Федеральное государственное автономное образовательное
учреждение высшего образования
«Национальный исследовательский университет
«Высшая школа экономики»

Факультет компьютерных наук
Основная образовательная программа
Прикладная математика и информатика

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Исследовательский проект на тему
"Атаки на мультиязычные модели"

Выполнил студент группы 171, 4 курса,
Биршерт Алексей Дмитриевич

Руководитель ВКР:
К. т. н., Доцент
Департамент больших данных и информационного поиска
Артемова Екатерина Леонидовна

Москва 2021

Содержание

Аннотация

В мультиязычных сообществах по всему миру распространён феномен смешения кодов, когда человек использует в речи более одного языка внутри одного предложения. Мультиязычные языковые модели показывают впечатляющее качество для разных задач обработки естественного языка. Однако реальные данные со смешением кодов очень дороги в сборе и разметке. Мы представляем две адверсариальные атаки по методу серого ящика, чтобы оценить возможное качество мультиязычных моделей на входных данных со смешением языков внутри одного предложения. Дополнительно мы предлагаем метод адверсариального предобучения для защиты от атак такого рода.

В своей работе мы решаем задачу одновременного заполнения слотов и распознавания интентов с качеством 98% accuracy по интентам и 95% F1 меры по слотам; понижаем качество моделей с 78% до 16% по метрике semantic accuracy с помощью адверсариальной атаки; повышаем качество моделей с 8.8% до 20% по метрике semantic accuracy с помощью предложенного метода защиты.

Ссылка на гитхаб с проектом - https://github.com/birshert/attack-lang-models.

Ключевые слова—Одновременное заполнение слотов и распознавание интентов, адверсариальные атаки, мультиязычные языковые модели, адверсариальное обучение

There is a common phenomenon in multilingual societies all around the world code-mixing, it consists in mixing different languages inside one utterance. Multilingual models have demonstrated incredible performance in various natural language processing tasks. However, real code-mixing data is very expensive to collect and label. We present two gray-box adversarial attacks, build to evaluate multilingual language models capacity to work with code-mixing input data. Additionally we present an adversarial pretraining method to make the models more robust to attacks.

In our work we solve the joint slot-filling and intent recognition task with 98% intent accuracy and 95% slots F1 score; bring models performance down from 78% to 16% in semantic accuracy metric with adversarial attack; increase models performance from 8.8% to 20% in semantic accuracy metric with proposed protection method.

Github project link - https://github.com/birshert/attack-lang-models.

Keywords—Joint slot-filling and intent recognition, adversarial attacks, multilingual language models, adversarial training

1 Введение

Последние несколько лет стали прорывными в области мультиязычных моделей и их обобщающей способности для других языков [1, 2, 8, 13]. Огромные мультиязычные модели выучивают универсальные языковые представления, что помогает им демонстрировать удивительные способности к переносу знаний с одного языка на другой. Простое дообучение предобученных моделей для какой-либо задачи на языке с большим количеством данных позволяет достичь хорошего качества на других языках.

Однако простой перенос между языками недостаточен для систем обработки естественного языка для понимания мультиязычных пользователей. Во многих сообществах в мире достаточно часто явление смешения кодов. Смешение кодов — это процесс, когда человек спонтанно смешивает различные языки внутри одного предложения или фразы. Такой феномен может проявляться как в письменной, так и в устной речи. Таким образом, важно сделать языковую модель устойчивой к смешению языков, чтобы модель адекватно работала со входными данными.

Несмотря на то, что реальные данные со смешением кодов очень важны для оценки качества языковых моделей, такие данные очень тяжело собирать и размечать в большом количестве.

В своей работе мы предполагаем, что качество моделей на адверсариальных атаках может служить нижней оценкой на реальное качество модели. Если языковая модель успешно справляется с адверсариальными пертурбациями со смешением кодов, то и в реальной жизни она будет успешно обрабатывать данные от мультиязычных пользователей.

В своей работе мы:

- Решаем задачу одновременного детектирования намерений пользователя и заполнения слотов для диалоговых помощников с помощью мультиязычных языковых моделей.

- Предлагаем две адверсариальные атаки по методу серого ящика — во

время атаки мы имеем доступ к ошибке модели на заданных данных. Насколько нам известно, это одни из первых мультиязычных адверсариальных атак для вышеописанной задачи.

- Предлагаем метод адверсариального предобучения.

В результате работы мы ожидаем получить следующие результаты:

- Мультиязычные модели обучены решать задачу заполнения слотов и классификации интентов.
- Проведены две адверсариальные атаки на каждую модель и замерено качество моделей на адверсариальных данных.
- Оценено влияние метода адверсариального предобучения на качество моделей на тестовой выборке и после адверсариальных атак.

Все свои эксперименты мы будем проводить с современными мультиязычными моделями - m-BERT [2] и XLM-RoBERTa [1]. В качестве датасета мы будем использовать корпус MultiAtis++ [14].

Актуальность темы подтверждается повышенным интересом со стороны научного сообщества. После начала работы над исследованием вышло как минимум две статьи на эту тему — в середине марта 2021 года [7, 10].

2 Обзор литературы

2.1 Мультиязычные модели

Языки с небольшим количеством данных часто не могут предоставить достаточного размера датасета для обучения с учителем. Существует подход для борьбы с этим, который заключается в построении кросс-язычных представлений. Эти представления нужно дообучать для специфичной задачи на языке с большим количеством ресурсов, чтобы показывать хорошее качество на других, менее ресурсоёмких языках [6].

Вслед за успехом модели Трансформер [11], недавние мультиязычные модели такие как m-BERT [2] и XLM-RoBERTa [1] переносят парадигму «предобучение → дообучение под специфическую задачу» в мультиязычную область. Они предобучают энкодеры на основе архитектуры Трансформера на текстовых данных с различными задачами языкового моделирования. Затем эти предобученные энкодеры могут быть дообучены для конкретной задачи на ресурсоёмком языке для которого есть много размеченных данных. Это известно как кросс-язычный перенос знаний.

В одних недавних исследованиях кросс-язычного переноса знаний было показано, что качество модели на ранее не виденных тестовых языках сильно зависит от количества обучающих данных и размера контекста [8]. В [13] было показано, что m-BERT показывает очень сильную способность к кросс-язычному переносу знаний. m-BERT превосходит по качеству мультиязычные эмбеддинги в четырёх из пяти исследуемых задач без какой-либо информации о связи языков.

Более современная и более сложная модель XLM-RoBERTa [1] показывает лучшее, чем m-BERT качество, однако требует массивных объемов обучающих данных для хорошей работы. В своём исследовании авторы XLM-RoBERTa показывают, что их модель является самой сильной мультиязычной моделью на текущий момент.

m-BERT обучается на корпусах Wikipedia и Books, в то время как XLM-

RoBERTa обучается на CommonCrawl, который содержит для многих языков на несколько порядков больше данных.

2.2 Классификация интентов и заполнение слотов

Повсеместное использование виртуальных ассистентов постепенно становится ежедневной реальностью с ростом их популярности. Богатство возможностей и качество работы ассистента напрямую влияет на удобство его использования. Хорошие ассистенты будут привлекать всё больше людей, занимая доли рынка. Ключевым аспектом в работе виртуального помощника является правильная классификация интентов и заполнение слотов в запросах. Интент — это желаемый результат запроса пользователя. Слоты — это слова или наборы слов, которые содержат релевантную интенту информацию.

Из-за тесной корреляции между задачами заполнения слотов и классификации интентов обычно используется одна модель для одновременного решения обеих задач [12]. Актуальные подходы последнего времени используют модели на основе Трансформера, например BERT [2]. Одним из популярных датасетов для этой задачи является датасет MultiAtis++ [14].

2.3 Смешение кодов в адверсариальных атаках на мультиязычные модели

Первые статьи в тематике адверсариальных атак с использованием смешения кодов на мультиязычные модели, такие как m-BERT [2], Unicoder [5] и XLM-RoBERTa [1] вышли в середине апреля 2021 года. В своей работе мы опирались и рассматривали две такие статьи - [10] и [7].

Первое исследование [10] посвящено анализу качества моделей m-BERT, Unicoder и XLM-RoBERTa на датасете XNLI под влиянием адверсариальных атак. В своей работе авторы анализируют две адверсариальные атаки, одна word-level (замена слов в предложении на их эквиваленты из набора язы-

ков), вторая phrase-level (замена частей предложения с помощью построения выравниваний). Так же авторы предлагают метод адверсариального предобучения, который заключается в генерации дополнительных адверсариальных пертурбаций для обучающей выборки и обучения на ней.

Второе исследование [7] посвящено анализу метода аугментации данных для задачи заполнения слотов и классификации предложений. В своей работе авторы ставят перед собой цель улучшить перенос знаний на новый неизвестный язык для мультиязычной языковой модели. Аналогично первой статье, во втором исследовании анализ сконцентрирован на качестве моделей после аугментирования тренировочных данных с помощью смешения кодов. Авторы предлагают метод chunk-level атаки, которая заключается в сегментации предложений по спанам слотов и замене соответствующих одинаковых сегментов между предложениями на разных языках. С помощью такой атаки они атакуют обучающую выборку датасета MultiAtis++ [14] и добавляют полученную адверсариальную выборку к исходной обучающей выборке.

В первой статье было установлено, что проведенные адверсариальные атаки очень сильно ухудшили качество мультиязычных языковых моделей. В первую очередь это может быть связано с тем, что датасет XNLI сам по себе является сложным датасетом и изначальное качество моделей не высоко. С другой стороны стоит отметить, что в адверсариальных атаках из этого исследования использовалась схема «один основной язык, много встраиваемых языков». В то время как эта схема не может отражать реалистичные данные со смешением кодов, так как большинство людей, которые могут смешивать коды в речи или на письме, билингвы [9].

Во второй статье было установлено, что добавление аугментированных данных в обучающую выборку приводит к улучшению качества для модели m-BERT. Использование дополнительных данных положительно повлияло на низкоресурсные языки. Так же это положительно сказалось на качестве на языках с отличными от английского морфологическими структурами — китайским и японским.

2.4 Машинный перевод и выравнивание слов

Для машинного перевода в своей работе мы будем использовать [4]. Созданная авторами статьи модель обучалась на внушительном датасете из 7.5 миллиардов предложений для 100 языков. Данная модель основна на архитектуре Трансформера и способна переводить с любого на любой язык в пределе ста обучающих. На текущий момент это одна из самых сильных моделей для машинного перевода, которая успешно справляется с переводом на любые, даже ранее низкоресурсные, языки.

Для построения выравниваний между параллельными предложениями на разных языках мы будем использовать [3]. Оригинальный подход авторов статьи использует эмбеддинги от мультиязычной языковой модели m-BERT [2]. Среди результатов постулируется превосходство данного подхода над всеми остальными на текущий момент.

3 Основная часть

3.1 Обучение моделей на датасете MultiAtis++

В своей работе мы обучаем языковые модели решать задачу задачи одновременного детектирования намерений пользователя и заполнения слотов для диалоговых помощников, направленных на выполнение конкретной задачи. Эта задача заключается в классификации предложений и всех слов в предложении.

3.1.1 Датасет

В качестве датасета в своей работе мы выбрали датасет MultiAtis++ [14]. В этом датасете представлены семь языков из трёх языковых семей — Индо-Европейская (английский, немецкий, французский, испанский, португальский), Японо-рюкюская (японский) и Сино-тибетская (китайский). Датасет является параллельным корпусом для задачи классификации интентов и разметки слотов - в 2020 году он был переведён с английского языка на остальные шесть. В обучающей выборке содержится 4978 предложений для каждого языка, в тестовой 893 предложения для каждого языка.

Intent	atis_flight						
Utterance en	show	me	flights	from	montreal	to	orlando
Slot labels en	O	O	O	O	B-fromloc.city_name	O	B-toloc.city_name
Utterance de	Zeige	mir	Flüge	von	Montreal	nach	Orlando
Slot labels de	O	O	O	O	B-fromloc.city_name	O	B-toloc.city_name

Таблица 3.1: Пример объекта из датасета MultiAtis++. На примере представлен объект на английском и немецком языке.

Каждый объект в датасете состоит из предложения, меток слов в BIO формате и интента (Таблица (3.1)). Перед началом работы с датасетом мы произвели предварительную очистку — убрали из обучающей и тестовой выборок объекты, для которых на любом из семи языков количество слов и

количество слотов не совпадали. Таким образом, в обучающей выборке осталось 4884 объекта для каждого языка, в тестовой выборке 755 объектов для каждого языка. Для составления списка используемых слотов и интентов использовалась обучающая выборка на английском языке. Мы использовали 121 различную метку слотов и 23 различных метки интентов.

3.1.2 Архитектура модели

В своей работе мы решаем задачу одновременной классификации интентов и разметки слотов в предложении с помощью одной модели. Модель имеет два выхода, первый предсказывает интенты, второй предсказывает метки слов. В качестве рассматриваемых архитектур были выбраны модели m-BERT [2] и XLM-RoBERTa [1]. Обе эти модели являются одними из самых сильных мультиязычных моделей на текущий момент. Каждая из них предобучена на более чем ста языках.

Обозначим количество блоков Трансформера за L, размер скрытых представлений за H и количество голов с внутренним вниманием за A. Тогда в используемой нами модели m-BERT L = 12, H = 768, A = 12, а суммарное количество параметров 110 миллионов. В используемой нами модели XLM-RoBERTa L = 12, H = 768, A = 12, а суммарное количество параметров 270 миллионов.

