Козачок А.В., Кузькин П.А. Использование кодировки BERT для борьбы с атакой MADLIB при обнаружении SMS-спама // Международный журнал информационных технологий и энергоэффективности. – 2023. –
Т. 8 № 10(36) с. 9–21

Международный журнал информационных технологий и энергоэффективности

Сайт журнала:
http://www.openaccessscience.ru/index.php/ijcse/

УДК 004

ИСПОЛЬЗОВАНИЕ КОДИРОВКИ BERT ДЛЯ БОРЬБЫ С АТАКОЙ MADLIB ПРИ ОБНАРУЖЕНИИ SMS-СПАМА

[1]Козачок А.В., [2]Кузькин П.А.
ФГБУО ВО «МИРЭА - РОССИЙСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ», Москва, Россия, (119454, г. Москва, просп. Вернадского, 78, стр. 4.), e-mail: [1]kozachok_a@mirea.ru, [2]kuzk1n.p.a@yandex.ru

Одна из уловок, используемых для обмана спам-фильтров, заключается в замене слов синонимами или похожими словами, которые делают сообщение неузнаваемым алгоритмами обнаружения. В этой статье мы исследуем, может ли недавняя разработка языковых моделей, чувствительных к семантике и контексту слов, таких как BERT от Google, быть полезной для преодоления этой состязательной атаки, называемой "Mad-lib. Используя набор данных из 5572 SMS-сообщений со спамом, мы сначала установили базовый уровень эффективности обнаружения, используя широко известные модели векторизации текстов (BoW и TFIDF) и новую модель BERT в сочетании с различными алгоритмами классификации (дерево решений, kNN, SVM, логистическая регрессия, наивный Байес, многослойный персептрон). Затем мы создали тезаурус словаря, содержащегося в этих сообщениях, и провели эксперимент с атакой "Mad-lib", в ходе которого мы модифицировали каждое сообщение из сохранённого подмножества данных (не использованного в базовом эксперименте) с разной частотой замены исходных слов синонимами из тезауруса. Наконец, мы оценили эффективность обнаружения трёх моделей векторизации текстов (BoW, TFIDF и BERT) в сочетании с лучшим классификатором из базового эксперимента (SVM). Мы обнаружили, что классические модели достигли 94% сбалансированной точности (BA) в исходном наборе данных, тогда как модель BERT получила 96%. С другой стороны, эксперимент с атакой "Mad-lib" показал, что кодировкам BERT удаётся поддерживать аналогичную производительность BA на уровне 96% при средней частоте замены 1,82 слова на сообщение и 95% при замене 3,34 слова на сообщение. В отличие от этого, производительность BA кодеров BoW и TFIDF снизилась по случайности. Эти результаты намекают на потенциальное преимущество моделей BERT для борьбы с подобными хитроумными атаками, в некоторой степени компенсируя неправильное использование семантических отношений в языке.

Ключевые слова: Классификация спама, состязательная спам-атака, кодирование BERT, модели машинного обучения, большие данные.

USING BERT ENCODING TO COMBAT MADLIB ATTACK WHEN SMS SPAM IS DETECTED

[1]Kozachok A.V., [2]Kuzkin P.A.
MIREA - RUSSIAN TECHNOLOGICAL UNIVERSITY, Moscow, Russia (119454, Moscow, avenue. Vernadsky, 78, b. 4), e-mail: [1]kozachok_a@mirea.ru, [2]kuzk1n.p.a@yandex.ru

One of the tricks used to trick spam filters is to replace words with synonyms or similar words that make the message unrecognizable by detection algorithms. In this article, we explore whether the recent development of language models sensitive to the semantics and context of words, such as Google's BERT, could be useful in

overcoming this adversarial attack called "Mad-lib. Using a dataset of 5572 spam SMS messages, we first established a baseline level of detection efficiency using well-known text vectorization models (BoW and TFIDF) and the new BERT model in combination with various classification algorithms (decision tree, kNN, SVM, logistic regression, naive Bayes, multilayer perceptron). Then we created a thesaurus of the dictionary contained in these messages and conducted an experiment with a "Mad-lib" attack, during which we modified each message from a saved subset of data (not used in the basic experiment) with a different frequency of replacing the original words with synonyms from the thesaurus. Finally, we evaluated the effectiveness of detecting three text vectorization models (BoW, TFIDF and BERT) in combination with the best classifier from the basic experiment (SVM). We found that the classical models achieved 94% balanced accuracy (BA) in the original dataset, whereas the BERT model got 96%. On the other hand, the experiment with the "Mad-lib" attack showed that BERT encodings manage to maintain similar BA performance at 96% with an average replacement frequency of 1.82 words per message and 95% with 3.34 words per message. In contrast, the performance of the BA coders BoW and TFIDF decreased by chance. These results hint at the potential advantage of BERT models for dealing with such clever attacks, to some extent compensating for the misuse of semantic relations in the language.

