НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
сентябрь–октябрь 2023 Том 23 № 5 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
September–October 2023 Vol. 23 No 5 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

doi: 10.17586/2226-1494-2023-23-5-946-954
УДК 004.89

Метод тестирования лингвистических моделей машинного обучения текстовыми состязательными примерами

Артем Бакытжанович Менисов[1]✉, Александр Григорьевич Ломако[2], Тимур Римович Сабиров[3]

[1,2,3] Военно-космическая академия имени А.Ф. Можайского, Санкт-Петербург, 197198, Российская Федерация

[1] vka@mil.ru✉, https://orcid.org/0000-0002-9955-2694
[2] vka@mil.ru, https://orcid.org/0000-0002-1764-1942
[3] vka@mil.ru, https://orcid.org/0000-0002-6807-2954

Аннотация

Введение. В настоящее время интерпретируемость лингвистических моделей машинного обучения неудовлетворительна в связи с несовершенством научно-методического аппарата описания функционирования как отдельных элементов, так и моделей в целом. Одной из проблем, связанной со слабой интерпретируемостью, является низкая надежность функционирования нейронных сетей, обрабатывающих тексты естественного языка. Известно, что небольшие возмущения в текстовых данных влияют на устойчивость нейронных сетей. В работе представлен метод тестирования лингвистических моделей машинного обучения на наличие угрозы проведения атак уклонения. **Метод.** Метод включает в себя следующие генерации текстовых состязательных примеров: случайная модификация текста и сеть генерации модификаций. Случайная модификация текста произведена с помощью омоглифов — переупорядочивания текста, добавления невидимых символов и удаления символов случайным образом. Сеть генерации модификаций основана на генеративно-состязательной архитектуре нейронных сетей. **Основные результаты.** Проведенные эксперименты продемонстрировали результативность метода тестирования на основе сети генерации текстовых состязательных примеров. Преимущество разработанного метода заключается в возможности генерации более естественных и разнообразных состязательных примеров, которые обладают меньшими ограничениями, не требуется многократных запросов к тестируемой модели. Это может быть применимо в более сложных сценариях тестирования, где взаимодействие с моделью ограничено. Эксперименты показали, что разработанный метод позволил добиться лучшего баланса результативности и скрытности текстовых состязательных примеров (например, протестированы модели GigaChat и YaGPT). **Обсуждение.** Результаты работы показали необходимость проведения тестирования на наличие дефектов и уязвимостей, которые могут эксплуатировать злоумышленники с целью снижения качества функционирования лингвистических моделей. Это указывает на большой потенциал в вопросах обеспечения надежности моделей машинного обучения. Перспективным направлением являются проблемы восстановления уровня защищенности (конфиденциальности, доступности и целостности) лингвистических моделей машинного обучения.

Ключевые слова

искусственный интеллект, обработка естественного языка, информационная безопасность, состязательные атаки, тестирование защищенности

Благодарности

Работа выполнена в рамках гранта Президента Российской Федерации для государственной поддержки молодых российских ученых — кандидатов наук МК-2485.2022.4.

Ссылка для цитирования: Менисов А.Б., Ломако А.Г., Сабиров Т.Р. Метод тестирования лингвистических моделей машинного обучения текстовыми состязательными примерами // Научно-технический вестник информационных технологий, механики и оптики. 2023. Т. 23, № 5. С. 946–954. doi: 10.17586/2226-1494-2023-23-5-946-954

Method for testing NLP models with text adversarial examples

Artem B. Menisov[1]✉, Aleksandr G. Lomako[2], Timur R. Sabirov[3]