3.1.3 Обучение

В своей работе мы будем сравнивать модели, обученные на всей обучающей выборке и только на части обучающей выборки на английском языке. Таким образом мы сможем проверить насколько устойчивы к нашим атакам модели с разными вариантами обучения.

Введем краткие обозначения для удобства — модели XLM-RoBERTa будут обозначаться как «xlm-r», модели m-BERT будут обозначаться как «m-bert». Если модель обучалась только на английской подвыборке, то мы будем добавлять в её название суффикс «en».

Каждая из моделей обучалась с одинаковыми гиперпараметрами - 10 эпох на обучающей выборке с длиной шага обучения 10^{-5} и размером батча в 64 объекта. В качестве функции ошибки использовалась кросс-энтропия:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\left[y\log\left(\hat{y}\right)\right] \tag{3.1}$$

В своей работе мы будем использовать следующие метрики качества:

- Доля предложений, в которых правильно классифицирован интент:

$$\textbf{Intent accuracy} = \#\text{sentences}\left[(I_{pred} = I_{true})\right] \tag{3.2}$$

- F1 мера для меток слотов (используется микро-усреднение по всем классам):

$$\textbf{Slots F1 score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{3.3}$$

- Доля предложений, в которых правильно классифицирован интент и верно классифицированы все слоты:

$$\textbf{Semantic accuracy} = \#\text{sentences}\left[(I_{pred} = I_{true}) \wedge (S_{pred} = S_{true})\right] \tag{3.4}$$

3.2 Адверсариальные атаки

В своей работе мы предлагаем два варианта gray-box адверсариальных атак — во время выполнения атаки мы имеем доступ к ошибке модели. Мы стремимся создать атаку такого рода, чтобы результирующая адверсариальная пертурбация предложения была как можно ближе к реалистичным предложениям со смешением кодов. Оценка качества на таких адверсариальных атаках может выступать в роли оценки снизу на качество соответствующих моделей в аналогичных задачах при наличии реального смешения кодов во входных данных.

Мы фокусируемся в основном на лексической части смешения кодов — когда некоторые слова заменяются за их аналоги из других языков. Во время атаки мы заменяем часть токенов в предложении на их эквиваленты из атакующих языков, метод определения эквивалентов зависит от типа атаки. Так как большинство людей, которые могут использовать смешение кодов в своей речи, билингвы, то в основном смешение кодов происходит между парой языков [9]. Таким образом, в своей работе мы предлагаем анализировать атаки состоящие во встраивании одного языка в другой.

3.2.1 Общий вид атаки

Общий принцип атаки одинаковый для обоих предлагаемых вариантов. Разница между методами заключается в способе генерации кандидатов на замену токену на $i-$ой позиции. В своей работе мы предлагаем следующий вид атаки — пусть мы имеем целевую модель, пару пример-метка и встраиваемый язык (Алгоритм (3.1)). Тогда мы перебираем токены в предложении в случайном порядке и стремимся заменить токен на его эквивалент из встраиваемого языка. Если это приведёт к увеличению ошибки модели, то мы заменяем токен на предложенного кандидата.

Algorithm 3.1 Общая схема адверсариальной атаки

Require: Пара пример-метка x, y; целевая модель $\mathcal{M}$; встраиваемый язык $\mathbb{L}$
Ensure: Адверсариальный пример x'

```
L_x = GetLoss(M, x, y)
for i in permutation(len(x)) do
    Candidates = GetCandidates(M, x, y, token_id = i)
    Losses = GetLoss(M, Candidates)
    if Candidates and max(Losses) > L_x then
        L_x = max(Losses)
        x, y = Candidates[argmax(Losses)]
    end if
end for
    return x
```

3.2.2 Word-level атака

Первый предлагаемый нами вариант атаки заключается в генерации эквивалентов из других языков с помощью перевода токенов на соответствующие языки (Алгоритм (3.2)). Атакуя таким образом, мы строим грубую оценку снизу, так как при атаке мы не учитываем контекста предложений и не учитываем многозначность слов. Этот вариант схож с атакой PolyGloss [10]. Примеры атаки на тестовую выборку для модели XLM-RoBERTa можно найти в таблицах (3.2),(3.3) и (3.4).

Для перевода слов на другие языки мы используем модель машинного перевода M2M 100 от компании Facebook [4]. Она содержит 418 миллионов параметров.

Псевдокод функции ExtendSlotLabels можно найти в приложении (Алгоритм (A.1)).

Utterance en	what	are	the	flights	from	tacoma	to	san	jose
Utterance adv	what	are	El	vuelos	from	tacoma	to	san	jose

Таблица 3.2: Пример 1 атаки модели XLM-RoBERTa (xlm-r) word-level атакой.

Utterance en	i	would	like	flight	information	from	phoenix	to	denver
Utterance adv	y	would	como	flight	Información	from	El Phoenix	para	El Denver

Таблица 3.3: Пример 2 атаки модели XLM-RoBERTa (xlm-r) word-level атакой.

Algorithm 3.2 Word-level атака

Require: Словарь переводов с исходного на встраиваемый язык $\mathbb{T}$
  function GETCANDIDATES($\mathcal{M}$, x, y, token_id)
    if x[token_id] in $\mathbb{T}[\mathbb{L}]$ **then**
      tokens = $\mathbb{T}[\mathbb{L}]$[x[token_id]]
      x[token_id] = tokens
      y[token_id] = ExtendSlotLabels(y[token_id], len(tokens))
    end if
    return x, y
  end function

Utterance en	what	are	the	flights	from	las	vegas	to	ontario
Utterance adv	what	sind	Die	flights	from	las	VEGAS	to	ontario

Таблица 3.4: Пример 3 атаки модели XLM-RoBERTa (xlm-r) word-level атакой.

3.2.3 Phrase-level атака

Второй предлагаемый нами вариант атаки заключается в генерации эквивалентов из других языков с помощью построения выравниваний между предложениями на разных языках (Алгоритм (3.3)). Одно предложение является переводом другого, для перевода можно использовать ту же модель машинного перевода [4], однако мы пользуемся тем, что у нас уже параллельный корпус. Кандидаты для каждого токена определяются как токены из предложения на встраиваемом языке, в которые был выровнен токен. Этот вариант атаки схож с атакой Bumblebee [10]. Примеры атаки на тестовую выборку для модели XLM-RoBERTa можно найти в таблицах (3.5),(3.6) и (3.7).

Для построения выравниваний мы используем модель awesome-align на основе m-BERT [3].

Utterance en	please	find	flights	available	from	kansas	city	to	newark
Utterance adv	encontre	find	flights	disponíveis	from	kansas	City	para	Newark

Таблица 3.5: Пример 1 атаки модели XLM-RoBERTa (xlm-r) phrase-level атакой.

Algorithm 3.3 Phrase-level атака

Require: Выравнивание предложения на исходном языке к предложению на целевом языке $\mathbb{A}$

function GetCandidates($\mathcal{M}$, x, y, token_id)
  if x[token_id] in $\mathbb{A}[\mathbb{L}]$ **then**
    tokens = $\mathbb{A}[\mathbb{L}]$[x[token_id]]
    x[token_id] = tokens
    y[token_id] = ExtendSlotLabels(y[token_id], len(tokens))
  end if
  return x, y
end function

Utterance en	what	are	the	flights	from	tacoma	to	san	jose
Utterance adv	cuáles	are	the	flights	from	tacoma	a	san	jose

Таблица 3.6: Пример 2 атаки модели XLM-RoBERTa (xlm-r) phrase-level атакой.

Utterance en	show	flights	saturday	evening	from	st.	louis	to	burbank
Utterance adv	show	flights	sábado	evening	from	St.	Louis	para	Burbank

Таблица 3.7: Пример 3 атаки модели XLM-RoBERTa (xlm-r) phrase-level атакой.

3.3 Метод адверсариального предобучения для защиты от адверсариальных атак

В своей работе мы предлагаем метод защиты от предложенных выше адверсариальных атак. Гипотеза заключается в том, что данный метод позволит увеличить качество не только на адверсариальных пертурбациях, но и на реальных данных со смешением кодов.

Предлагаемый нами метод адверсариального предобучения состоит из нескольких шагов:

1 Генерация выборки для задачи маскированного моделирования языка.

2 Дообучение тела мультиязычной модели на сгенерированной выборке в режиме предсказания маскированных токенов.

3 Загрузка дообученного тела модели перед началом обучения для задачи одновременного заполнения слотов и классификации интентов.

Algorithm 3.4 Генерация адверсариальной выборки

Require: Обучающая выборка датасета X, набор встраиваемых языков $\mathbb{L}_1, \dots \mathbb{L}_n$
Ensure: Адверсариальная выборка X'

```
X' = [ ]
for 𝕃 in 𝕃_1, ... 𝕃_n do
    for x in X do
        for i in permutation(len(x)) do
            Candidates = GetCandidates(ℳ, x, y, token_id = i)
            if Candidates and 𝒰(0, 1) > 0.5 then
                x, _ = random.choice(Candidates)
            end if
        end for
        X'.append(x)
    end for
end for
return X'
```

3.3.1 Генерация адверсариальной выборки

Для генерации выборки используется адаптация алгоритма phrase-level адверсариальной атаки (Алгоритм (3.4)). Разница заключается в том, что токены заменяются на их эквиваленты с некоторой вероятностью. Таким образом, для генерации выборки не требуется обученная модель.

Выборка является конкатенацией сгенерированных выборок для всех языков представленных в датасете кроме английского . Каждая из подвыборок генерируется встраиванием целевого языка в обучающую выборку датасета MultiAtis++ на английском языке. Псевдокод функции GetCandidates представлен в секции про атаки (Алгоритм (3.3)).

После генерации у нас получается 6 подвыборок по 4884 предложения в каждой. Итоговая выборка состоит из 29304 предложений, мы делим эту выборку в отношении 9 к 1 на обучающую и тестовую. Примеры объектов из адверсариальной выборки - (3.8), (3.9) и (3.10).

Utterance en	what	flights	from	las	vegas	to	phoenix	on saturday
Utterance adv	Quels	vols	de	las	Vegas	à	phoenix	on saturday

Таблица 3.8: Пример 1 из адверсариальной выборки.

Utterance en	show	me	the	flights	from	denver	to	westchester county
Utterance adv	Mostre	me	os	flights	de	denver	to	Westchester County

Таблица 3.9: Пример 2 из адверсариальной выборки.

Utterance en	show	me	all	flights	from	san	francisco	to atlanta
Utterance adv	show	me	all	flights	de	São	Francisco	to Atlanta

Таблица 3.10: Пример 3 из адверсариальной выборки.

3.3.2 Дообучение тела модели

После генерации адверсариальной выборки мы дообучаем предобученную мультиязычную модель на этой выборке. Модель обучается в режиме задачи

маскированного моделирования языка.

Для обучения модели для такой задачи мы отбираем 15% токенов и предсказываем их с помощью модели. 80% отобранных токенов заменяются на токен маски, 10% заменяются на случайные слова из словаря, остальные 10% остаются неизменными [2]. Мы дообучаем обе мультиязычные модели m-BERT и XLM-RoBERTa с одинаковыми гиперпараметрами - 10 эпох с размером батча 64 и длиной шага 10^{-5}. После дообучения мы сохраняем тело модели для дальнейшего использования.

3.3.3 Загрузка дообученного тела модели

Перед обучением мультиязычной модели для задачи одновременного заполнения слотов и классификации интентов мы загружаем дообученное тело модели.

Для моделей, которые были предобучены с помощью метода адверсариального предобучения, мы будем добавлять в название суффикс «adv» (3.1.3).

3.4 Результаты

3.4.1 Решение задачи классификации интентов и заполнения слотов

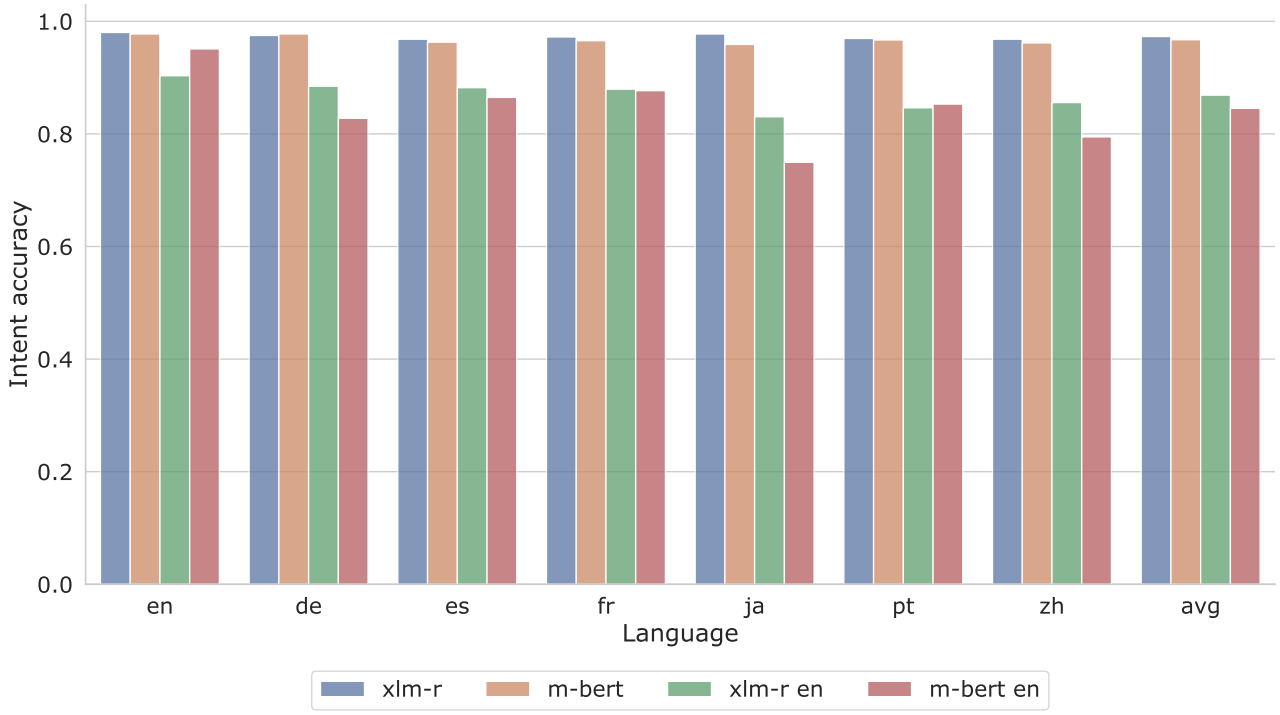

Рис. 3.1: Сравнение моделей между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Intent accuracy**.