Keywords: Spam classification, adversarial spam attack, BERT encoding, machine learning models, big data.

Введение

Нежелательная электронная почта (спам) остается глобальной проблемой, на долю которой, по данным некоторых поставщиков сетевой безопасности, приходится до 85% ежедневного трафика сообщений. Несмотря на то, что спам-фильтры используют преимущества технологий искусственного интеллекта для повышения эффективности обнаружения, эти алгоритмы всё еще могут быть обмануты злоумышленными атаками, т.е. тщательно продуманными модификациями контента, которые пытаются обойти фильтры, но, тем не менее, легко распознаваемы человеком, путём введения либо безобидных, не связанных между собой, либо запутанных слов или символов [16, 23]. Одна из таких стратегий, называемая атакой "Mad-lib", состоит в замене терминов, относящихся к спамовым, синонимами или подобными словами, предотвращающими распознавание сообщения фильтром как нежелательной почты [16].

В нашем эксперименте мы предполагаем использовать последние достижения в семантических и контекстно-зависимых языковых моделях, разработанных для задач обработки естественного языка (NLP), для борьбы с атаками, основанными на замене слов. Одной из них является модель BERT, разработанная Google [8], которая продемонстрировала самую современную производительность в одиннадцати задачах NLP. По сути, эта модель способна представлять короткий документ (состоящий из последовательности до 512 слов) в виде числового вектора, встроенного в пространство из 768 позиций, что соответствует плотному и распределённому представлению функций документа. В отличие от встраиваемых представлений отдельных слов (таких как Word2Vec, GloVe или FastText, см., например, [15]), BERT представляет собой модель глубокой сети, которая включает внутренние блоки механизмов внимания, которые кодируют последовательность слов в векторы в зависимости от контекста [8], фиксируя лексические, семантические и грамматические особенности, связанные с порядком, в котором, как правило, одно слово предшествует другому или следует за ним в определенных предложениях. Одним из выходных данных BERT является векторное представление входного документа, которое мы будем использовать для поиска сходств (расстояний) между спам-сообщениями в результирующем пространстве встраивания, аналогично упомянутые выше встраивания слов используются для сопоставления похожих слов с близкими местоположениями.

Исходя из этого, в данной работе мы намерены оценить полезность применения модели BERT для распознавания текстовых последовательностей спама, которые отличаются только некоторыми лексическими терминами, но которые все ещё сохраняют своё нежелательное намерение, тем самым способствуя обнаружению атак типа Mad-lib.

1.1. Похожие работы

Тактика состязательной атаки обычно предполагает тщательную обработку содержимого входных данных, чтобы нарушить ожидаемое поведение модели прогнозирования [17]. Изучение враждебной среды привлекло внимание более десяти лет назад, когда были обнаружены уязвимости спам-фильтров, сталкивающихся с такого рода манипуляциями [4]. С тех пор многие враждебные атаки и способы защиты были описаны в различных приложениях, таких как оскорбительные комментарии в Интернете и обнаружение ненормативной лексики [10, 22, 23, 25], классификация медицинских изображений [9] или идентификация объектов в компьютерном зрении [11, 2], и это лишь некоторые из них.

В случае задач классификации текста атаки обычно выполняются путём искажения признаков или содержания текстовой последовательности [23]. Более конкретно, в области состязательных атак на спам-фильтры было охарактеризовано несколько приёмов [12, 16, 24]: отравление, введение хороших слов, запутывание спам-слов, смена ярлыков и замена синонимов. Наше исследование сосредоточено на последнем, используя проактивный подход [4], т.е. предвосхищая, моделируя стратегию соперничества и противодействуя ей.

Что касается использования кодировок BERT для извлечения признаков спама, недавно была предложена модифицированная трансформаторная модель для улучшения производительности обнаружения классификаторов спама [19]. Другие модифицированные модели, производные от BERT, были предложены для эффективного обнаружения вредоносных фишинговых электронных писем [18], в то время как BERT с расширенной функциональностью также применялся для фильтрации многоязычных спам-сообщений [7] и для блокировки поддельных публикаций о COVID [13], с многообещающими результатами.

1.2. Ценность исследования

Основные положениями исследования, которые имеют ценность:

- показано, что кодеры BoW, TFIDF и BERT способны извлекать признаки для идентификации спама, используя широко используемые алгоритмы классификации, причём BERT работает немного лучше [3];
- описана автоматическая состязательная процедура для проведения атаки Mad-lib на выбранный набор данных;
- предоставлены эмпирические доказательства того, что BERT способен противостоять атакам Mad-lib, в то время как BoW или TFIDF уязвимы.

2. Методы

2.1. Дорожная карта исследования

Исследование проводилось в соответствии с этапами, проиллюстрированными на дорожной карте (Рисунок 1), которые описаны далее.