[1,2,3] Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation

[1] vka@mil.ru✉, https://orcid.org/0000-0002-9955-2694

[2] vka@mil.ru, https://orcid.org/0000-0002-1764-1942

[3] vka@mil.ru, https://orcid.org/0000-0002-6807-2954

Abstract

At present, the interpretability of Natural Language Processing (NLP) models is unsatisfactory due to the imperfection of the scientific and methodological apparatus for describing the functioning of both individual elements and models as a whole. One of the problems associated with poor interpretability is the low reliability of the functioning of neural networks that process natural language texts. Small perturbations in text data are known to affect the stability of neural networks. The paper presents a method for testing NLP models for the threat of evasion attacks. The method includes the following text adversarial examples generations: random text modification and modification generation network. Random text modification is made using homoglyphs, rearranging text, adding invisible characters and removing characters randomly. The modification generation network is based on a generative adversarial architecture of neural networks. The conducted experiments demonstrated the effectiveness of the testing method based on the network for generating text adversarial examples. The advantage of the developed method is, firstly, in the possibility of generating more natural and diverse adversarial examples, which have less restrictions, and, secondly, that multiple requests to the model under test are not required. This may be applicable in more complex test scenarios where interaction with the model is limited. The experiments showed that the developed method allowed achieving a relatively better balance of effectiveness and stealth of textual adversarial examples (e.g. GigaChat and YaGPT models tested). The results of the work showed the need to test for defects and vulnerabilities that can be exploited by attackers in order to reduce the quality of the functioning of NLP models. This indicates a lot of potential in terms of ensuring the reliability of machine learning models. A promising direction is the problem of restoring the level of security (confidentiality, availability and integrity) of NLP models.

Keywords

artificial intelligence, natural language processing, information security, adversarial attacks, security testing

Acknowledgements

The work was carried out within the framework of the grant of the President of the Russian Federation for state support of young Russian scientists — candidates of sciences MK-2485.2022.4.

For citation: Menisov A.B., Lomako A.G., Sabirov T.R. Method for testing NLP models with text adversarial examples. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2023, vol. 23, no. 5, pp. 946–954 (in Russian). doi: 10.17586/2226-1494-2023-23-5-946-954

Введение

В настоящее время сложность технологических решений в области обработки текстов естественного языка, применяемых в информационных системах организаций, постоянно растет. Невозможность качественной проверки данных для обучения и эксплуатации, а также сложность интерпретации результатов работы систем искусственного интеллекта дают злоумышленникам возможность проводить различные компьютерные атаки на модели машинного обучения.

Модели машинного обучения уязвимы для многих действий злоумышленников[1]. Угроза состязательных атак стала одной из важных проблем приложений машинного обучения. Термин «состязательные» нужно понимать в смысле противодействия работе системы искусственного интеллекта [1]. Состязательные атаки манипулируют поведением моделей машинного обучения — модифицированные входные данные приводят к неверному результату их функционирования. В настоящей работе под «текстовыми состязательными атаками» понимается преднамеренное снижение качества функционирования лингвистических моделей машинного обучения, вызванное модификацией входных данных[2]. Большая часть исследований состязательных атак относится к работе с изображениями [2]. Текстовые состязательные примеры по своей природе труднее создать из-за дискретной природы естественного языка. В отличие от изображений, в которых значения пикселов могут регулироваться практически незаметно, модификация текстов естественного языка более заметна и относительно легко идентифицирована. Видно, когда злоумышленники вставляют односимвольные орфографические ошибки [3], а перефразирование [4] часто меняет смысл текста.

Анализ исследований по атакам на лингвистические модели [5–8] показал, что текстовые состязательные примеры можно разделить на следующие типы модификаций данных:

— пунктуационные, которые добавляют или удаляют знаки препинания из текста;
— на уровне символов, которые изменяют отдельные символы в словах, чтобы токенизатор обрабатывал несколько экземпляров, что приводит к снижению производительности;
— на уровне слов, которые используют состязательные слова-кандидаты с учетом ограничения сходства;

[1] Банк данных угроз безопасности информации [Электронный ресурс]. Режим доступа: https://bdu.fstec.ru/threat (дата обращения: 30.05.2023).

[2] ГОСТ Р 59276-2020 Способы обеспечения доверия. Общие положения. Введ. 01.03.93. М.: Стандартинформ, 2021. 11 с.

— на уровне предложений, которые объединяют несколько типов возмущений, что делает результат атаки более кумулятивным.

Для лингвистических моделей машинного обучения входные текстовые данные, отображающиеся как символы с выбранным шрифтом, должны быть представлены в специальном коде, например, Unicode. Текстовые состязательные атаки основаны на механизмах модификации текста для формирования различных закодированных значений с помощью омоглифов — переупорядочивания, добавления невидимых символов и удаления символов.