Мы решили задачу классификации интентов и заполнения слотов. На графике (3.1) можно увидеть сравнительную диаграмму для метрики Intent accuracy. В данной секции с результатами мы будем приводить графики результатов только для этой метрики, графики двух других метрик и подробные таблицы с результатами можно найти в приложениях (графики в (В), таблицы в (Г)).

В своей работе мы обнаружили, что модель XLM-RoBERTa, обучавшаяся на всей тренировочной выборке имеет слегка лучшее качество, чем модель m-BERT. Это объяснимо за счёт того, что XLM-RoBERTa обучалась на корпусе CommonCrawl, который на несколько порядков больше, чем аналогичный датасет для m-BERT. Однако в целом обе модели показали примерно одинаковое качество и говорить о каких-то значительных различиях в данном контексте не приходится.

Так же мы обнаружили, что модели, обучавшиеся на английской тренировочной подвыборке имеют качество ощутимо хуже, чем их аналоги с полной выборки. В дополнение к этому мы заметили, что модель XLM-RoBERTa показывает качество значительно хуже, чем m-BERT после обучения только на английском. Это можно объяснить за счёт того, что данная языковая модель больше и требует большего количества данных для обучения. Данное сравнение показывает, что перенос знаний ещё не в полной мере может соперничать с обучением модели на переведенных данных.

3.4.2 Качество моделей после адверсариальных атак

Мы провели две адверсариальные атаки на каждую из моделей и замерили качество. Примеры атак на все модели можно найти в приложении (Б).

На графике (3.2) можно увидеть сравнительную диаграмму после word-level атаки для метрики Intent accuracy. На графике (3.3) можно увидеть сравнительную диаграмму после phrase-level атаки для метрики Intent accuracy.

В своей работе мы обнаружили, что как и предполагалось качество после word-level атаки хуже, чем после phrase-level атаки. Так же мы выяснили, что в целом модели XLM-RoBERTa и m-BERT достаточно хорошо справились с атаками и показали высокое качество.

В своей работе мы убедились, что модель XLM-RoBERTa оказалось более робастной в данной задаче после адверсариальных атак, чем модель m-BERT. Так же мы заметили, что модели, обученные на английской подвыборке гораздо хуже справлялись с атаками, нежели их аналоги, обученные на полной выборке. Наиболее сильное влияние на качество оказывала атака с применением португальского языка, а так же китайского и японского. Это объясняется низкоресурсностью португальского языка и иной морфологической структурой азиатских языков.

Дополнительно мы заметили, что m-BERT обученный только на английской подвыборке гораздо лучше справляется с phrase-level атакой, однако хуже справляется с word-level атакой, чем XLM-RoBERTa. Это объяснимо

тем, что m-BERT обучается на Wikipedia, которая может в какой-то мере предоставить параллельные статьи на одну и ту же тему, таким образом у данной модели в обучающих данных присутствует некоторое выравнивание, что позволило расположить языки соответствующим образом в признаковом пространстве. Таким образом, m-BERT оказался более чувствителен к синтаксическим пертурбациям. XLM-RoBERTa же обучалась на моноязычных данных, но гораздо большего объема, что привело к лучшему качеству после word-level атаки.

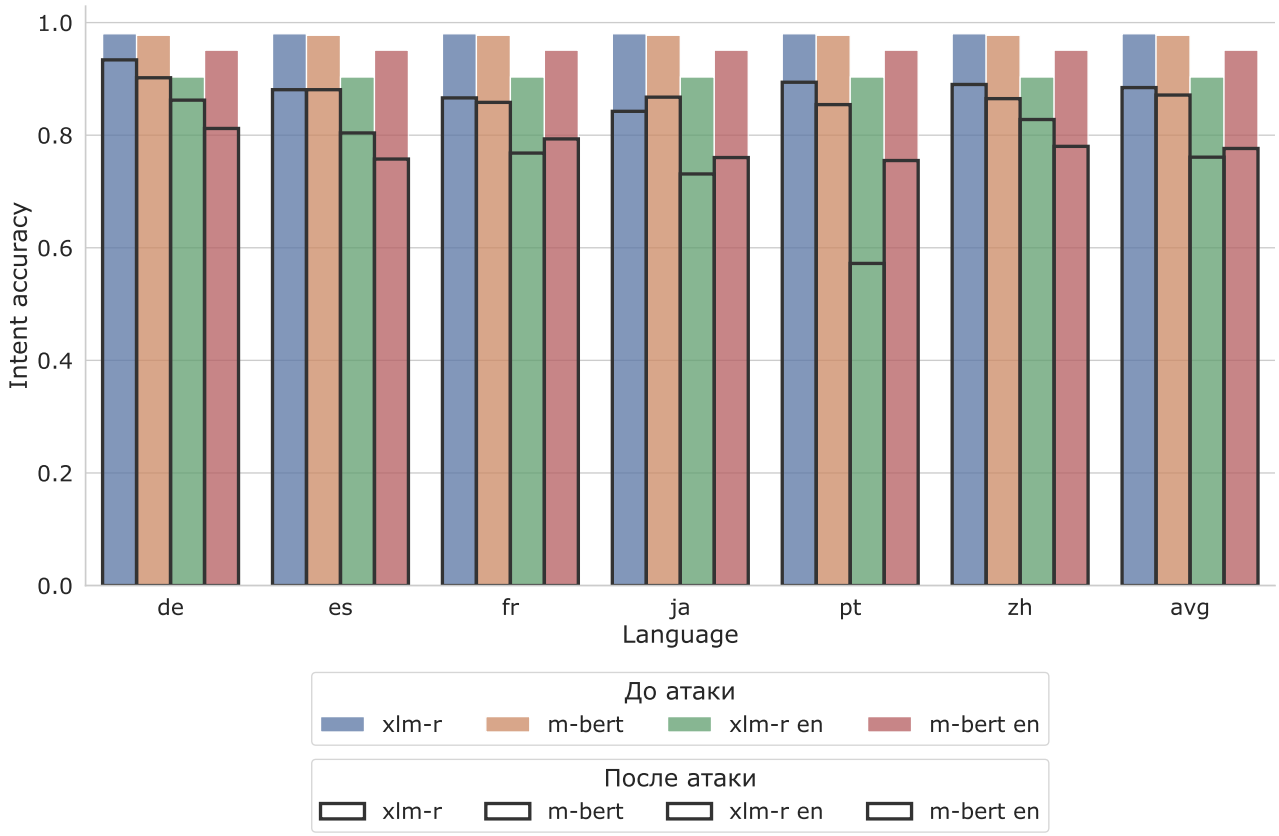

Рис. 3.2: Сравнение моделей между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**.

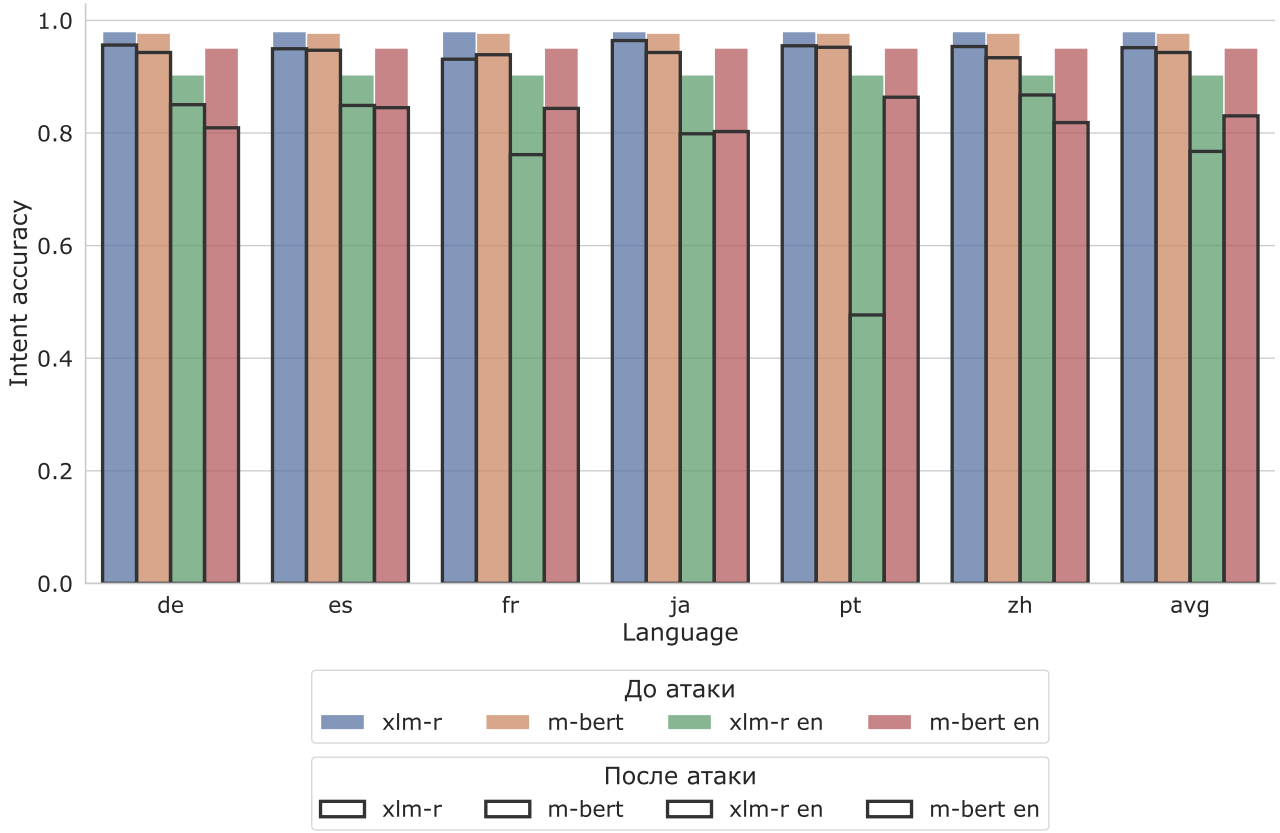

Рис. 3.3: Сравнение моделей между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**.

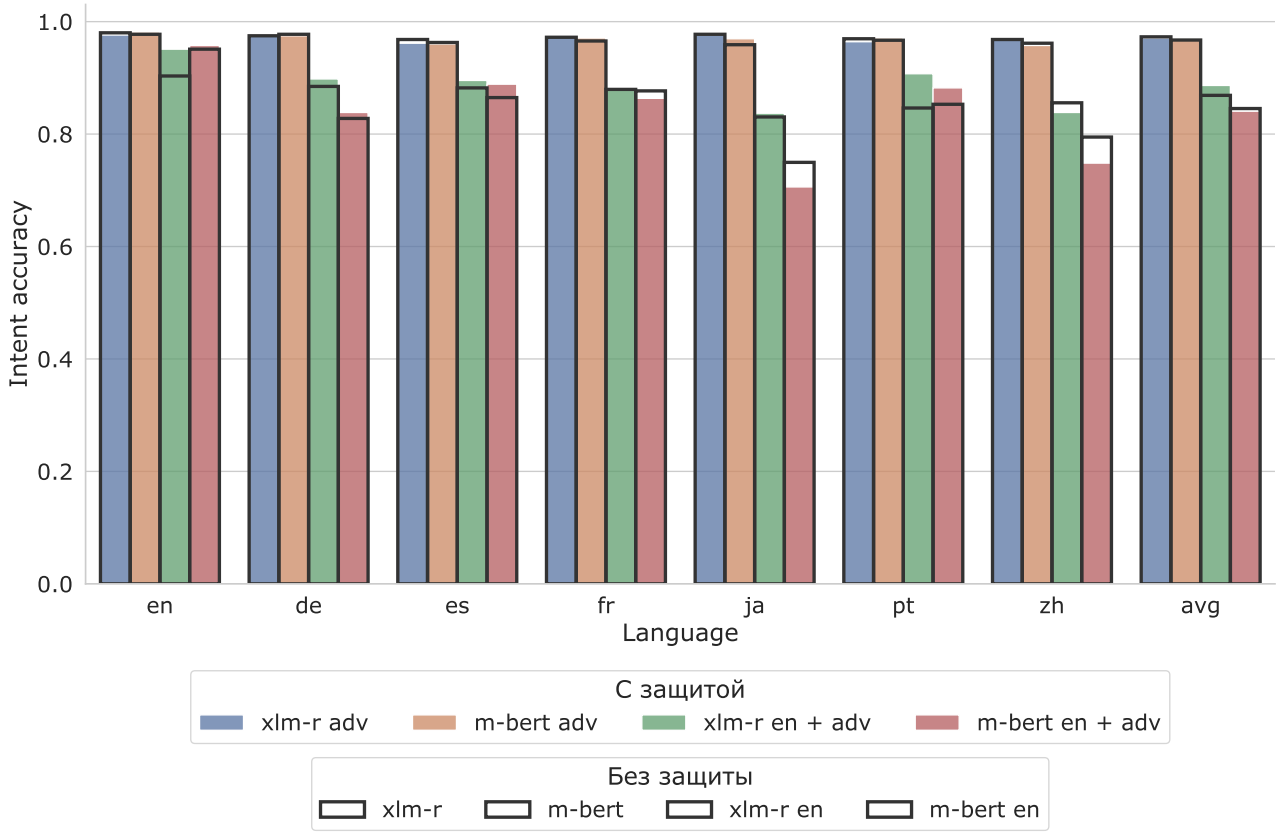

Рис. 3.4: Сравнение моделей **с защитой** между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Intent accuracy**.