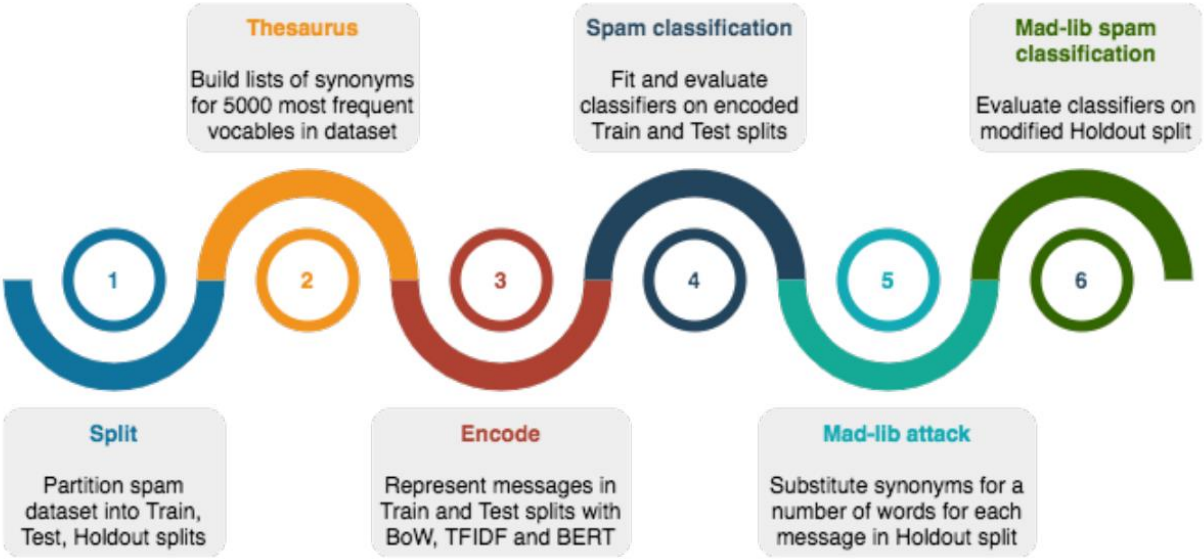

Рисунок 1 – Дорожная карта исследования

1) **Разделение набора данных.** Мы работали с набором данных SMS-спама из репозитория UCI. Набор данных несбалансирован, поскольку из общего числа 5574 сообщений 4827 помечены как нежелательные и только 747 - как спам. Сообщения довольно короткие (при средней длине в 14,5 слов они представляют интересную проблему для алгоритмов фильтрации на основе контента [3]). Мы использовали случайную выборку без замены, чтобы разделить этот набор данных на три подмножества: обучение (60%), тестирование (20%) и удержание (20%).

2) **Создание тезауруса.** Мы извлекли словарь из 5000 наиболее часто встречающихся терминов из всего набора данных и использовали их в качестве ключевых слов в тезаурусе. Для каждого ключевого слова список синонимов был автоматически удалён с соответствующей страницы ввода на веб-сайте www.dictionary.com.

3) **Кодировка документа.** Сообщения в каждом разделении представлены с использованием двух кодировок, обычно используемых при фильтрации спама, Bag-of-Words (BoW) и Inverse Frequency of Document Frequency (TFIDF) [15], а также недавно введённых представлений двунаправленного кодера от Transformers (BERT) [8]. BoW и TFIDF – это упрощённые представления, которые сопоставляют слова в документе с вектором частот, индексируемым словарём (последний нормализуется по доле документов, содержащих эти слова). Эти сопоставления фиксируют лексические особенности, игнорируя синтаксис или семантику. Для этих моделей мы предварительно обрабатываем текст, удаляя стоп-слова на английском языке, переводя его в нижний регистр и применяя стемминг и токенизацию. Однако BERT – это языковая модель, обучаемая как глубокая двунаправленная сеть, обусловленная как левым, так и правым контекстом слов во вводимом тексте, а также учитывающая семантические отношения. Одним из выходных данных на верхнем уровне сети является вектор из 768 позиций, который кодирует вложение всего входного предложения. Мы будем использовать его как вектор контекстных характеристик и семантических связей между последовательностью слов, составляющих сообщение, уделяя особое внимание его способности генерировать похожие спам-сообщения, отличающиеся лексическими

вариациями, в близких местах пространства встраивания, независимо от фактической интерпретации этих характеристик. Кроме того, очистка текста для этой модели была минимальной, в основном преобразование в нижний регистр и применение токенизатора BERT [8].

4) **Классификация спама.** На этом этапе была проведена первая серия экспериментов, чтобы оценить, насколько хорошо алгоритмы классификации работают с исходными сообщениями. Для этой цели мы использовали обучающие и тестовые разбиения, представленные тремя кодировками в качестве входных характеристик различных алгоритмов классификации, которые регулярно используются для задач классификации текста [15, 1, 14]: дерево решений, наивный Байес, kNN, метод опорных векторов (SVM), логистическая регрессия и многослойный персептрон (MLP).