Опишем более подробно каждый механизм модификации текстовых данных.

Омоглифы — схожие отображения (глифы) разных символов [9], например, символы «М» (\u041c) и «M» (\u004d) являются омоглифами и могут применяться для проведения состязательной атаки.

Переупорядочивание заключается в изменении порядка символов в тексте. Этот механизм модификации используется злоумышленниками в фишинговых атаках при изменении названия вредоносных файлов[1]. Например, для файла «mytextfile.txt» при добавлении символа (\u202e) «mytext[\u202e]file.txt» произойдет следующая модификация «mytexttxt.elif». Антифишинговые системы электронной почты не заблокируют такой файл, и злоумышленники смогут передать вредоносные программы в систему пользователя.

Модификация текста при добавлении невидимых символов, приводит к нарушению кодировки без визуального эффекта. Невидимые символы часто применяются для форматирования текста, например, использование пробела нулевой ширины (ZERO-WIDTH SPACE).

При удалении используются управляющие символы, предназначенные для удаления соседних символов. Примером в Unicode является символ Delete (DEL). Удаления эффективны, потому что они позволяют вставлять символы в текст без их рендеринга, но визуально незаметны.

Злоумышленники могут использовать перечисленные способы модификации текстовых данных для целенаправленного манипулирования результатами лингвистических моделей.

В условиях «белого ящика», когда злоумышленники знают параметры модели машинного обучения, модифицировать текстовые данные можно с помощью алгоритмов, основанных на градиенте, например, FGSM [10] или UAP [11]. В условиях «черного ящика», когда параметры модели машинного обучения неизвестны, злоумышленники могут адаптировать состязательные примеры, которые ухудшили качество функционирования другой модели [12].

Традиционные подходы к компьютерной безопасности [13] изолируют системы от «внешнего пользователя» с помощью комбинации брандмауэров, паролей, шифрования данных и других мер контроля доступа. Напротив, разработчики систем искусственного интеллекта часто приглашают различных пользователей, работающих со своими моделями нейронных сетей и нуждающихся в данных, с целью совместного тестирования разработанных моделей.

[1] What is a Right-to-Left Override Attack? [Электронный ресурс]. Режим доступа: https://cybriant.com/what-is-a-right-to-left-override-attack/ (дата обращения: 30.05.2023).

Формализация тестирования лингвистических моделей

Для формализации задачи тестирования моделей машинного обучения, обрабатывающих текстовые данные, возьмем задачу рубрикации текстов и адаптируем ее для других задач компьютерной лингвистики.

Пусть рубрикация текстов в обычных условиях происходит как $f\colon X \to Y$, где $\{y_1, \ldots, y_i\} \subset Y$ — пространство рубрик, а $x = \{x_1, \ldots, x_j\} \subset X$ — пространство слов. При текстовых состязательных атаках модифицированные данные $\hat{x}$ вызовут неправильную рубрикацию $f(\hat{x}) \neq y$.

Основная цель таких атак — нарушить заявленное качество функционирования модели глубокого обучения небольшими изменениями входных данных. Однако у этих атак есть три основных недостатка: искажения букв могут быть обнаружены средствами проверки грамматики; искажения букв и слов могут изменить значение; образец текста может стать нечитаемым.

Таким образом, дополнительная цель злоумышленников — маскировка состязательных атак, т. е. способность скрывать состязательный контент от безопасных входных данных (текстов).

Определим показатели эффективности текстовых состязательных атак:

— метрика результативности атак, измеряющая несоответствие между обычными и модифицированными данными:

$$M_{res}(x_j, \hat{x}_j) = 1 - \frac{|f(x_j) \cap f(\hat{x}_j)|}{|f(x_j)|}, \qquad (1)$$

другими словами, метрика определена в интервале [0, 1] и показывает, что при $M_{res} = 1$ все модифицированные данные привели к неверному результату, и наоборот, при $M_{res} = 0$ состязательная атака не достигла успеха;

— метрика скрытности атаки, измеряющая возможность обнаружения модификации данных:

$$M_{hid}(y_j, \hat{x}_j) = \begin{cases} 0, \text{ если } f(\hat{x}_j) \approx y_j \\ 1, \text{ в другом случае} \end{cases}.$$

Описание метода тестирования лингвистических моделей

Цель разработанного метода — не развитие способов снижения качества функционирования лингвистических моделей, а повышение надежности и объяснимости таких моделей.

