

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 мультязычные модели"

Выполнил студент группы 171, 4 курса,
Биршерт Алексей Дмитриевич

Руководитель ВКР:
Доцент,
Департамент больших данных и информационного поиска
Артемова Екатерина Леонидовна

Москва 2021

Содержание

1	Введение	4
2	Обзор литературы	5
2.1	Мультиязычные модели	5
2.2	Классификация интенгов и заполнение слотов	6
2.3	Смешение кодов в адверсариальных атаках на мультиязычные модели	7
2.4	Машинный перевод и выравнивание слов	7
3	Основная часть	8
3.1	Обучение моделей на датасете MultiAtis++	8
3.1.1	Датасет	8
3.1.2	Архитектура модели	9
3.1.3	Обучение	9
3.2	Адверсариальные атаки	10
3.2.1	Общий вид атаки	11
3.2.2	Word level атака	11
3.2.3	Phrase-level атака	12
3.3	Метод адверсариального предобучения для защиты от адверсариальных атак	13
3.3.1	Генерация адверсариальной выборки	13
3.3.2	Дообучение тела модели	14
3.3.3	Загрузка дообученного тела модели	15
3.4	Результаты	15
3.4.1	Решение задачи классификации интенгов и заполнения слотов	15
3.4.2	Качество моделей после адверсариальных атак	16
3.4.3	Влияние метода адверсариального предобучения	18
4	Заключение	22

Список литературы	23
Приложения	25
А. Алгоритм замены слотов в атаке	25

Аннотация

[illegible]

Ключевые слова—Ключевые слова

Some words in abstract. Some words in abstract. Some words in abstract. Some
words in abstract. Some words in abstract. Some words in abstract. Some words in
abstract. Some words in abstract. Some words in abstract. Some words in abstract.
Some words in abstract. Some words in abstract. Some words in abstract. Some
words in abstract. Some words in abstract. Some words in abstract. Some words in
abstract. Some words in abstract. Some words in abstract. Some words in abstract.
Some words in abstract. Some words in abstract. Some words in abstract. Some
words in abstract. Some words in abstract. Some words in abstract. Some words in
abstract. Some words in abstract. Some words in abstract. Some words in abstract.
Some words in abstract. Some words in abstract. Some words in abstract. Some
words in abstract. Some words in abstract. Some words in abstract. Some words in
in abstract.

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

Keywords—Keywords

1 Введение

Последние несколько лет стали прорывными в области мультязычных моделей и их обобщающей способности для других языков [1, 2, 7, 13]. Огромные мультязычные 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 мультязычных 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 мультязычных пользователей.

В своей работе мы:

- Решаем задачу одновременного детектирования намерений пользователя и заполнения слотов для диалоговых помощников с помощью мультязычных 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 мультязычных адверсариальных атак для вышеописанной задачи.

- Предлагаем метод адверсариального предобучения.

В результате работы мы ожидаем получить следующие результаты:

- Мультязычные модели обучены решать задачу заполнения слотов и классификации интенгов.
- 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 мультязычными 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 три статьи на эту тему — две в марте [6, 10] и одна в конце апреля [9] 2021 года.

2 Обзор литературы

2.1 Мультязычные 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 [5].

Вслед за успехом модели Трансформер [11], недавние мультязычные модели такие как m-BERT [2] и XLM-RoBERTa [1] переносят парадигму «предобучение → дообучение под специфическую задачу» в мультязычную 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 [7]. В [13] было показано, что m-BERT показывает очень сильную способность к кросс-язычному переносу знаний. m-BERT превосходит по качеству мультязычные эмбединги 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 мультязычной моделью на текущий момент.

m-BERT обучается на

2.2 Классификация интенгов и заполнение слотов

[12]

2.3 Смещение кодов в адверсариальных атаках на мультязычные модели

Основная - [10]. Побочная - [6]. Пуперпобочная - [9].

2.4 Машинный перевод и выравнивание слов

Перевод - [4].

Выравнивание - [3].

3 Основная часть

3.1 Обучение моделей на датасете MultiAtis++

В своей работе мы обучаем языковые модели решать задачу 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 интенгов 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 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 интенга (Таблица (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 интенгов использовалась обучающая выборка на английском языке. Мы использовали 121 различную метку слотов и 23 различных метки интенгов. Список id используемых объектов, а также списки используемых слотов и интенгов можно найти в приложении.