5) **Атака “Mad-lib”.** Были проведены две атаки на удерживаемое подмножество, где в каждом сообщении была предпринята попытка заменить 5 или 10 слов, выбранных случайным образом, используя синонимы из ранее созданного тезауруса. В результате были получены два модифицированных подмножества Mad-lib.

6) **Классификация спама “Mad-lib”.** В этой второй серии экспериментов ранее обученные классификаторы оценивались в модифицированных наборах “Mad-lib”, после кодирования с помощью трёх вышеупомянутых моделей представления.

2.2. Описание эксперимента

Эксперименты проводились в соответствии с блок-схемой, описанной на Рисунке 2. Набор данных разделён на три раздела: обучающий, тестовый и удерживающий. Первый эксперимент был проведён для оценки базовой эффективности обнаружения спама на исходном наборе данных для целей сравнения в последующем эксперименте со спамом с атакой “Mad-lib”.

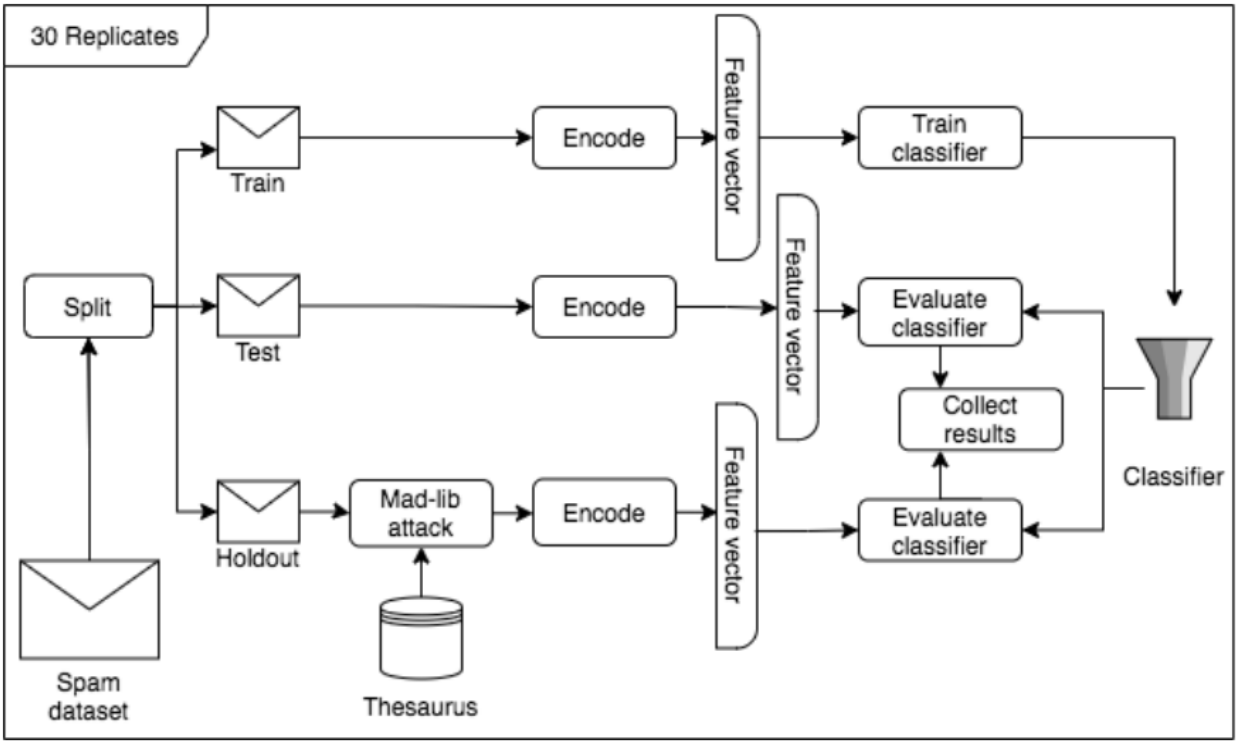

Рисунок 2 – Блок-схема исследования

Первоначально сообщения в обучающем и тестовом наборах были закодированы с помощью трёх моделей представления (BoW, TFIDF, BERT) для получения векторов из 768 признаков (поскольку это неотъемлемый размер плотных векторов, генерируемых BERT, мы устанавливаем базовый размер словаря для BoW и TFIDF соответственно). Затем полученные векторы признаков передаются в вышеупомянутые алгоритмы классификации. Каждый классификатор обучается с использованием закодированных векторов разделения вместе с их соответствующими метками. После обучения их производительность оценивается в тестовом разбиении с использованием показателей точности (ACC), прецизионности (PR), чувствительности (SE) [26] и сбалансированной точности (BA) [6]. Последний показатель был сочтён наиболее подходящим для данной конкретной задачи, учитывая, что набор данных сильно несбалансирован. Они определяются следующими уравнениями:

$$ACC = \frac{TP + TN}{P + N}, \qquad BA = \frac{1}{2}\left(\frac{TP}{P} + \frac{TN}{N}\right), \qquad PR\frac{TP}{TP + FP}, \qquad SE = \frac{TP}{TP + FN},$$

где P и N – общее количество сообщений спамовых и легитимных соответственно, TP и FP - правильно и неправильно классифицированный спам, а TN и FN - правильно и неправильно классифицированные легитимные сообщения, соответственно. Результаты собираются в общей сложности из 30 реплик (с различными выборками обучающих и тестовых разделов), чтобы уменьшить их вариабельность из-за случайности процедуры выборки.