Метод включает следующие варианты к генерации текстовых состязательных примеров: случайная модификация текста и сеть генерации модификаций.

Случайная модификация текста — самый простой подход тестирования, который добавляет омоглифы, переупорядочивает текст, добавляет невидимые символы и удаляет символы случайным образом.

Чтобы снизить качество функционирования злоумышленнику необходимо предварительно создать набор модификаций (T) и набор возможных местоположений в тексте (K). Более конкретно, злоумышленник атакует модель со следующими двумя особенностями.

1. Вместо того, чтобы использовать фиксированную модификацию для всех данных, каждый раз злоумышленник выбирает новое изменение из равномерного распределения, т. е. $t_i \sim U[0, 1]$. Так как модификации выбираются случайным образом из равномерного распределения, то существует большой диапазон возможных модификаций T.
2. Вместо размещения модификации в фиксированном месте текста, или определенном слове, злоумышленники размещают его в случайном месте к, выбранном из предопределенного набора местоположений ($к \in K$).

Первая особенность не ограничена только равномерным распределением, злоумышленники могут использовать различные законы распределения для создания и расположения модификаций.

Вторая особенность к случайной генерации и расположению модификаций успешно реализует текстовые состязательные атаки, однако злоумышленник ограничен, поскольку модификации выбираются из заданного распределения и независимо от целевой модели. Современные результаты генеративно-состязательных нейронных сетей показали их эффективность во многих областях: генерация текста, изображений и др.

Сеть генерации модификаций обладает следующими особенностями:

1) генерируются модифицированные данные;
2) обучение происходит с целевой моделью, выступающей вместо дискриминатора.

В начале злоумышленники определяют множество возможных местоположений модификаций K. Затем обучают сеть генерации модификаций совместно с тестируемой моделью следующим образом:

1) каждую эпоху обучения сети генерации модификаций злоумышленники получают данные от тестируемой модели, вычисляя метрики качества обучения каждой эпохи и ошибок;
2) генерируется n текстовых состязательных примеров;
3) при анализе выходных данных вычисляются результативность атак (1) и обновляется модель генерации модификаций.

Экспериментальное исследование

Создание крупных предварительно обученных лингвистических моделей стало популярным для решения все более сложных и разнообразных задач в области обработки текстов естественного языка. Текстовые состязательные атаки имеют несколько категорий (рис. 1).

Для проведения экспериментов было отобрано несколько лингвистических моделей, представленных в табл. 1.

Рис. 1. Таксономия текстовых состязательных атак.
ИНС — искусственные нейронные сети

Fig. 1. Taxonomy of text adversarial attacks

Таблица 1. Лингвистические модели, отобранные для проведения эксперимента
Table 1. Linguistic models selected for the experiment

Название модели	Прикладная задача	Архитектура	Набор обучающих данных	Качество
IE-Net [14]	Вопросно-ответные системы	Двоичные нейронные сети	SQuAD (Stanford Question Answering Dataset)	F-мера = 0,932
CB-NTR [15]	Рубрикация текста	BERT	Reuters-21578	F-мера = 0,907
ACE [16]	Выявления поименованных сущностей	LSTM, Transformer	CoNLL-2003	F-мера = 0,946
AraBERTv1 [17]	Семантический поиск	BERT	Large-Scale Arabic Book Reviews	Точность (Accuracy) = = 0,867
Bi-LSTM [18]	Выявление фейков	Bi-LSTM	FakeNewsNet	Точность (Accuracy) = = 0,822

Для формирования текстовых состязательных примеров использованы обучающие данные для каждой лингвистической модели. Для тестирования применены случайная модификация текста и сеть генерации модификаций. На каждую модель машинного обучения сформировано 100 состязательных примеров.

Для случайной модификации текста выбраны три доли модификации текста. Результаты случайной модификации представлены в табл. 2.

Для генерации состязательных примеров с помощью сети генерации обучающими данными являлись модифицированные данные из первого этапа эксперимента (случайная модификация). Характеристики обучения генератора и дискриминатора представлены в табл. 3.