3.1.2 Архитектура модели

В своей работе мы решаем задачу одновременной классификации интенгов и разметки слотов в предложении с помощью одной модели. Модель имеет два выхода, первый предсказывает интенги, 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 мультязычных 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.

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:

$$L = -\frac{1}{n} \sum_{i=1}^n [y \log(\hat{y})] \quad (1)$$

В своей работе мы будем использовать следующие метрики качества:

- Доля предложений, в которых правильно классифицирован интент:

$$\text{Intent accuracy} = \#\text{sentences} [(I_{pred} = I_{true})] \quad (2)$$

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

$$\text{Slots F1 score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (3)$$

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

$$\text{Semantic accuracy} = \#\text{sentences} [(I_{pred} = I_{true}) \wedge (S_{pred} = S_{true})] \quad (4)$$

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. Для этого мы заменяем часть токенов в предложении на их эквиваленты из других языков. 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.

Так как большинство людей, которые могут использовать смешение кодов в своей речи, билингвы, то в основном смешение кодов происходит между па-

рой языков [8]. 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 токёну на i -ой позиции. В своей работе мы предлагаем следующий вид атаки — пусть мы имеем целевую модель, пару пример-метка и встраиваемый язык (Алгоритм (1)). Тогда мы перебираем токёны в предложении в случайном порядке и стремимся заменить токёну на его эквивалент из встраиваемого языка. Если это приведёт к увеличению ошибки модели, то мы заменяем токёну на предложенного кандидата.

Algorithm 1 Общая схема адверсариальной атаки

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

Ensure: Адверсариальный пример x'

```

 $\mathcal{L}_x = \text{GetLoss}(\mathcal{M}, x, y)$ 
for  $i$  in permutation(len( $x$ )) do
    Candidates = GetCandidates( $\mathcal{M}, x, y, \text{token\_id} = i$ )
    Losses = GetLoss( $\mathcal{M}, \text{Candidates}$ )
    if Candidates and max(Losses) >  $\mathcal{L}_x$  then
         $\mathcal{L}_x = \text{max}(\text{Losses})$ 
         $x, y = \text{Candidates}[\text{argmax}(\text{Losses})]$ 
    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 токёнов на соответствующие языки (Алгоритм (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].

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 (5)).

Algorithm 2 Word-level атака

Require: Словарь переводов с исходного на встраиваемый язык \mathbb{T}

```

function GETCANDIDATES( $\mathcal{M}$ ,  $x$ ,  $y$ , token_id)
  if  $x[\text{token\_id}]$  in  $\mathbb{T}[\mathbb{L}]$  then
    tokens =  $\mathbb{T}[\mathbb{L}][x[\text{token\_id}]]$ 
     $x[\text{token\_id}]$  = tokens
     $y[\text{token\_id}]$  = ExtendSlotLabels( $y[\text{token\_id}]$ , len(tokens))
  end if
  return  $x$ ,  $y$ 
end function

```

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. 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].

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

Algorithm 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  $x[\text{token\_id}]$  in  $\mathbb{A}[\mathbb{L}]$  then
    tokens =  $\mathbb{A}[\mathbb{L}][x[\text{token\_id}]]$ 
     $x[\text{token\_id}]$  = tokens
     $y[\text{token\_id}]$  = ExtendSlotLabels( $y[\text{token\_id}]$ , len(tokens))
  end if
  return  $x$ ,  $y$ 
end function
```

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 смещением кодов.

Предлагаемый нами метод адверсариального предобучения состоит из нескольких шагов:

- 1 Генерация выборки для задачи маскированного моделирования языка.
- 2 Дообучение тела мультязычной модели на сгенерированной выборке в режиме предсказания маскированных токенов.
- 3 Загрузка дообученного тела модели перед началом обучения для задачи одновременного заполнения слотов и классификации интенгов.

3.3.1 Генерация адверсариальной выборки

Для генерации выборки используется адаптация алгоритма phrase-level адверсариальной атаки (Алгоритм (4)). Разница заключается в том, что токены заменяются на их эквиваленты с некоторой вероятностью. Таким образом, для генерации выборки не требуется обученная модель.