Вторая серия экспериментов была сосредоточена на оценке того, как ранее обученные классификаторы реагируют на атаку “Madlib” (замена некоторых слов синонимами) с использованием различных моделей представления. Были проведены две различные атаки, одна из которых пыталась заменить 5 или 10 слов случайным образом. Необходимо также

обратить внимание, что, поскольку некоторые слова могут не иметь синонимов в тезаурусе, фактическое количество замен может быть меньше (см. примеры в Таблице 1). Как только атака завершена, измененные сообщения кодируются с помощью трёх моделей представления для получения соответствующих векторов признаков, и они передаются в ранее обученные классификаторы, чтобы оценить их производительность с помощью тех же показателей, упомянутых выше. В этом случае результаты 30 реплик усредняются, чтобы уменьшить вариабельность из-за случайной выборки и процессов замены слов.

Таблица 1 – Четыре примера атак “Mad-lib” (сверху до атаки, снизу – после)

What will we do in the shower, baby? what will we do in the **shower bath infant**
Good Morning my Dear ..... Have a great & successful day. good **day** my **darling** have a great **ampere victorious today**
Refused a loan? Secured or Unsecured? Can’t get credit? Call free now 0800 195 6669 or text back ’help’ & we will! **turn down** a loan secured or unsecured can t **turn** credit call free now 0800 195 6669 or text back **care** we will
Camera - You are awarded a SiPix Digital Camera ! call 09061221066 fromm landline. Delivery within 28 days. **cine-camera** you are **grant** a sipix **digital-analog converter cartridge** call 09061221066 fromm landline **serving** within 28 days

2.3. Детали реализации

Модели и эксперименты были реализованы на языке Python 3.8.5 с использованием библиотек scikit-learn 0.24.0 [20], PyDictionary [5] и SimpleTransformers [21], которые были выполнены в Google Colab с ускорителем GPU.

Выбранные параметры модели для алгоритмов, используемых в экспериментах, показаны в Таблице 2.

Таблица 2 – Выбранные параметры модели для алгоритмов, используемых в экспериментах

Алгоритмы классификации	
Decision Tree	max depth=10
Naive Bayes	default parameters
kNN	k = 15
SVM (linear)	C=1, loss=‘squared hinge’
Logistic Regression	default parameters
MLP	hidden layer sizes=(10,), alpha=1, max iter=1000
SVM (gaussian)	gamma=.01, C=100
Модели представления (векторизации)	
BoW, TFIDF	stemming, lowercase, stop words, max features=768
BERT	model=‘xlm-r-bert-base-nli-stsb-mean-tokens’

3. Результаты

3.1. Эксперименты по обнаружению спама

Результаты этих экспериментов обобщены в Таблица 3, где приведены средние значения и стандартные отклонения показателей производительности, сгруппированные по модели кодирования и алгоритму классификации.

Таблица 3 – Результаты классификации спама

Кодировщик	**Классификатор**	**Метрики**			
		BA	ACC	SE	PR
BERT	Decision Tree	85.2±1.7%	93.3±0.9%	73.9±3.1%	76.6±4.1%
	Naive Bayes	93.1±1.0%	95.8±0.5%	89.3±2.0%	82.0±3.2%
	kNN	93.2±1.3%	97.0±0.5%	87.9±2.7%	89.8±2.4%
	SVM (linear)	96.3±0.9%	98.4±0.3%	93.2±1.8%	95.3±1.9%
	Logistic Regression	96.3±0.9%	98.7±0.3%	93.0±1.8%	97.5±1.3%
	MLP	96.6±0.9%	98.8±0.3%	93.5±1.8%	97.4±1.3%
	SVM (gaussian)	95.1±1.1%	98.6±0.4%	90.3±2.2%	99.4±0.9%
BoW	Decision Tree	84.1±1.8%	95.0±0.5%	69.3±3.8%	91.3±3.2%
	Naive Bayes	82.6±1.2%	76.4±1.5%	91.0±2.2%	35.1±2.7%
	kNN	59.8±1.7%	89.3±0.8%	19.7±3.4%	99.5±1.7%
	SVM (linear)	93.8±1.2%	97.7±0.3%	88.6±2.5%	93.3±1.5%
	Logistic Regression	92.8±1.4%	97.9±0.4%	85.9±2.7%	97.8±1.4%
	MLP	92.2±1.3%	97.7±0.4%	84.7±2.6%	97.8±1.5%
	SVM (gaussian)	93.6±1.4%	97.5±0.4%	88.3±2.8%	92.6±2.0%
TFIDF	Decision Tree	86.0±2.0%	94.9±0.6%	73.8±4.3%	86.1±3.6%
	Naive Bayes	82.8±1.0%	77.9±1.5%	89.6±2.5%	36.4±2.4%
	kNN	55.4±1.1%	88.2±1.0%	10.8±2.2%	99.4±1.9%
	SVM (linear)	94.0±1.4%	98.1±0.5%	88.5±2.8%	96.6±2.0%
	Logistic Regression	87.3±1.6%	96.5±0.5%	74.9±3.1%	98.0±1.4%
	MLP	88.0±1.5%	96.6±0.5%	76.2±3.1%	97.9±1.3%
	SVM (gaussian)	93.9±1.4%	98.0±0.4%	88.4±2.9%	96.3±1.8%