Результаты применения сети генерации модификаций представлены в табл. 4.

Обсуждение

Генерация текстовых состязательных примеров имеет относительно более короткую историю, чем генерация графических состязательных примеров, потому что сложно произвести модификацию дискретных данных, при этом сохраняя синтаксис, грамматику и семантику.

Отметим масштабируемость текстовых состязательных атак. Масштабируемость означает, что состязательные примеры, сгенерированные для одной нейронной сети, могут также эффективно использоваться для атаки другой нейронной сети. Это свойство часто применяется в атаках черного ящика, так как детали машинного обучения не сильно влияют на метод тестирования. Для нейронных сетей масштабируемость разделим на три уровня: одна и та же архитектура с разными данными;

Таблица 2. Результаты случайной модификации для генерации текстовых состязательных примеров
Table 2. Random modification results for generating text adversarial examples

Модель	Качество моделей машинного обучения									
	без модификации	после модификации								
		при доле омографов в тексте, %			при доле невидимых символов в тексте, %			при доле символов переупорядочивания, %		
		1	5	10	1	5	10	1	5	10
IE-Net	указаны в табл. 1 (столбец «Качество»)	0,894	0,645	0,342	0,844	0,676	0,289	0,932	0,876	0,765
CB-NTR		0,804	0,639	0,299	0,809	0,630	0,253	0,865	0,735	0,715
ACE		0,811	0,620	0,337	0,818	0,618	0,289	0,847	0,762	0,730
AraBERTv1		0,771	0,645	0,315	0,757	0,584	0,270	0,762	0,754	0,720
Bi-LSTM		0,754	0,589	0,323	0,735	0,515	0,264	0,745	0,734	0,701

Таблица 3. Характеристики сети генерации модификаций
Table 3. Characteristics of the modification generation network

Компоненты	Архитектура	Оптимизатор	Количество эпох	Функция потерь	Качество (значение функции потерь)
Генератор	4 Linear, ReLU	Adam	5	Перекрестная энтропия	1,1210
Дискриминатор	3 Linear, ReLU				0,1974

Таблица 4. Снижение качества моделей машинного обучения от применения сети генерации текстовых состязательных примеров

Table 4. Decrease in the quality of machine learning models from the use of a network for generating text adversarial examples

Модель	Качество		
	Метрика	без модификации	после модификации
IE-Net	F-мера	0,932	0,679
CB-NTR	F-мера	0,907	0,661
ACE	F-мера	0,946	0,642
AraBERTv1	Точность (Accuracy)	0,867	0,593
Bi-LSTM	Точность (Accuracy)	0,822	0,600

Рис. 2. Способы защиты от текстовых состязательных атак

Fig. 2. Protection against text contention attacks

разные архитектуры с одним и тем же приложением; разные архитектуры с разными данными.

В связи с тем, что все лингвистические модели уязвимы для атак злоумышленников, важно разработать соответствующие методы защиты для повышения надежности, прежде чем модели будут развернуты в информационной инфраструктуре. Любая эмпирическая защита не дает гарантии надежности моделей и в итоге может быть нарушена другими модификациями со стороны злоумышленника, поэтому дальнейшие исследования в области парирования текстовых состязательных атак должны быть направлены на разработку сертифицированной защиты, которая может обеспечить требуемую надежность.

Основная цель защиты от текстовых состязательных атак — повышение надежности лингвистических моделей машинного обучения с помощью состязательного обучения и фильтрации (рис. 2). Однако обучение на текстовых состязательных образцах, сгенерированных на основе модификаций, может приводить к снижению качества решения прикладных задач и занимать много вычислительных ресурсов.

Для проверки качества двух вариантов подхода использована не только метрика снижения качества работы лингвистических моделей, но и скрытность, измерение которой возможно с помощью проверки орфографии (например, Microsoft Office 2019). Результативность и скрытность тестирования разных вариантов модификации представлены в табл. 5 и 6.

Учтем, что невидимые символы и символы переупорядочивания имеют ограниченное множество, а при добавлении их в систему фильтрации позволит значительно увеличивать их выявление.