3.4.3 Влияние метода адверсариального предобучения

Мы дообучили обе модели XLM-RoBERTa и m-BERT на нашей адверсариальной выборке в режиме маскированного моделирования языка. На графике (3.4) можно видеть сравнительную диаграмму влияния метода защиты на качество моделей на тестовой выборке датасета MultiAtis++.

В своей работе мы обнаружили, что дополнительная защита не повлияла на качество на тестовой выборке у моделей, обучаемых на полной тренировочной выборке. В то же время модели обучаемые на английской подвыборке XLM-RoBERTa немного увеличила качество, а m-BERT уменьшил. Это можно объяснить тем, что XLM-RoBERTa мы дали немного информации про отношения между языками, что позволило немного сместить взаиморасположение языков в признаковом пространстве. В то же время эта же информация могла негативно повлиять на m-BERT с точки зрения выравниваний между языками.

На графике (3.5) можно видеть сравнительную диаграмму влияния метода защиты на качество моделей после word-level атаки для метрики Intent accuracy. На графике (3.6) можно видеть сравнительную диаграмму влияния метода защиты на качество моделей после phrase-level атаки для метрики Intent accuracy.

Во время исследования мы выяснили, что защита добавила качества после атак модели m-BERT, всем её вариациям. Особенно подросло качество после word-level атаки, это можно объяснить тем, что мы внесли некоторый шум в распределение языков в пространстве признаков m-BERT и таким образом сделали его менее чувствительным к синтаксическим пертурбациям.

Так же мы выяснили, что защита негативно сказалась на качестве на азиатских языках после атаки, в большинстве своём качество упало для моделей с использованием защиты. Это можно объяснить тем, что мы внесли некорректную информацию о взаиморасположении слов между собой в азиатских языках, что ухудшило качество и сделало модели более чувствительными к

атакам.

В своей работе мы заметили, что защита влияет наиболее сильно на модели, обученные на английской подвыборке, нежели на всей тренировочной выборке датасета. Это можно объяснить тем, что модели, обучаемые на полной выборке, могут получить информацию об относительном расположении языков и в принципе обучиться для какого-то языка, в то время как другие модели такого сделать не могут.

В целом защита больше повлияла на метрику Semantic accuracy, остальные две метрики изменяются в значительно меньшем соотношении.

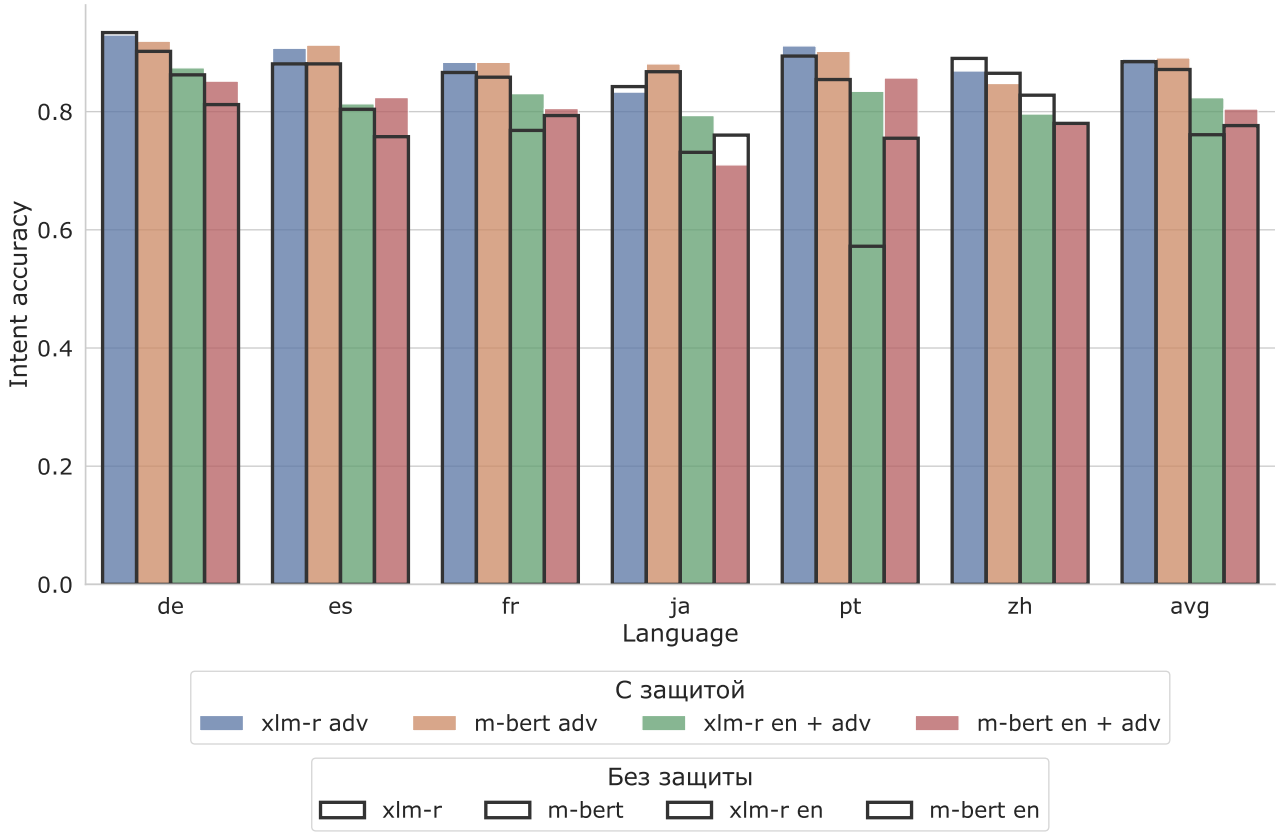

Рис. 3.5: Сравнение моделей **с защитой** между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**.

Так же хотелось бы сравнить между собой предложенный метод адверсариального предобучения и простое обучение на всей тренировочной выборке. Можно рассматривать обучение на всех языках как своего рода защиту, которая повышает качество и позволяет моделям быть более устойчивыми к атакам и показывать в целом лучшее качество. В своей работе мы получили результаты, которые позволяют сделать вывод, что если у нас есть обуча-

ющие данные для задачи заполнения слотов и классификации интентов на одном языке, то будет лучше перевести эти данные на набор возможных языков и разметить, а затем обучить модель на всех таких данных.

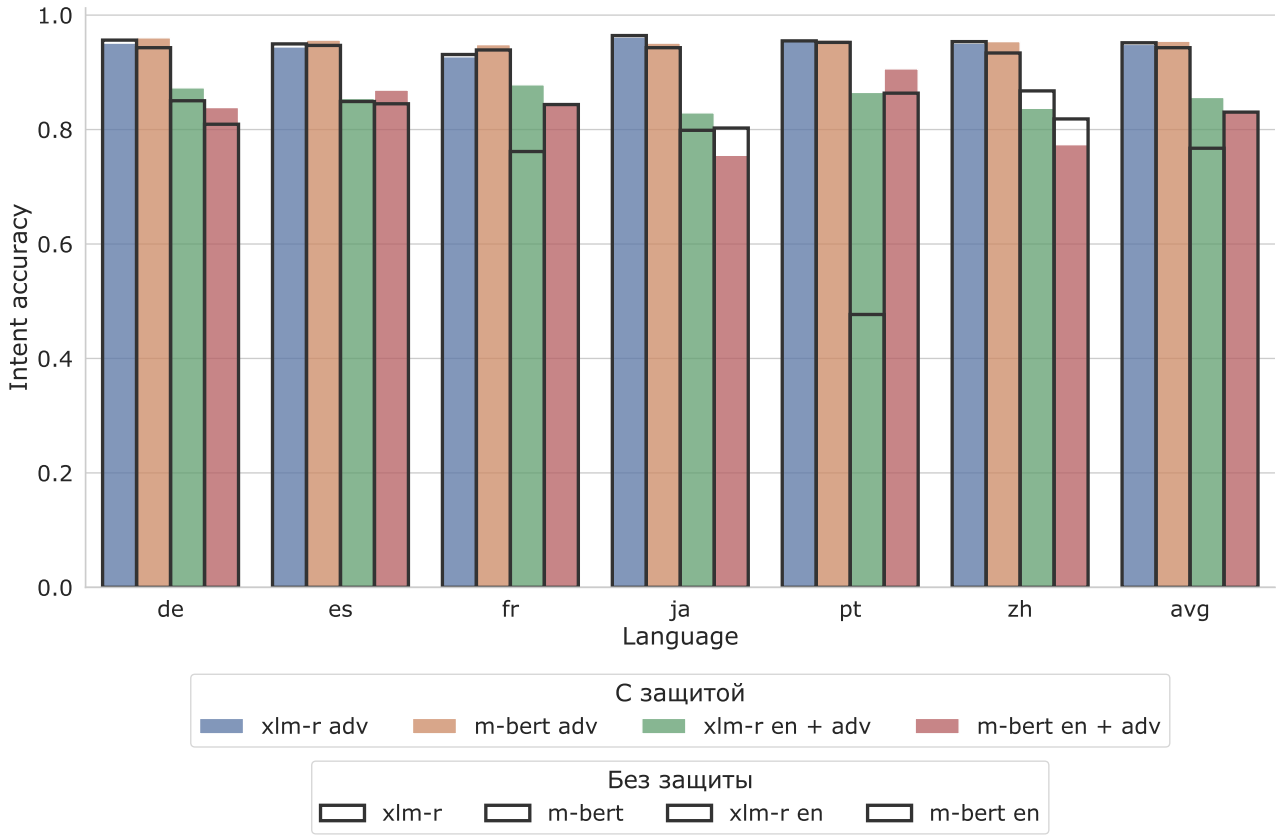

Рис. 3.6: Сравнение моделей **с защитой** между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**.

4 Заключение

В своей работе мы выполнили все поставленные перед началом работы цели, а именно:

- Обучили мультиязычные модели для задачи заполнения слотов и классификации интентов
- Провели две адверсариальные атаки на обученные модели и замерили качество
- Обучили модели по предложенному методу адверсариального предобучения и атаковали полученные модели
- Проанализировали полученные результаты

Мы выяснили, что мультиязычные языковые модели m-BERT и XLM-RoBERTa могут хорошо справляться со смешением кодов в задаче заполнения слотов и классификации интентов. Так же мы выяснили, что модели обученные на тренировочной выборке из семи языков более робастные, чем обученные на выборке на английском языке.

В качестве дальнейшей работы можно рассматривать следующие направления:

- Рассмотреть в экспериментах большее количество мультиязычных моделей
- Рассмотреть в экспериментах альтернативные варианты защиты от атак
- Распространить опыт экспериментов с адверсариальными атаками на другие области обработки естественного языка

Список литературы

[1] Alexis Conneau и др. «Unsupervised Cross-lingual Representation Learning at Scale». В: *ACL*. 2020.

[2] Jacob Devlin и др. «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding». В: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, с. 4171—4186.

[3] Zi-Yi Dou и Graham Neubig. «Word Alignment by Fine-tuning Embeddings on Parallel Corpora». В: *EACL*. 2021.

[4] Angela Fan и др. «Beyond English-Centric Multilingual Machine Translation». В: *ArXiv* abs/2010.11125 (2020).

[5] Haoyang Huang и др. «Unicoder: A Universal Language Encoder by Pre-training with Multiple Cross-lingual Tasks». В: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, нояб. 2019, с. 2485—2494. DOI: 10.18653/v1/D19-1252. URL: https://www.aclweb.org/anthology/D19-1252.

[6] Alexandre Klementiev, Ivan Titov и Binod Bhattarai. «Inducing Crosslingual Distributed Representations of Words». В: *Proceedings of COLING 2012*. Mumbai, India: The COLING 2012 Organizing Committee, дек. 2012, с. 1459—1474. URL: https://www.aclweb.org/anthology/C12-1089.

[7] Jitin Krishnan и др. «Multilingual Code-Switching for Zero-Shot Cross-Lingual Intent Prediction and Slot Filling». В: *ArXiv* abs/2103.07792 (2021).

[8] Chi-Liang Liu и др. «What makes multilingual BERT multilingual?» В: *ArXiv* abs/2010.10938 (2020).

[9] Shana Poplack, DAVID SANKOFF и CHRISTOPHER MILLER. «The social correlates and linguistic processes of lexical borrowing and assimilation». В: *Linguistics* 26 (1988), с. 47—104.