Грубо говоря, результаты сильно варьировались в зависимости от алгоритма классификации. Для кодеров BoW и TFIDF самые низкие показатели были получены с помощью kNN (BA: 59,8% и 55,4% соответственно), а самые высокие - с помощью SVM (BA: 93,8% и 94% соответственно). В случае BERT вариабельность менее заметна (все классификаторы получили BA более 93%, за исключением дерева принятия решений с 85%), при этом MLP является лучшим, достигая BA в 96,6%, за ним следует SVM с 96,3%.

Изучая показатели ACC и SE, мы подтверждаем аналогичные результаты, полученные в предыдущих исследованиях (например, [3]). Мы отмечаем, что характеристики, полученные для SE и PR с использованием представления BERT, более однородны, чем у двух других кодеров.

3.2. Эксперименты со спам-атакой “Madlib”

Результаты этих экспериментов обобщены в Таблице 4, где приведены средние значения и стандартные отклонения показателей производительности, сгруппированные по количеству попыток в модели атаки и кодирования (в качестве классификатора был выбран линейный SVM, поскольку он обеспечивал лучшие показатели во всех трёх моделях).

Таблица 4 – Результаты классификации после спам-атаки “Madlib”

Количество попыток замены слов	**Кодировщик**	**Средняя частота замены слов**	**Метрики**			
			BA	ACC	SE	PR
0	BERT	0.00	96.6±0.9%	98.3±0.4%	94.4±1.9%	92.8±2.3%
	BoW	0.00	54.9±3.8%	79.4±2.5%	21.5±8.0%	21.7±6.5%
	TFIDF	0.00	50.0±0.3%	86.6±0.8%	0.3±0.6%	13.7±28.4%
5	BERT	1.82	96.2±1.0%	97.6±0.5%	94.2±2.0%	88.4±3.1%
	BoW	1.82	55.2±3.7%	80.7±2.2%	20.4±7.7%	23.4±7.0%
	TFIDF	1.82	50.0±0.3%	86.6±0.7%	0.3±0.6%	15.4±29.1%
10	BERT	3.34	95.2±0.9%	96.8±0.6%	93.0±1.7%	84.8±2.8%
	BoW	3.34	55.2±3.3%	82.1±2.1%	18.7±6.8%	25.8±7.8%
	TFIDF	3.34	50.0±0.2%	86.6±0.7%	0.2±0.4%	13.8±30.2%

В целом, результаты подтверждают предположение о полезности модели BERT для противодействия такого рода атакам. Мы сосредоточимся на изучении показателя BA для этого анализа. При первой атаке с нулевыми заменами (то есть с использованием разделения на удержания без изменения исходных сообщений) производительность SVM сохраняется на уровне 96,6%. С другой стороны, для атак с 5 и 10 попытками замены (что соответствует в среднем 1,82 и 3,34 реальным заменам, как объяснено выше) показатель точности модели BERT немного снизился до 96,2% и 95,2% соответственно, примерно на 1% меньше по сравнению с базовым экспериментом.

Напротив, эти результаты также показывают, что в отношении BA производительность кодеров BoW и TFIDF ухудшается на уровнях, близких к случайному. При изучении показателя SE для кодера BoW наблюдается резкое падение до 21,5%, то есть включение терминов, не входящих в выборку, сильно влияет на обнаружение признаков, обычно ассоциируемых со словами, относящимися к спаму, феномен, который усиливается, когда в каждом сообщении делаются подстановки “Mad-lib”.

Заключение

Это исследование предоставило эмпирические доказательства перспективности кодировок BERT в борьбе со спам-атакой Mad-lib. Мы полагаем, что это связано со способностью данной модели представлять семантические и контекстуальные функции языка. Кроме того, другие преимущества BERT заключаются в том, что он не требует предварительной обработки (очистки) текста, а также в его способности распознавать термины, не входящие в словарный запас, благодаря присущему ему методу токенизации. С

вычислительной точки зрения BERT тяжелее, чем более простые кодеры BoW, которые обеспечивают сопоставимую производительность со спамом, не поддающимся подделке злоумышленниками “Mad-lib”.