Для создания текстовых состязательных примеров применяются различные методы на основе градиентного спуска [19], генетического алгоритма [8], роя частиц [20], жадных алгоритмов [21] и автоэнкодеров [22]. Несмотря на то, что эти методы могут обеспечить высокую результативность, они обладают главным недостатком – необходимостью обеспечения неограниченного доступа к тестируемой лингвистической модели.

Сеть генерации модификаций обладает лучшим балансом результативности и скрытности тестовых состязательных атак, что позволяет значительно снизить устойчивость функционирования любых лингвистических моделей. Подтвержденная уязвимость позволила снизить качество таких моделей как GigaChat[1], и YaGPT[2]. На рис. 3 красным цветом выделены состязательные примеры, которые визуально идентичны нормальным и соответствуют безопасным данным, что и требовалось получить в настоящей работе.

[1] GigaChat — нейросетевая модель от Сбера на русском языке [Электронный ресурс]. Режим доступа: https://developers.sber.ru/portal/products/gigachat (дата обращения: 30.05.2023).

[2] Алиса, давай придумаем... [Электронный ресурс]. Режим доступа: https://yandex.ru/project/alice/yagpt (дата обращения: 30.05.2023).

Таблица 5. Результативность текстовых состязательных примеров, рассчитанная по формуле (1)
Table 5. Performance of text adversarial examples, calculated by formula (1)

Модель	Качество моделей машинного обучения после модификации									Сеть генерации
	при доле омографов в тексте, %			при доле невидимых символов в тексте, %			при доле символов переупорядочивания, %			
	1	**5**	**10**	**1**	**5**	**10**	**1**	**5**	**10**	
IE-Net	0,038	0,287	0,590	0,088	0,256	0,643	0,000	0,056	0,167	0,252
CB-NTR	0,103	0,268	0,608	0,098	0,277	0,654	0,042	0,172	0,192	0,246
ACE	0,135	0,326	0,609	0,128	0,328	0,657	0,099	0,184	0,216	0,303
AraBERTv1	0,096	0,222	0,552	0,110	0,283	0,597	0,105	0,113	0,147	0,274
Bi-LSTM	0,068	0,233	0,499	0,087	0,307	0,558	0,077	0,088	0,121	0,221
Среднее	**0,088**	**0,267**	**0,572**	**0,102**	**0,290**	**0,622**	**0,065**	**0,123**	**0,169**	**0,259**

Таблица 6. Скрытность 100 текстовых состязательных примеров
(диапазон от 0 до 100, где 0 — лучшее значение скрытности)
Table 6. Stealth of 100 text adversarial examples (range 0 to 100, where 0 is the best stealth value)

Тип воздействия			Скрытность	Среднее значение скрытности за тип воздействия
Модификация	при доле омографов в тексте, %	1	19	44
		5	37	
		10	76	
	при доле невидимых символов в тексте, %	1	8	25
		5	23	
		10	64	
	при доле символов переупорядочивания, %	1	13	31
		5	26	
		10	54	
Сеть генерации			24	24