[10] Samson Tan и Shafiq Joty. «Code-Mixing on Sesame Street: Dawn of the Adversarial Polyglots». В: *ArXiv* abs/2103.09593 (2021).

[11] Ashish Vaswani и др. «Attention is All you Need». В: *ArXiv* abs/1706.03762 (2017).

[12] H. Weld и др. «A survey of joint intent detection and slot-filling models in natural language understanding». В: *ArXiv* abs/2101.08091 (2021).

[13] Shijie Wu и Mark Dredze. «Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT». В: *EMNLP/IJCNLP*. 2019.

[14] Weijia Xu, Batool Haider и Saab Mansour. «End-to-End Slot Alignment and Recognition for Cross-Lingual NLU». В: *ArXiv* abs/2004.14353 (2020).

Приложения

A Алгоритм замены слотов в атаке

Algorithm A.1 Алгоритм замены слотов в атаке

```
function ExtendSlotLabels(slot_label, num_tokens)
    slot_labels = [slot_label]
    if num_tokens > 1 then
        if slot_label.startswith('B') then
            slot_labels += ['I' + slot_label[1:]] · (num_tokens - 1)
        else
            slot_labels ·= num_tokens
        end if
    end if
    return slot_labels
end function
```

Б Примеры адверсариальных атак на модели

Utterance en	show	me	the	cheapest	one	way	flights	from	montreal	to	orlando	
Utterance adv	spectacle	me	the	Le	moins	cher	one	Le	chemin	vols	de	Montréal à Orlando

Таблица Б.1: Пример атаки модели m-BERT (m-bert) word-level атакой.

Utterance en	show	me	flights	from	fort	worth	to	san jose
Utterance adv	Zeige	me	Flüge	von	fort	worth	nach	San Jose

Таблица Б.2: Пример атаки модели m-BERT (m-bert) phrase-level атакой.

Utterance en	find	flight	from	memphis	to	cincinnati	on sunday
Utterance adv	Encontrar	flight	from	Memphis	to	cincinnati	En sunday

Таблица Б.3: Пример атаки модели m-BERT (m-bert en) word-level атакой.

Utterance en	please	list	flights	from	philadelphia	to	san francisco
Utterance adv	bitte	bitte	flights	from	Philadelphia	to	San Francisco

Таблица Б.4: Пример атаки модели m-BERT (m-bert en) phrase-level атакой.

Utterance en	list	a	flight	on	american	airlines	from	toronto	to	san diego
Utterance adv	Lista	a	flight	on	estadounidense	Aerolíneas	de	Torrente	to	San Diego

Таблица Б.5: Пример атаки модели m-BERT (m-bert adv) word-level атакой.

Utterance en	show	me	all	the	flights	from	burbank	to milwaukee
Utterance adv	show	me	todos	the	voos	from	Burbank	to Milwaukee

Таблица Б.6: Пример атаки модели m-BERT (m-bert adv) phrase-level атакой.

Utterance en	what	flights	travel	from	las	vegas	to	los	angeles
Utterance adv	Ce	que	vols	travel	de	Les	VEGAS	to	los Les Anges

Таблица Б.7: Пример атаки модели m-BERT (m-bert en + adv) word-level атакой.

Utterance en	list	american	airlines	flights	from	houston	to	milwaukee	departing	friday	pm		
Utterance adv	list	American	Airlines	-Flüge	flights	from	Houston	nach	Milwaukee	abfliegen	die	Freitag	nachmittag	pm

Таблица Б.8: Пример атаки модели m-BERT (m-bert en + adv) phrase-level атакой.

Utterance en	please	list	the	flights	from	newark	to	los	angeles
Utterance adv	please	Liste	Le	flights	de	newark	to	los	Les	Anges

Таблица Б.9: Пример атаки модели XLM-RoBERTa (xlm-r) word-level атакой.

Utterance en	show	me	the	nonstop	flights	from	toronto	to	st.	petersburg
Utterance adv	show	me	the	nonstop	vuelos	de	toronto	a	san	petersburg

Таблица Б.10: Пример атаки модели XLM-RoBERTa (xlm-r) phrase-level атакой.

Utterance en	list	flights	from	seattle	to	salt	lake	city	on	delta	be1
Utterance adv	list	flights	from	Seattle	to	salt	El	lago	city	En	El	delta	B1

Таблица Б.11: Пример атаки модели XLM-RoBERTa (xlm-r en) word-level атакой.

Utterance en	which	different	airlines	go	from	las	vegas	to	new	york	city
Utterance adv	which	different	aériennes	go	from	Las	vegas	à	New	York	City

Таблица Б.12: Пример атаки модели XLM-RoBERTa (xlm-r en) phrase-level атакой.

Utterance en	list	the	flights	from	westchester	county	to	denver	on	june	seventh
Utterance adv	Lista	the	vuelos	from	El	Westchester	Condado	para	El	Denver	on	Junio	seventh

Таблица Б.13: Пример атаки модели XLM-RoBERTa (xlm-r adv) word-level атакой.

Utterance en	which	flights	on	us	air	go	from	orlando	to	cleveland
Utterance adv	Quels	vols	d’	US	Air	go	d’	orlando	to	cleveland

Таблица Б.14: Пример атаки модели XLM-RoBERTa (xlm-r adv) phrase-level атакой.

Utterance en	show	me	the	flights	between	houston	and	orlando	
Utterance adv	spectacle	me	Le	vols	between	à	Houston	et	Orlando

Таблица Б.15: Пример атаки модели XLM-RoBERTa (xlm-r en + adv) word-level атакой.

Utterance en	show	me	the	flights	between	houston	and	orlando
Utterance adv	muéstrame	muéstrame	los	vuelos	between	houston	y	orlando

Таблица Б.16: Пример атаки модели XLM-RoBERTa (xlm-r en + adv) phrase-level атакой.

B Графики с результатами экспериментов

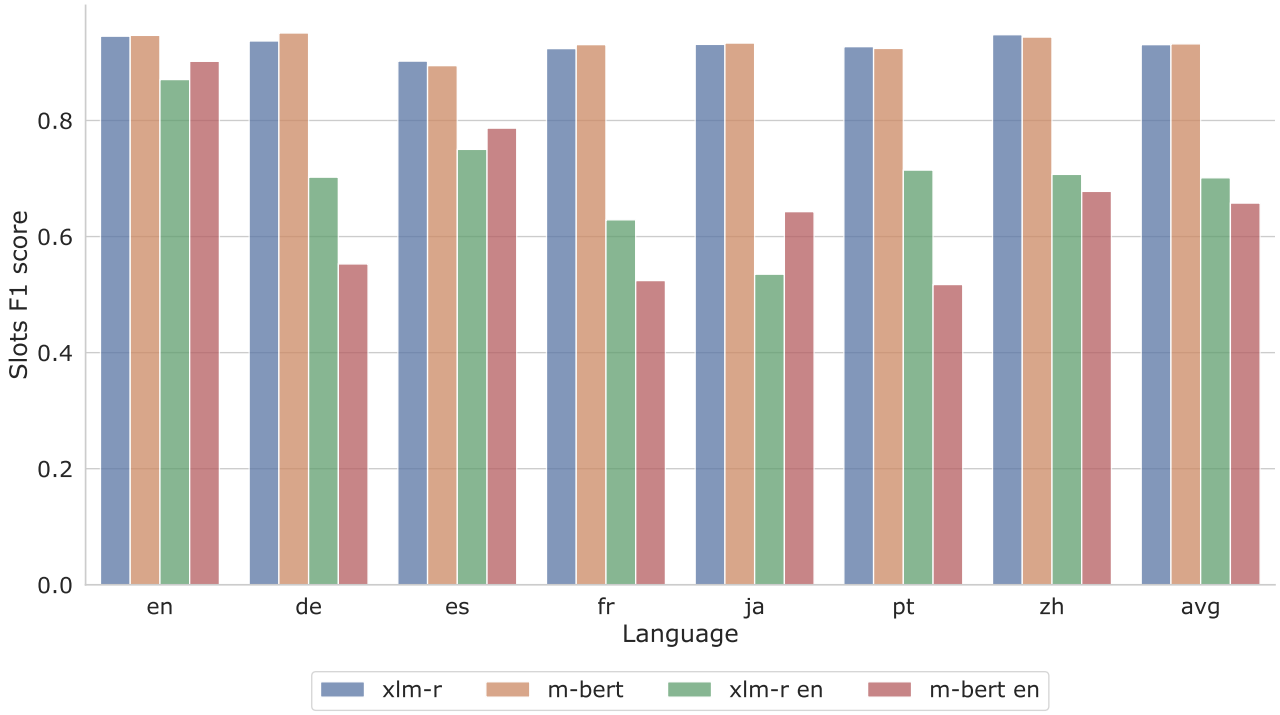

Рис. B.1: Сравнение моделей между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Slots F1 score**.

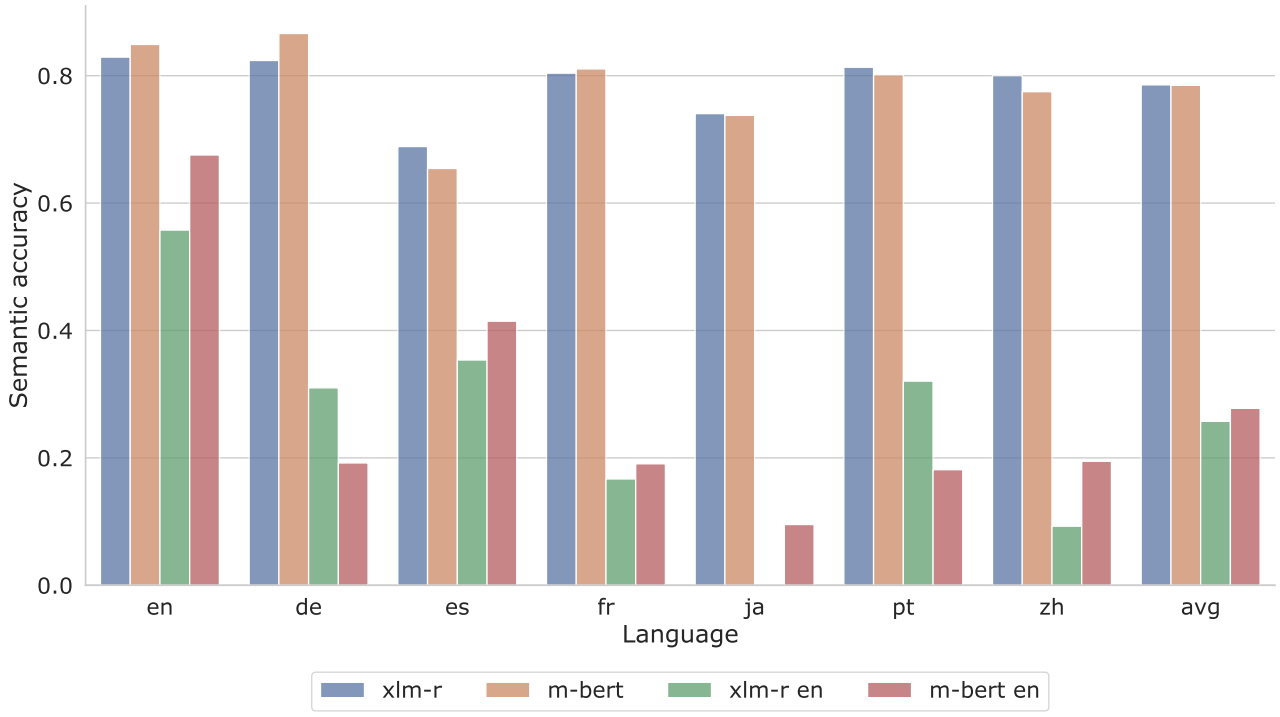

Рис. B.2: Сравнение моделей между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Semantic accuracy**.

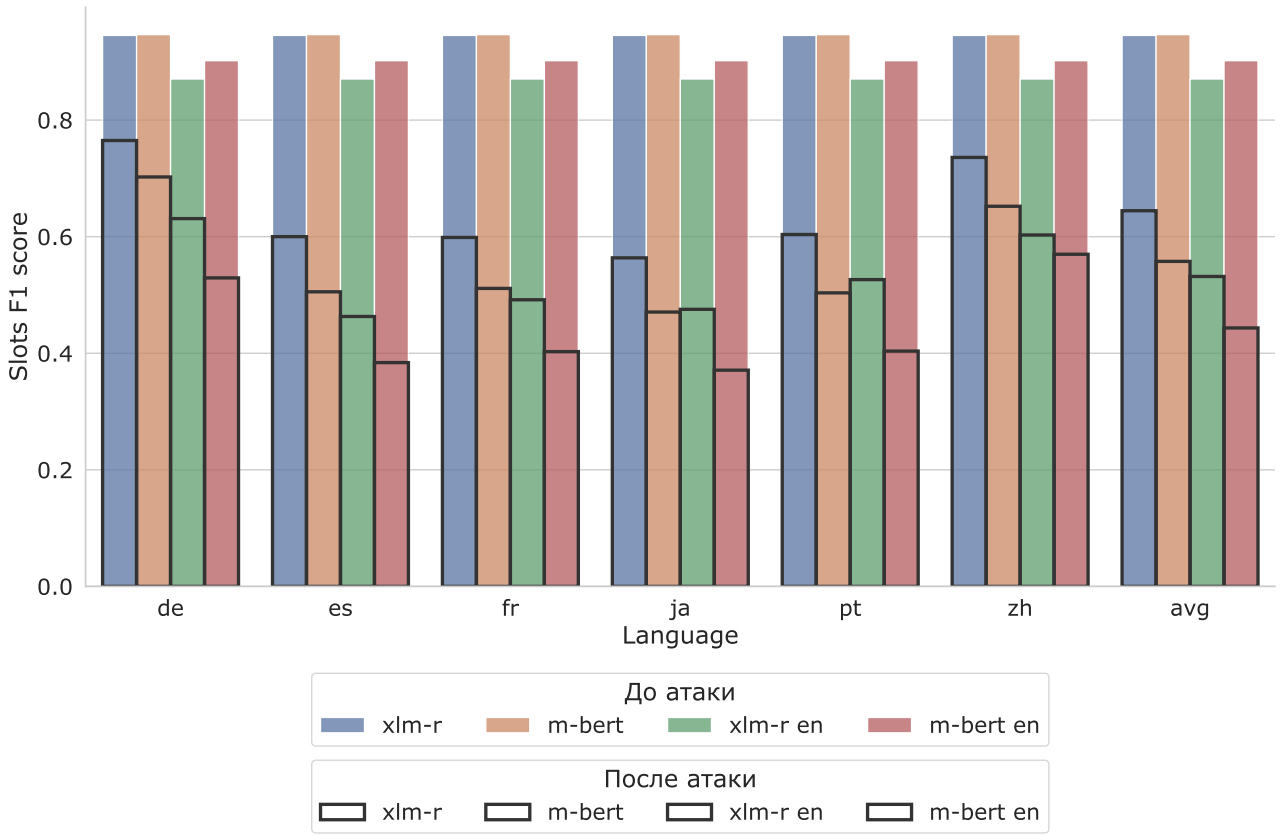

Рис. В.3: Сравнение моделей между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**.