Поэтому мы предполагаем, что комбинация моделей кодирования была бы реалистичной конфигурацией, лежащей в основе современных спам-фильтров, для обнаружения изменений в поведении, подразумевающих необходимость переподготовки фильтров (например, активация оповещения, когда производительность BoW и BERT начинает сильно отличаться).

Кроме того, мы надеемся, что кодировки BERT помогут противостоять не только враждебному сценарию, описанному в этом документе, но и другим связанным с ним атакам.

Список литературы

1. Чару Аггарвал и Чэнсян Чжай. Обзор алгоритмов классификации текстов. В разделе Интеллектуальный анализ текстовых данных, С.163-222. Спрингер, 2012.
2. Навид Ахтар и Аджмал Миан. Угроза враждебных атак на глубокое обучение в компьютерном зрении: обзор. Доступ IEEE, 6:С.14410-14430, 2018.
3. Тиаго А Алмейда, Хосе Мария Г Идальго и Акебо Ямаками. Вклад в исследование фильтрации SMS-спама: новая коллекция и результаты. В материалах 11-го симпозиума ACM по разработке документов, страницы С.259-262, 2011.
4. Баттиста Биджио и Фабио Роли. Дикие закономерности: десять лет спустя после появления состязательного машинного обучения. Распознавание образов, 84:С.317-331, 2018.
5. Прадипта-Бора. PyDictionary: Модуль ”Реального” словаря для Python (версия 2.0.1), https://github.com/geekpradd/pydictionary , 2021 год.
6. Кей Хеннинг Бродерсен, Чен Сун Онг, Клаас Энно Стефан и Йоахим М. Бухманн. Сбалансированная точность и ее апостериорное распределение. В 2010 году состоялась 20-я международная конференция по распознаванию образов, С. 3121-3124. IEEE, 2010.
7. Цзе Цао и Чэнчжэ Лай. Двуязычная модель обнаружения разнотипного спама, основанная на M-BERT. В IEEE Global Communications Conference, С.1-6. IEEE, 2020 год.
8. Джейкоб Девлин, Мин-Вей Чанг, Кентон Ли и др. БЕРТ: Предварительная тренировка глубоких двунаправленных преобразователей для понимания языка. arXiv: С. 1810.04805, 2018.
9. Сэмюэл Г. Финлейсон, Джон Д. Бауэрс, Джоичи Ито, Джонатан Л. Циттрейн, Эндрю Л. Бим и Айзек С. Кохане. Враждебные атаки на медицинское машинное обучение. Наука, 363(6433):С.1287-1289, 2019.
10. Хоссейн Хоссейни, Срирам Каннан, Баосен Чжан и др. Обманывает перспективный API Google, созданный для обнаружения токсичных комментариев. arXiv: 1702.08138, 2017.
11. Хоссейн Хоссейни, Байсен Сяо и Радха Пувендран. API Google cloud vision не устойчив к помехам. В 2017 году состоялась 16-я международная конференция IEEE по машинному обучению и приложениям (ICMLA), С.101-105. IEEE, 2017.
12. Ниддал Х. Имам и Вассилиос Г. Вассилакис. Обзор атак на детекторы спама в Twitter в конкурентной среде. Робототехника, 8(3):50, 2019.

13. Дебанджана Кар, Мохит Бхардвадж и др. Пожалуйста, никаких слухов! Многоязычный подход для обнаружения поддельных твитов, связанных с COVID. arXiv:2010.06906, 2020.
14. Вандана Корде и Си Намрата Махендер. Классификация текстов и классификаторы: Обзор. Международный журнал по искусственному интеллекту и приложениям, 3(2):85, 2012.
15. Камран Ковсари, Киана Джафари Мейманди, Моджтаба Хайдарисафа и др. Алгоритмы классификации текстов: обзор. Информация, 10(4):150, 2019.
16. Бхаргав Кучипуди, Рави Теджа Наннапанени и Ци Ляо. Состязательное машинное обучение для спам-фильтров. В материалах 15-й международной конференции по доступности, надежности и безопасности, С.1-6, 2020 год.
17. Павел Ласков и Ричард Липпманн. Машинное обучение в состязательной среде. Машинное обучение, (2):115-119, 2010.
18. Янгу Ли, Джошуа Сакс и Ричард Харанг. КАТБЕРТ: Tiny BERT с учетом контекста для обнаружения электронных писем социальной инженерии. arXiv:2010.03484, 2020.
19. Сяосю Лю. Модель преобразования спама для обнаружения SMS-спама. Магистерская диссертация, Университет Оттавы/University of Ottawa, 2021.
20. Фабиан Педрегоса, Гаэль Варокво и др. Scikit-learn: Машинное обучение на Python. Журнал исследований машинного обучения, 12:2825-2830, 2011.
21. Тилина Раджапаксе. Простые трансформеры (2021), https://simpletransformers.ai .
22. Нестор Родригес и Серхио Рохас-Галеано. Защита модели языковой токсичности Google от враждебных атак. Препринт arXiv arXiv: 1801.018
23. Серхио Рохас-Галеано. О препятствовании запутыванию непристойностей. Транзакции ACM в Интернете (TWEB), 11(2):.1-24, 2017.
24. Серхио А Рохас-Галеано. Выявление неалфавитных разновидностей спам-триггерных словосочетаний. Дина, 80(182):15-24, 2013.
25. Сара Суд, Джадд Антин и др. Использование ненормативной лексики в онлайн-сообществах. В материалах конференции SIGCHI по человеческому фактору в вычислительных системах, 2012.
26. Ала Тарват. Методы оценки классификации. Прикладная вычислительная техника и информатика, 17 декабря 2021 года.