Algorithm 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  $\mathbb{L}$  in  $\mathbb{L}_1, \dots, \mathbb{L}_n$  do  
  for x in X do  
    for i in permutation(len(x)) do  
      Candidates = GetCandidates( $\mathcal{M}$ , x, y, token_id = i)  
      if Candidates and  $\mathcal{U}(0, 1) > 0.5$  then  
        x, _ = random.choice(Candidates)  
      end if  
    end for  
    X'.append(x)  
  end for  
end for  
return X'
```

Выборка является конкатенацией сгенерированных выборок для всех шести языков кроме английского представленных в датасете. 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)).

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.

3.3.2 Дообучение тела модели

После генерации адверсариальной выборки мы дообучаем предобученную мультязычную 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. Мы дообучаем обе мультязычные 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 мультязычной модели для задачи одновременного заполнения слотов и классификации интенгов мы загружаем дообученное тело модели.

3.4 Результаты

3.4.1 Решение задачи классификации интенгов и заполнения слотов

	en	de	es	fr	ja	pt	zh	avg
xlm-r	0.980	0.976	0.967	0.971	0.970	0.967	0.964	0.971
m-bert	0.979	0.976	0.958	0.968	0.955	0.968	0.956	0.966
xlm-r en	0.902	0.875	0.878	0.879	0.785	0.775	0.848	0.849
m-bert en	0.952	0.820	0.870	0.875	0.747	0.838	0.816	0.846

Таблица 2: 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	0.944	0.939	0.908	0.924	0.929	0.924	0.942	0.930
m-bert	0.947	0.945	0.885	0.926	0.935	0.924	0.945	0.930
xlm-r en	0.870	0.669	0.751	0.612	0.573	0.673	0.738	0.698
m-bert en	0.899	0.558	0.783	0.534	0.622	0.518	0.680	0.656

Таблица 3: 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	0.826	0.826	0.697	0.807	0.739	0.800	0.779	0.782
m-bert	0.854	0.854	0.653	0.804	0.740	0.808	0.796	0.787
xlm-r en	0.559	0.250	0.347	0.151	0.000	0.204	0.132	0.235
m-bert en	0.672	0.188	0.401	0.195	0.052	0.177	0.213	0.271

Таблица 4: Сравнение моделей между собой на тестовой выборке датасета MultiAtis++ по метрике **Semantic accuracy**. По колонкам языки тестовых подвыборок, по рядам тестируемые модели.

3.4.2 Качество моделей после адверсариальных атак

	de	es	fr	ja	pt	zh	avg
xlm-r	0.931	0.877	0.849	0.825	0.901	0.872	0.876
m-bert	0.893	0.891	0.872	0.820	0.853	0.852	0.863
xlm-r en	0.809	0.783	0.774	0.677	0.554	0.728	0.721
m-bert en	0.811	0.760	0.793	0.723	0.760	0.777	0.771

Таблица 5: 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	0.767	0.589	0.603	0.552	0.598	0.747	0.642
m-bert	0.685	0.517	0.510	0.428	0.494	0.684	0.553
xlm-r en	0.642	0.467	0.499	0.508	0.543	0.641	0.550
m-bert en	0.539	0.385	0.419	0.362	0.391	0.585	0.447

Таблица 6: 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	0.343	0.109	0.086	0.170	0.077	0.278	0.177
m-bert	0.228	0.083	0.058	0.081	0.038	0.213	0.117
xlm-r en	0.201	0.057	0.056	0.013	0.023	0.042	0.065
m-bert en	0.136	0.029	0.032	0.005	0.011	0.115	0.055

Таблица 7: 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	0.954	0.946	0.928	0.952	0.964	0.950	0.949
m-bert	0.948	0.935	0.939	0.951	0.940	0.934	0.941
xlm-r en	0.808	0.836	0.740	0.750	0.442	0.784	0.727
m-bert en	0.809	0.833	0.834	0.805	0.861	0.829	0.829

Таблица 8: 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	0.802	0.829	0.751	0.444	0.813	0.609	0.708
m-bert	0.784	0.804	0.758	0.450	0.783	0.619	0.700
xlm-r en	0.627	0.704	0.569	0.365	0.680	0.561	0.584
m-bert en	0.539	0.699	0.531	0.366	0.530	0.563	0.538

Таблица 9: 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	0.511	0.511	0.336	0.115	0.522	0.209	0.368
m-bert	0.487	0.438	0.344	0.114	0.433	0.256	0.345
xlm-r en	0.163	0.229	0.099	0.013	0.070	0.064	0.106
m-bert en	0.122	0.219	0.085	0.041	0.087	0.106	0.110