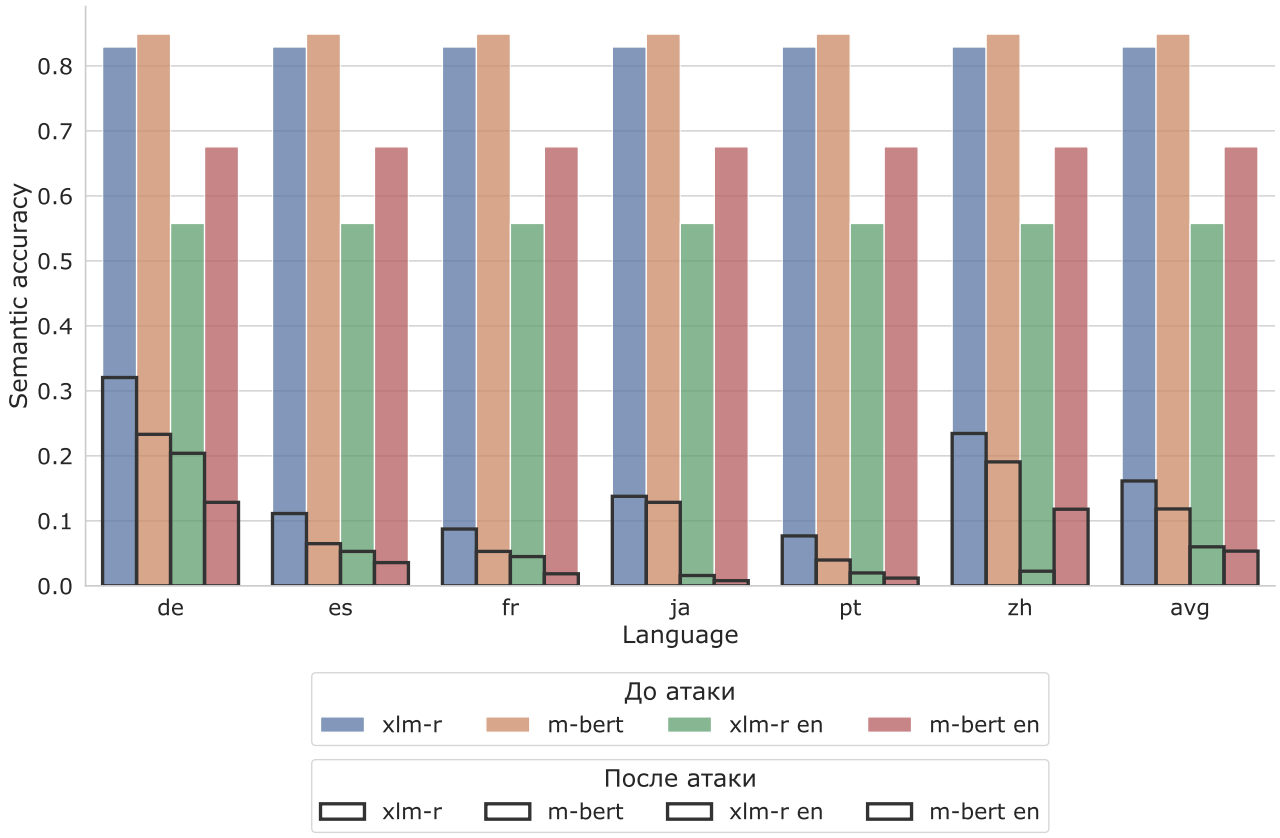

Рис. В.4: Сравнение моделей между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**.

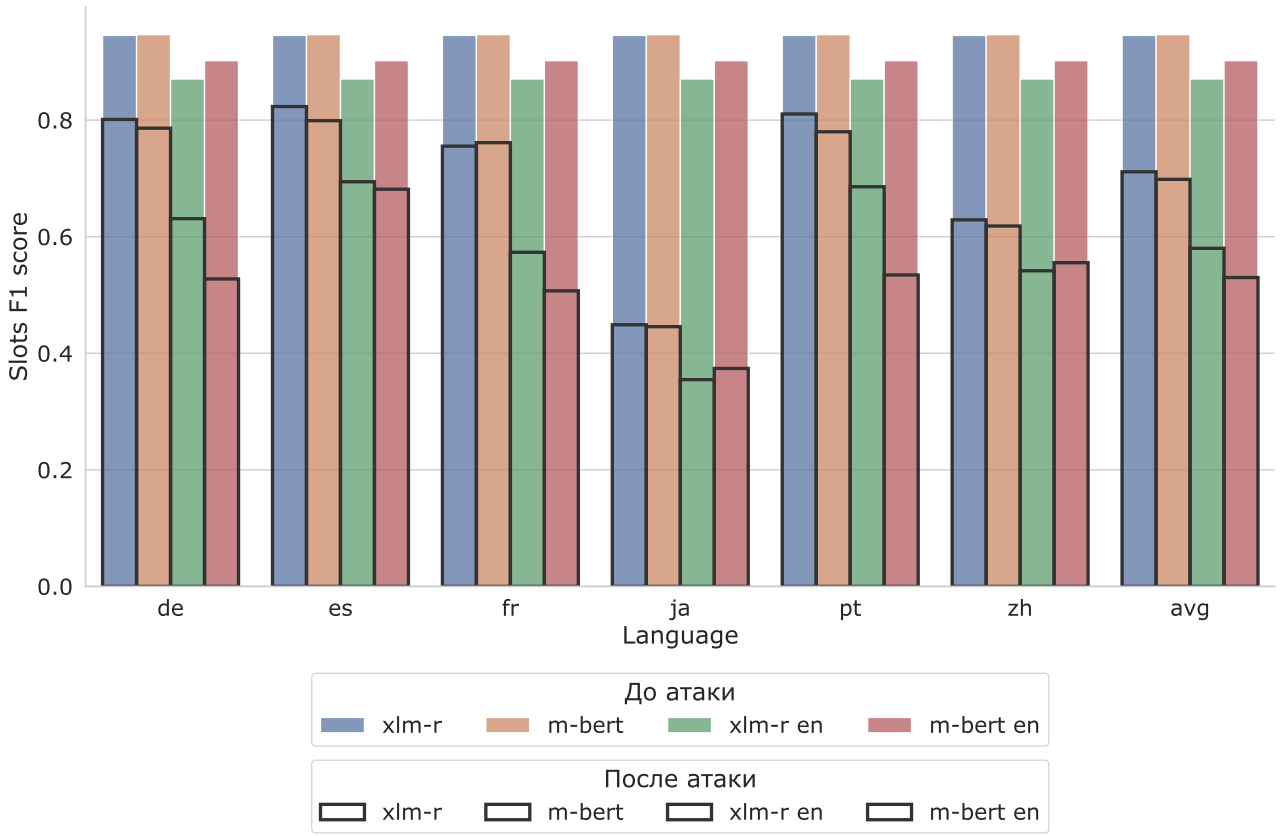

Рис. B.5: Сравнение моделей между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**.

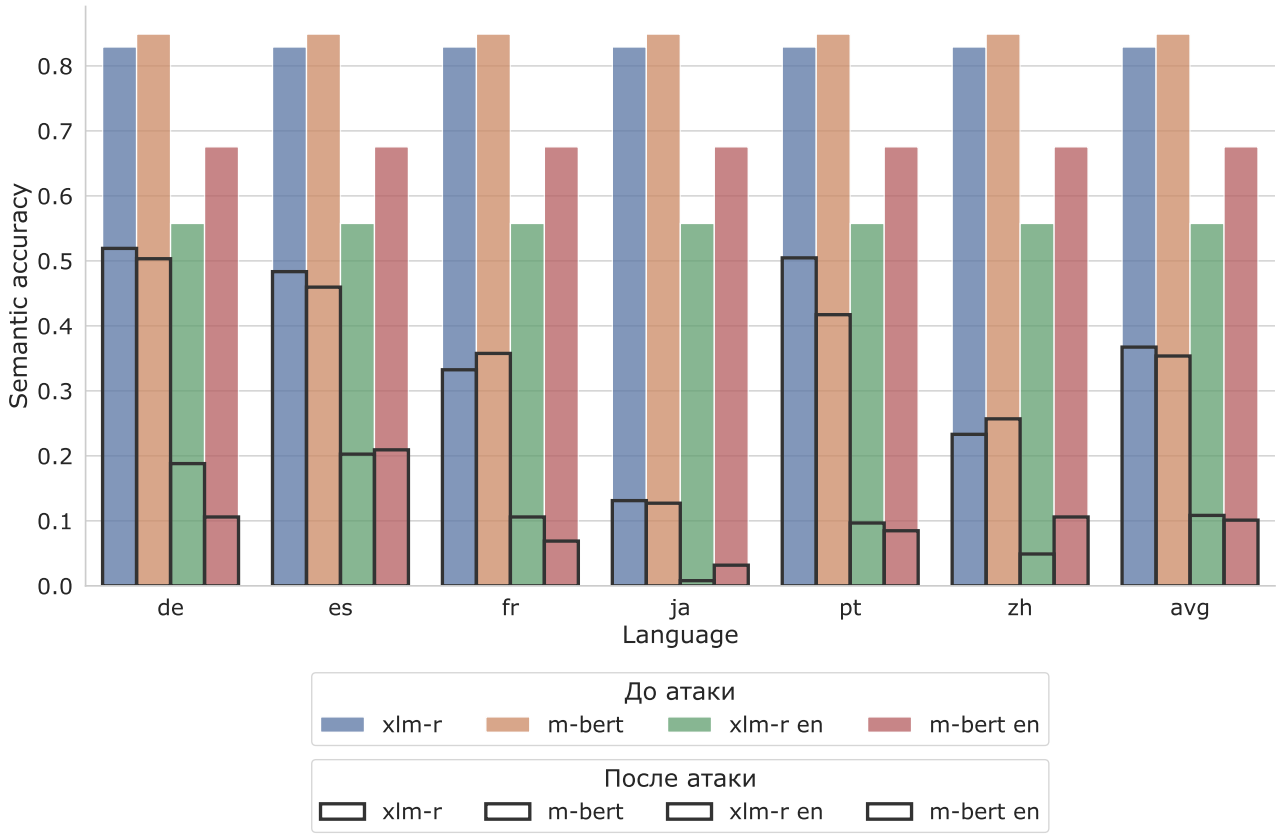

Рис. B.6: Сравнение моделей между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**.