References

1. Charu C Aggarwal and ChengXiang Zhai. A survey of text classification algorithms. In Mining text data, pp.163–222. Springer, 2012.
2. Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. IEEE Access, 6:pp.14410–14430, 2018.
3. Tiago A Almeida, Jos´e Mar´ıa G Hidalgo, and Akebo Yamakami. Contributions to the study of SMS spam filtering: new collection and results. In Proceedings of the 11th ACM Symposium on Document Engineering, pp. 259–262, 2011.
4. Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognition,pp.84:317–331, 2018.

5. Pradipta Bora. PyDictionary: A ”Real” Dictionary Module for Python (version 2.0.1), https://github.com/geekpradd/pydictionary, 2021.
6. Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M Buhmann. The balanced accuracy and its posterior distribution. In 2010 20th International Conference on Pattern Recognition, pp. 3121–3124. IEEE, 2010.
7. Jie Cao and Chengzhe Lai. A Bilingual Multi-type Spam Detection Model Based on M-BERT. In IEEE Global Communications Conference, pp. 1–6. IEEE, 2020.
8. Jacob Devlin, Ming-Wei Chang, Kenton Lee, et al. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805, 2018.
9. Samuel G Finlayson, John D Bowers, Joichi Ito, Jonathan L Zittrain, Andrew L Beam, and Isaac S Kohane. Adversarial attacks on medical Machine Learning. Science, 363(6433):pp.1287–1289, 2019.
10. Hossein Hosseini, Sreeram Kannan, Baosen Zhang, et al. Deceiving Google’s perspective API built for detecting toxic comments. arXiv:1702.08138, 2017.
11. Hossein Hosseini, Baicen Xiao, and Radha Poovendran. Google’s cloud vision API is not robust to noise. In 2017 16th IEEE international conference on machine learning and applications (ICMLA), pp. 101–105. IEEE, 2017.
12. Niddal H Imam and Vassilios G Vassilakis. A survey of attacks against twitter spam detectors in an adversarial environment. Robotics, 8(3):pp.50, 2019.
13. Debanjana Kar, Mohit Bhardwaj, et al. No Rumours Please! A Multi-Indic-Lingual Approach for COVID Fake-Tweet Detection. arXiv:2010.06906, 2020.
14. Vandana Korde and C Namrata Mahender. Text classification and classifiers: A survey. International Journal of Artificial Intelligence & Applications, 3(2):85, 2012.
15. Kamran Kowsari, Kiana Jafari Meimandi, Mojtaba Heidarysafa, et al. Text classification algorithms: A survey. Information, 10(4):150, 2019.
16. Bhargav Kuchipudi, Ravi Teja Nannapaneni, and Qi Liao. Adversarial machine learning for spam filters. In Proceedings of the 15th International Conference on Availability, Reliability and Security, pp. 1–6, 2020.
17. Pavel Laskov and Richard Lippmann. Machine learning in adversarial environments. Machine Learning, (2):115–119, 2010.
18. Younghoo Lee, Joshua Saxe, and Richard Harang. CATBERT: Context-Aware Tiny BERT for Detecting Social Engineering Emails. arXiv:2010.03484, 2020.
19. Xiaoxu Liu. A Spam Transformer Model for SMS Spam Detection. Master’s thesis, Universit´e d’Ottawa/University of Ottawa, 2021.
20. Fabian Pedregosa, Gael Varoquaux, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.
21. Thilina Rajapakse. Simple Transformers (2021), https://simpletransformers.ai.
22. Nestor Rodriguez and Sergio Rojas-Galeano. Shielding google’s language toxicity model against adversarial attacks. arXiv preprint arXiv:1801.01828, 2018.
23. Sergio Rojas-Galeano. On obstructing obscenity obfuscation. ACM Transactions on the Web (TWEB), 11(2):1–24, 2017.
24. Sergio A Rojas-Galeano. Revealing non-alphabetical guises of spam-trigger vocables. Dyna, 80(182):15–24, 2013.

25. Sara Sood, Judd Antin, et al. Profanity use in online communities. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2012.
26. Alaa Tharwat. Classification assessment methods. Applied Computing and Informatics, 17, 2021.