Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay,

Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas et al. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model // arxiv.org . - 2022. - №https://arxiv.org/abs/2211.05100.

4. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // arxiv.org . - 2018. - №https://arxiv.org/abs/1810.04805v2.
5. Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.
6. Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, Colin Raffel mT5: A massively multilingual pre-trained text-to-text transformer // arxiv.org . - 2020. - №https://arxiv.org/abs/2010.11934v3.
7. Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov Enriching Word Vectors with Subword Information // arxiv.org . - 2016. - №https://arxiv.org/abs/1607.04606v2.
8. Ryuichiro Higashinaka, Kotaro Funakoshi, Yuka Kobayashi, and Michimasa Inaba. 2016. The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics. In Proceedings of the Tenth International Conference on Language Resources

and Evaluation (LREC'16), pages 3146–3150, Portorož, Slovenia. European Language Resources Association (ELRA).

9. Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean Efficient Estimation of Word Representations in Vector Space // arxiv.org . - 2013. - №https://arxiv.org/abs/1301.3781.
10. Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov RoBERTa: A Robustly Optimized BERT Pretraining Approach // arxiv.org . - 2019. - №https://arxiv.org/abs/1907.11692.

8. Приложение

1. EasyNMT // GitHub URL: https://github.com/UKPLab/EasyNMT (дата обращения: 24.05.2023).
2. Google Translate API URL: https://console.cloud.google.com/tos?id=translate (дата обращения: 24.05.2023).
3. Intermediate Language Choice in Chat-related Tasks // GitHub URL: https://github.com/TagirRamilevich/Intermediate-Language-Choice-in-Chat-related-Tasks (дата обращения: 24.05.2023).