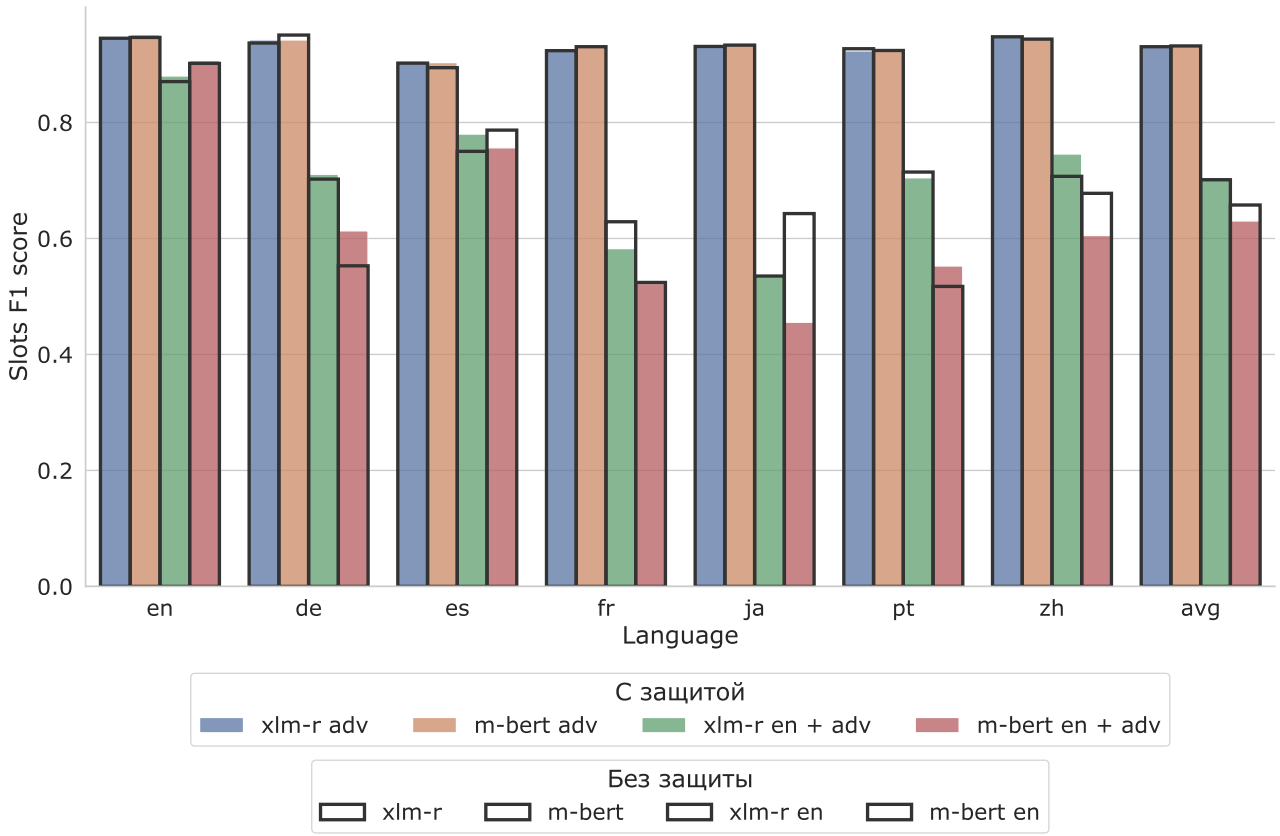

Рис. В.7: Сравнение моделей **с защитой** между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Slots F1 score**.

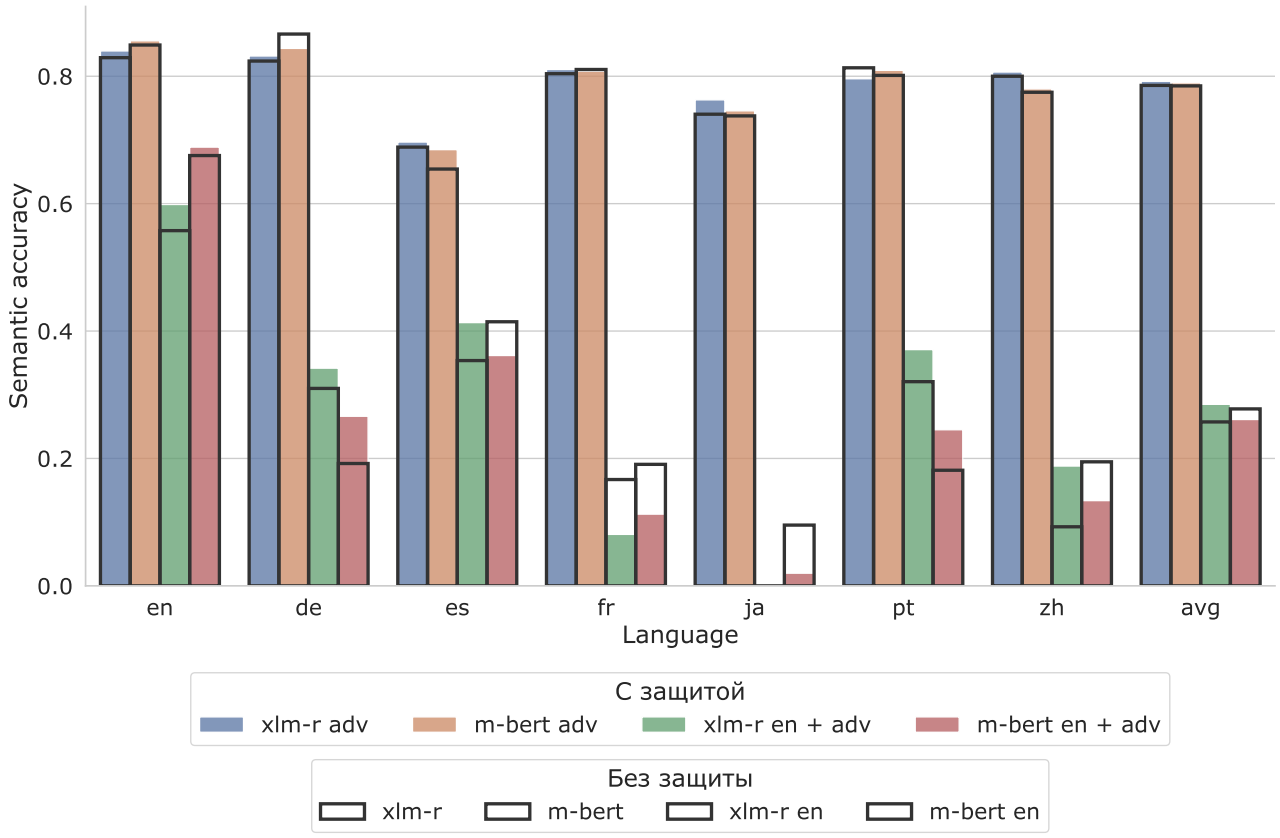

Рис. В.8: Сравнение моделей **с защитой** между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Semantic accuracy**.

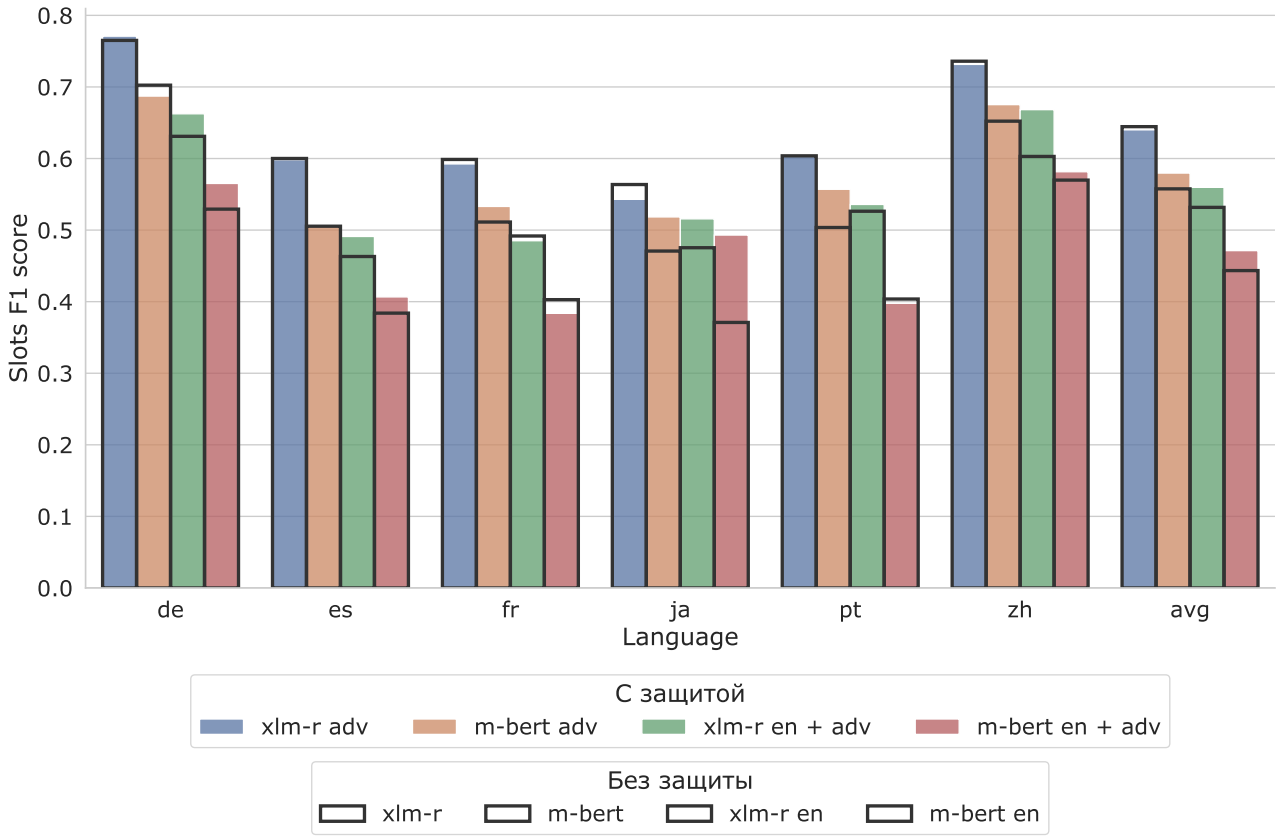

Рис. B.9: Сравнение моделей **с защитой** между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**.

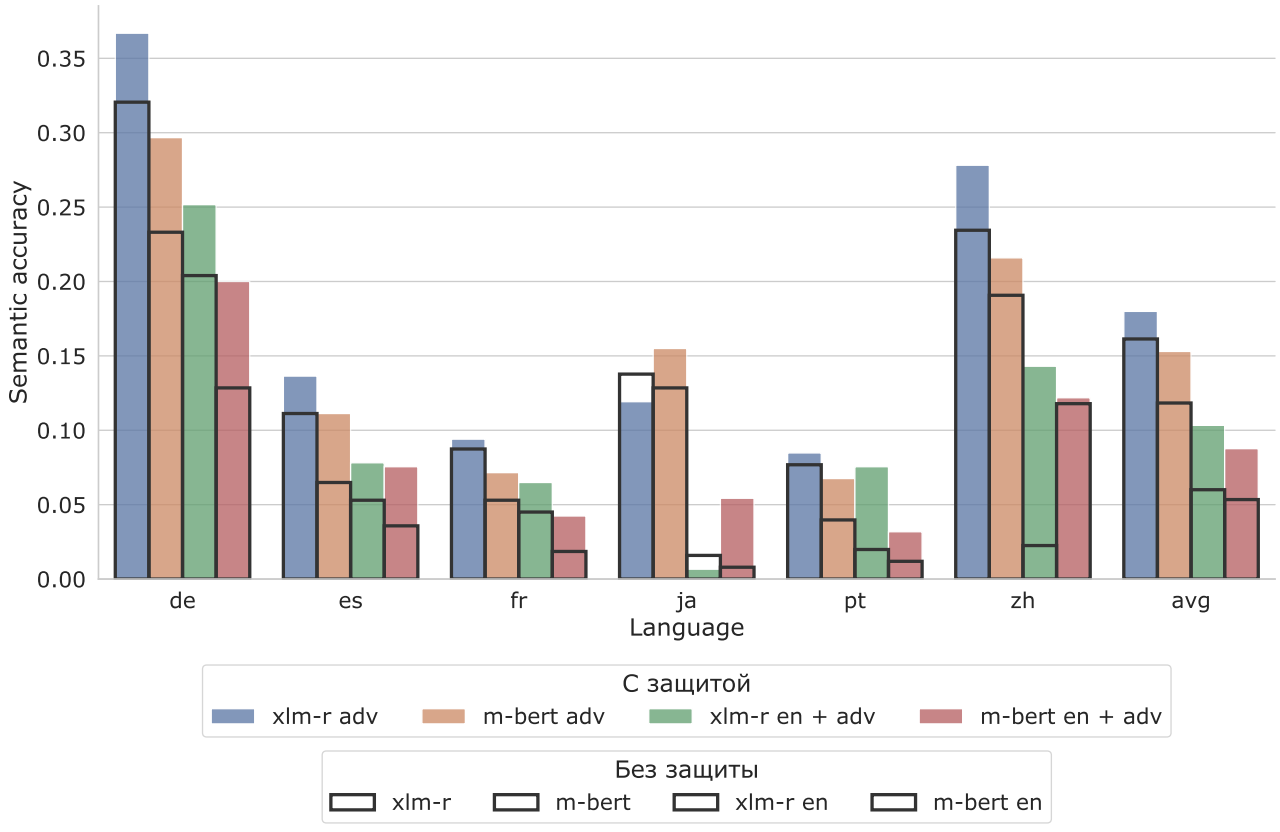

Рис. B.10: Сравнение моделей **с защитой** между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**.