Таблица 10: 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.

3.4.3 Влияние метода адверсариального предобучения

	en	de	es	fr	ja	pt	zh	avg
xlm-r adv	0.981	0.974	0.964	0.976	0.972	0.967	0.967	0.972
m-bert adv	0.975	0.976	0.964	0.972	0.960	0.970	0.962	0.968
xlm-r en + adv	0.928	0.890	0.913	0.872	0.789	0.881	0.816	0.870
m-bert en + adv	0.959	0.848	0.901	0.893	0.719	0.901	0.759	0.854

Таблица 11: 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	0.947	0.940	0.906	0.929	0.928	0.929	0.946	0.932
m-bert adv	0.950	0.942	0.900	0.928	0.935	0.920	0.946	0.932
xlm-r en + adv	0.888	0.729	0.788	0.623	0.447	0.743	0.718	0.705
m-bert en + adv	0.900	0.566	0.759	0.557	0.416	0.554	0.604	0.622

Таблица 12: 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	0.833	0.832	0.693	0.813	0.746	0.809	0.792	0.788
m-bert adv	0.861	0.854	0.682	0.805	0.738	0.799	0.797	0.791
xlm-r en + adv	0.613	0.397	0.404	0.109	0.005	0.419	0.136	0.298
m-bert en + adv	0.674	0.266	0.366	0.265	0.004	0.278	0.136	0.284

Таблица 13: 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	0.935	0.885	0.895	0.838	0.918	0.857	0.888
m-bert adv	0.923	0.894	0.890	0.866	0.899	0.885	0.893
xlm-r en + adv	0.842	0.817	0.812	0.613	0.811	0.721	0.769
m-bert en + adv	0.865	0.828	0.854	0.760	0.856	0.750	0.819

Таблица 14: 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	0.768	0.609	0.591	0.519	0.608	0.744	0.640
m-bert adv	0.704	0.532	0.535	0.470	0.563	0.685	0.581
xlm-r en + adv	0.648	0.509	0.508	0.455	0.542	0.663	0.554
m-bert en + adv	0.531	0.405	0.371	0.441	0.417	0.563	0.455

Таблица 15: 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	0.347	0.134	0.093	0.099	0.089	0.283	0.174
m-bert adv	0.302	0.114	0.087	0.121	0.070	0.237	0.155
xlm-r en + adv	0.236	0.094	0.065	0.012	0.073	0.128	0.101
m-bert en + adv	0.187	0.073	0.036	0.044	0.038	0.110	0.081

Таблица 16: 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	0.959	0.958	0.930	0.960	0.958	0.947	0.952
m-bert adv	0.956	0.950	0.944	0.958	0.955	0.946	0.951
xlm-r en + adv	0.870	0.857	0.811	0.795	0.853	0.776	0.827
m-bert en + adv	0.846	0.890	0.893	0.766	0.901	0.784	0.847

Таблица 17: 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	0.809	0.847	0.771	0.432	0.820	0.617	0.716
m-bert adv	0.807	0.848	0.790	0.446	0.822	0.635	0.725
xlm-r en + adv	0.683	0.776	0.648	0.326	0.721	0.570	0.621
m-bert en + adv	0.613	0.754	0.621	0.324	0.631	0.523	0.578

Таблица 18: 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	0.538	0.536	0.347	0.140	0.536	0.256	0.392
m-bert adv	0.544	0.551	0.466	0.136	0.559	0.286	0.424
xlm-r en + adv	0.303	0.375	0.196	0.008	0.322	0.079	0.214
m-bert en + adv	0.254	0.358	0.241	0.019	0.306	0.109	0.214

Таблица 19: 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.

4 Заключение

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] Alexandre Klementiev, Ivan Titov и Binod Bhattacharai. «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>.
- [6] Jitin Krishnan и др. «Multilingual Code-Switching for Zero-Shot Cross-Lingual Intent Prediction and Slot Filling». В: *ArXiv abs/2103.07792* (2021).
- [7] Chi-Liang Liu и др. «What makes multilingual BERT multilingual?» В: *ArXiv abs/2010.10938* (2020).
- [8] 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.
- [9] Sebastin Santy, Anirudh Srinivasan и Monojit Choudhury. «BERTologiCoMix: How does Code-Mixing interact with Multilingual BERT?» В: *Proceedings of the Second Workshop on Domain Adaptation for NLP*. Kyiv, Ukraine: Association for Computational Linguistics, апр. 2021, с. 111—121. URL: <https://www.aclweb.org/anthology/2021.adaptnlp-1.12>.

- [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

Приложение А. Алгоритм замены слотов в атаке

Algorithm 5 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
```