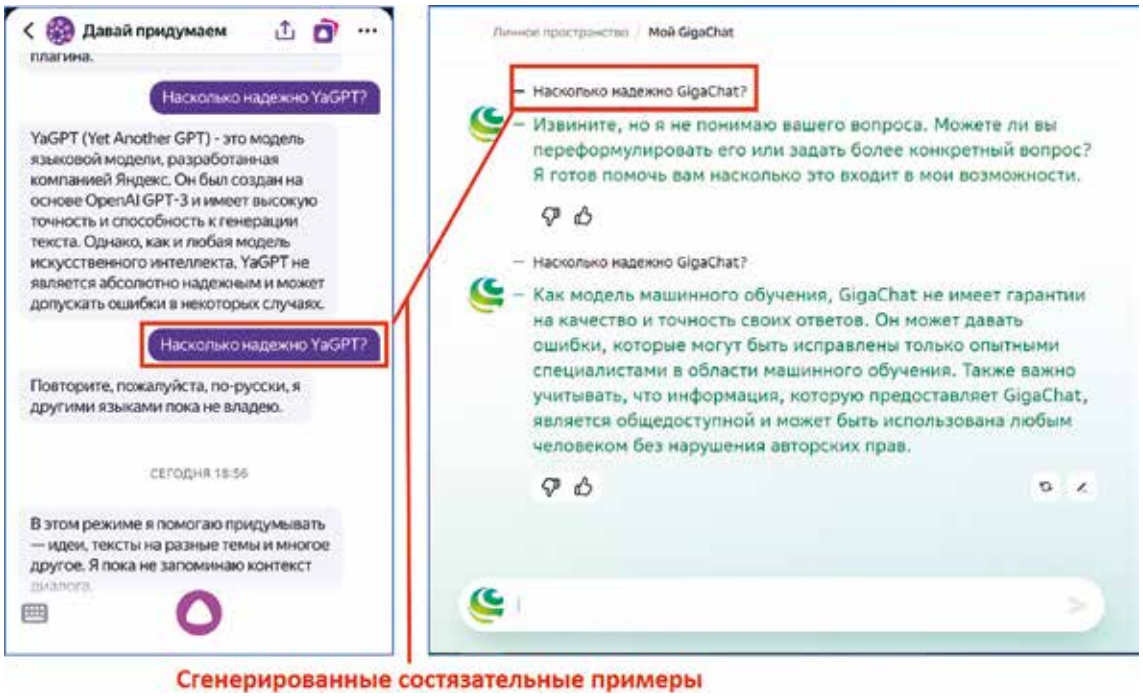

Рис. 3. Пример снижения качества результатов YaGPT и GigaChat
Fig. 3. An example of decreasing the quality of YaGPT and GigaChat results

Заключение

В работе представлен новый метод тестирования лингвистических моделей машинного обучения. Метод позволяет тестировать модели машинного обучения не только на этапе разработки, но и манипулировать результатами реальных, развернутых коммерческих систем искусственного интеллекта. При проведении экспериментальных исследований обнаружено, что тестирование успешно распространяется на разные прикладные модели машинного обучения, включая вопросно-ответные модели, рубрикации, выявления поименованных сущностей, семантического поиска и выявлении фейков.

Существуют простые средства защиты — средства орфографии и очистки ввода, которые необходимо принять как барьерный модуль для лингвистических моделей, чтобы снизить риск состязательных атак.

Перспективными направлениями работы являются следующие:

— разработка научно-методического аппарата повышения надежности функционирования моделей разных парадигм обучения (с усилением федеративного и трансферного обучения);
— разработка программных реализаций алгоритмов эксплуатации уязвимостей моделей машинного обучения от разных типов воздействия.

Литература

1. Намиот Д.Е., Ильюшин Е.А., Чижов И.В. Атаки на системы машинного обучения-общие проблемы и методы // International Journal of Open Information Technologies. 2022. Т. 10. № 3. С. 17–22.
2. Goodfellow I.J., Shlens J., Szegedy C. Explaining and harnessing adversarial examples // arXiv. 2014. arXiv:1412.6572. https://doi.org/10.48550/arXiv.1412.6572
3. Xu W., Agrawal S., Briakou E., Martindale M.J., Marine C. Understanding and detecting hallucinations in neural machine translation via model introspection // Transactions of the Association for Computational Linguistics. 2023. V. 11. P. 546–564. https://doi.org/10.1162/tacl_a_00563
4. Chang G., Gao H., Yao Z., Xiong H. TextGuise: Adaptive adversarial example attacks on text classification model // Neurocomputing. 2023. V. 529. P. 190–203. https://doi.org/10.1016/j.neucom.2023.01.071
5. Wallace E., Feng S., Kandpal N., Gardner M., Singh S. Universal adversarial triggers for attacking and analyzing NLP // Proc. of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019. P. 2153–2162. https://doi.org/10.18653/v1/d19-1221
6. Alshemali B., Kalita J. Improving the reliability of deep neural networks in NLP: A review // Knowledge-Based Systems. 2020. V. 191. P. 105210. https://doi.org/10.1016/j.knosys.2019.105210
7. Chang K.W., He H., Jia R., Singh S. Robustness and adversarial examples in natural language processing // Proc. of the 2021 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts. 2021. P. 22–26. https://doi.org/10.18653/v1/2021.emnlp-tutorials.5
8. Dong H., Dong J., Yuan S., Guan Z. Adversarial attack and defense on natural language processing in deep learning: a survey and perspective // Lecture Notes in Computer Science. 2023. V. 13655. P. 409–424. https://doi.org/10.1007/978-3-031-20096-0_31
9. Margarov G., Tomeyan G., Pereira M.J.V. Plagiarism detection system for Armenian language // Proc. of the 2017 Computer Science and Information Technologies (CSIT). 2017. P. 185–189. https://doi.org/10.1109/csitechnol.2017.8312168
10. Lupart S., Clinchant S. A study on FGSM adversarial training for neural retrieval // Lecture Notes in Computer Science. 2023. V. 13981. P. 484–492. https://doi.org/10.1007/978-3-031-28238-6_39
11. Du P., Zheng X., Liu L., Ma H. Defending against universal attack via curvature-aware category adversarial training // Proc. of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2022. P. 2470–2474. https://doi.org/10.1109/icassp43922.2022.9746983
12. Wu C., Zhang R., Guo J., De Rijke M., Fan Y., Cheng X. PRADA: Practical black-box adversarial attacks against neural ranking models // ACM Transactions on Information Systems. 2023. V. 41. N 4. P. 1–27. https://doi.org/10.1145/3576923
13. Goldblum M., Tsipras D., Xie C., Chen X., Schwarzschild A., Song D., Madry A., Li B., Goldstein T. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses // IEEE