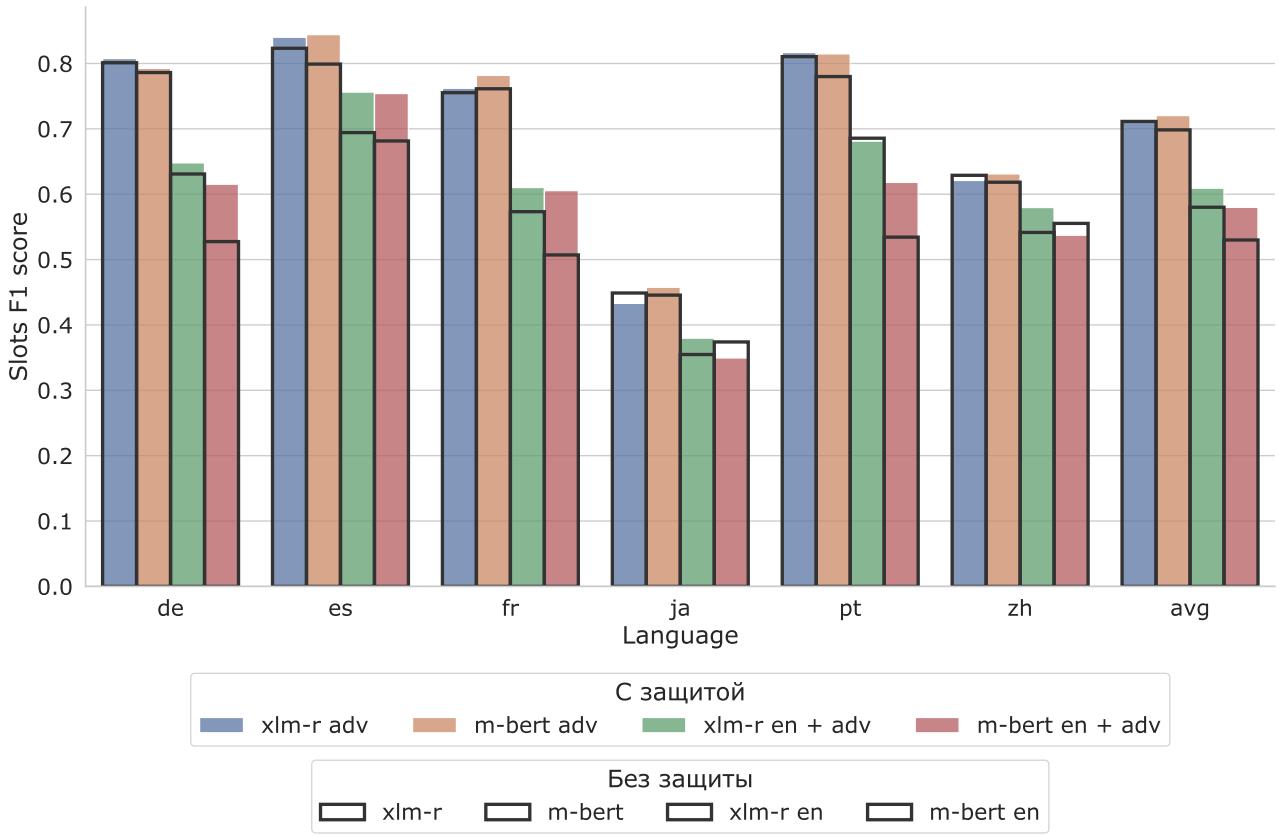

Рис. B.11: Сравнение моделей **с защитой** между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**.

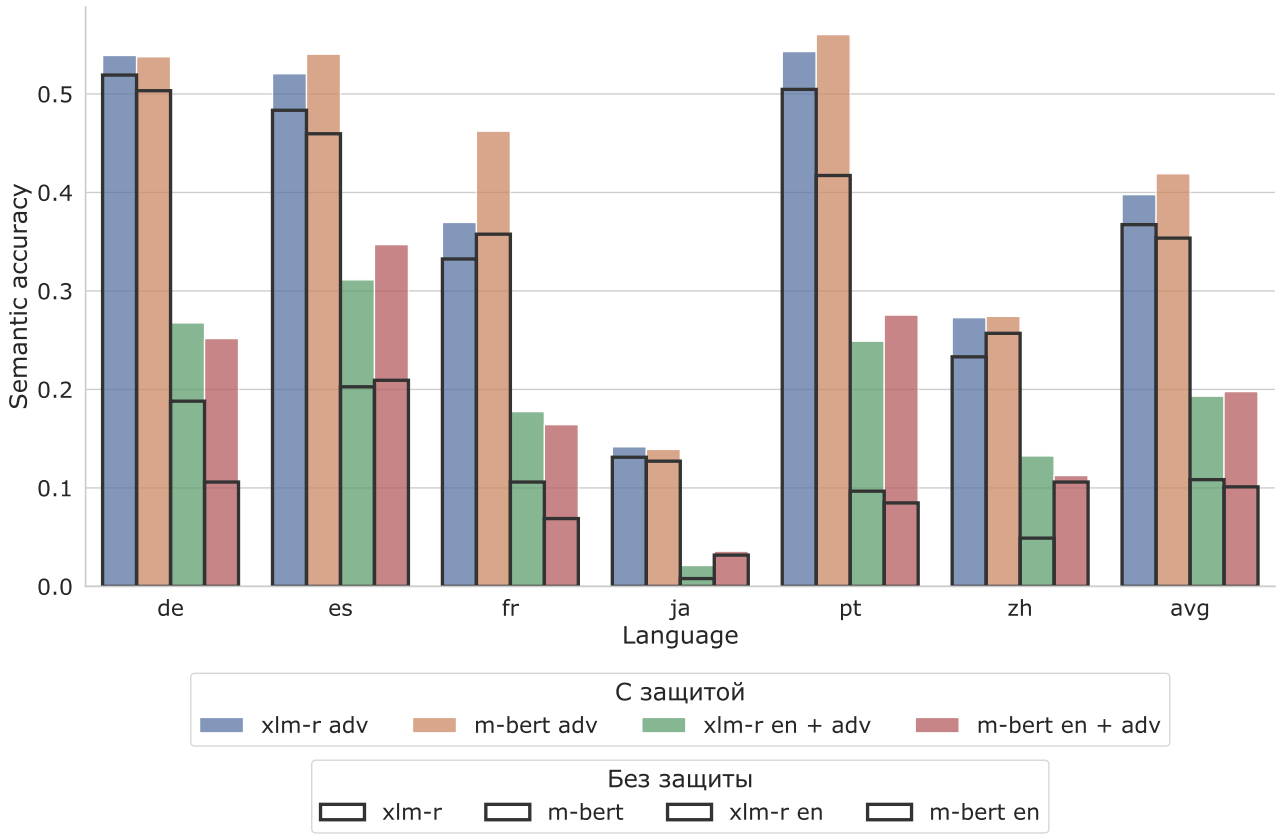

Рис. B.12: Сравнение моделей **с защитой** между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**.

Г Таблицы с результатами экспериментов

	en	de	es	fr	ja	pt	zh	avg
xlm-r	0.980	0.975	0.968	0.972	0.977	0.970	0.968	0.973
m-bert	0.977	0.977	0.963	0.966	0.959	0.967	0.962	0.967
xlm-r en	0.903	0.885	0.882	0.879	0.830	0.846	0.856	0.869
m-bert en	0.951	0.828	0.865	0.877	0.750	0.853	0.795	0.845

Таблица Г.1: Сравнение моделей между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Intent accuracy**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

	en	de	es	fr	ja	pt	zh	avg
xlm-r	0.945	0.937	0.902	0.924	0.931	0.927	0.948	0.931
m-bert	0.947	0.951	0.895	0.931	0.933	0.924	0.944	0.932
xlm-r en	0.871	0.702	0.750	0.629	0.535	0.715	0.707	0.701
m-bert en	0.902	0.553	0.787	0.524	0.643	0.517	0.678	0.658

Таблица Г.2: Сравнение моделей между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Slots F1 score**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

	en	de	es	fr	ja	pt	zh	avg
xlm-r	0.829	0.824	0.689	0.804	0.740	0.813	0.800	0.786
m-bert	0.849	0.866	0.654	0.811	0.738	0.801	0.775	0.785
xlm-r en	0.558	0.310	0.354	0.167	0.000	0.321	0.093	0.257
m-bert en	0.675	0.192	0.415	0.191	0.095	0.181	0.195	0.278

Таблица Г.3: Сравнение моделей между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r	0.934	0.881	0.866	0.842	0.894	0.890	0.885
m-bert	0.902	0.881	0.858	0.868	0.854	0.865	0.871
xlm-r en	0.862	0.804	0.768	0.731	0.572	0.828	0.761
m-bert en	0.812	0.758	0.793	0.760	0.755	0.780	0.776

Таблица Г.4: Сравнение моделей между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r	0.765	0.600	0.599	0.564	0.604	0.736	0.645
m-bert	0.702	0.505	0.511	0.471	0.504	0.652	0.558
xlm-r en	0.631	0.463	0.492	0.475	0.526	0.603	0.532
m-bert en	0.529	0.384	0.403	0.371	0.404	0.570	0.443

Таблица Г.5: Сравнение моделей между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r	0.321	0.111	0.087	0.138	0.077	0.234	0.161
m-bert	0.233	0.065	0.053	0.128	0.040	0.191	0.118
xlm-r en	0.204	0.053	0.045	0.016	0.020	0.023	0.060
m-bert en	0.128	0.036	0.019	0.008	0.012	0.118	0.053

Таблица Г.6: Сравнение моделей между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r	0.956	0.950	0.931	0.964	0.955	0.954	0.952
m-bert	0.943	0.947	0.939	0.943	0.952	0.934	0.943
xlm-r en	0.850	0.849	0.762	0.799	0.477	0.868	0.767
m-bert en	0.809	0.845	0.844	0.803	0.864	0.819	0.830

Таблица Г.7: Сравнение моделей между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r	0.801	0.823	0.755	0.449	0.810	0.629	0.711
m-bert	0.786	0.799	0.761	0.446	0.780	0.618	0.698
xlm-r en	0.631	0.694	0.573	0.355	0.686	0.541	0.580
m-bert en	0.528	0.681	0.507	0.374	0.534	0.555	0.530

Таблица Г.8: Сравнение моделей между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r	0.519	0.483	0.332	0.131	0.505	0.233	0.367
m-bert	0.503	0.460	0.358	0.127	0.417	0.257	0.354
xlm-r en	0.188	0.203	0.106	0.008	0.097	0.049	0.108
m-bert en	0.106	0.209	0.069	0.032	0.085	0.106	0.101

Таблица Г.9: Сравнение моделей между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	en	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.976	0.975	0.962	0.975	0.976	0.964	0.968	0.971
m-bert adv	0.981	0.975	0.960	0.971	0.970	0.971	0.958	0.969
xlm-r en + adv	0.951	0.898	0.895	0.878	0.837	0.907	0.838	0.886
m-bert en + adv	0.958	0.838	0.889	0.864	0.706	0.882	0.748	0.841

Таблица Г.10: Сравнение моделей **с защитой** между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Intent accuracy**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

	en	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.948	0.942	0.906	0.927	0.933	0.924	0.950	0.933
m-bert adv	0.952	0.942	0.903	0.932	0.934	0.925	0.945	0.933
xlm-r en + adv	0.880	0.711	0.780	0.583	0.534	0.705	0.746	0.705
m-bert en + adv	0.907	0.613	0.756	0.522	0.456	0.553	0.605	0.630

Таблица Г.11: Сравнение моделей **с защитой** между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Slots F1 score**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

	en	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.840	0.832	0.697	0.811	0.763	0.796	0.807	0.792
m-bert adv	0.856	0.844	0.685	0.808	0.746	0.809	0.780	0.790
xlm-r en + adv	0.599	0.342	0.413	0.081	0.001	0.371	0.188	0.285
m-bert en + adv	0.689	0.266	0.362	0.113	0.020	0.245	0.134	0.261

Таблица Г.12: Сравнение моделей **с защитой** между собой **на тестовой выборке** датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.930	0.907	0.883	0.833	0.911	0.869	0.889
m-bert adv	0.919	0.913	0.883	0.881	0.902	0.848	0.891
xlm-r en + adv	0.874	0.813	0.830	0.793	0.834	0.796	0.824
m-bert en + adv	0.852	0.824	0.805	0.710	0.857	0.779	0.804

Таблица Г.13: Сравнение моделей **с защитой** между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.771	0.598	0.592	0.543	0.604	0.731	0.640
m-bert adv	0.687	0.507	0.533	0.518	0.557	0.675	0.580
xlm-r en + adv	0.662	0.491	0.485	0.516	0.536	0.668	0.560
m-bert en + adv	0.565	0.407	0.384	0.493	0.398	0.582	0.471

Таблица Г.14: Сравнение моделей **с защитой** между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.367	0.136	0.094	0.119	0.085	0.278	0.180
m-bert adv	0.297	0.111	0.072	0.155	0.068	0.216	0.153
xlm-r en + adv	0.252	0.078	0.065	0.007	0.075	0.143	0.103
m-bert en + adv	0.200	0.075	0.042	0.054	0.032	0.122	0.088

Таблица Г.15: Сравнение моделей **с защитой** между собой после **word-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.951	0.944	0.927	0.962	0.958	0.951	0.949
m-bert adv	0.960	0.956	0.948	0.951	0.956	0.954	0.954
xlm-r en + adv	0.873	0.854	0.878	0.829	0.865	0.837	0.856
m-bert en + adv	0.838	0.869	0.846	0.755	0.906	0.774	0.831

Таблица Г.16: Сравнение моделей **с защитой** между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Intent accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.808	0.840	0.762	0.433	0.817	0.621	0.713
m-bert adv	0.793	0.844	0.782	0.458	0.815	0.631	0.720
xlm-r en + adv	0.648	0.756	0.610	0.380	0.681	0.580	0.609
m-bert en + adv	0.615	0.754	0.606	0.350	0.618	0.537	0.580

Таблица Г.17: Сравнение моделей **с защитой** между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Slots F1 score**. По колонкам встраиваемые языки, по рядам тестируемые модели.

	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.539	0.521	0.370	0.142	0.543	0.273	0.398
m-bert adv	0.538	0.540	0.462	0.139	0.560	0.274	0.419
xlm-r en + adv	0.268	0.311	0.177	0.021	0.249	0.132	0.193
m-bert en + adv	0.252	0.347	0.164	0.036	0.275	0.113	0.198

Таблица Г.18: Сравнение моделей **с защитой** между собой после **phrase-level** атаки на тестовую выборку датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам встраиваемые языки, по рядам тестируемые модели.