References

1. Ilyushin E., Namiot D., Chizhov I. Attacks on machine learning systems — common problems and methods. *International Journal of Open Information Technologies*, 2022, vol. 10, no. 3, pp. 17–22. (in Russian)
2. Goodfellow I.J., Shlens J., Szegedy C. Explaining and harnessing adversarial examples. *arXiv*, 2014, arXiv:1412.6572. https://doi.org/10.48550/arXiv.1412.6572
3. Xu W., Agrawal S., Briakou E., Martindale M.J., Marine C. Understanding and detecting hallucinations in neural machine translation via model introspection. *Transactions of the Association for Computational Linguistics*, 2023, vol. 11, pp. 546–564. https://doi.org/10.1162/tacl_a_00563
4. Chang G., Gao H., Yao Z., Xiong H. TextGuise: Adaptive adversarial example attacks on text classification model. *Neurocomputing*, 2023, vol. 529, pp. 190–203. https://doi.org/10.1016/j.neucom.2023.01.071
5. Wallace E., Feng S., Kandpal N., Gardner M., Singh S. Universal adversarial triggers for attacking and analyzing NLP. *Proc. of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 2153–2162. https://doi.org/10.18653/v1/d19-1221
6. Alshemali B., Kalita J. Improving the reliability of deep neural networks in NLP: A review. *Knowledge-Based Systems*, 2020, vol. 191, pp. 105210. https://doi.org/10.1016/j.knosys.2019.105210
7. Chang K.W., He H., Jia R., Singh S. Robustness and adversarial examples in natural language processing. *Proc. of the 2021 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, 2021, pp. 22–26. https://doi.org/10.18653/v1/2021.emnlp-tutorials.5
8. Dong H., Dong J., Yuan S., Guan Z. Adversarial attack and defense on natural language processing in deep learning: a survey and perspective. *Lecture Notes in Computer Science*, 2023, vol. 13655, pp. 409–424. https://doi.org/10.1007/978-3-031-20096-0_31
9. Margarov G., Tomeyan G., Pereira M.J.V. Plagiarism detection system for Armenian language. *Proc. of the 2017 Computer Science and Information Technologies (CSIT)*, 2017, pp. 185–189. https://doi.org/10.1109/csitechnol.2017.8312168
10. Lupart S., Clinchant S. A study on FGSM adversarial training for neural retrieval. *Lecture Notes in Computer Science*, 2023, vol. 13981, pp. 484–492. https://doi.org/10.1007/978-3-031-28238-6_39
11. Du P., Zheng X., Liu L., Ma H. Defending against universal attack via curvature-aware category adversarial training. *Proc. of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 2470–2474. https://doi.org/10.1109/icassp43922.2022.9746983
12. Wu C., Zhang R., Guo J., De Rijke M., Fan Y., Cheng X. PRADA: Practical black-box adversarial attacks against neural ranking models. *ACM Transactions on Information Systems*, 2023, vol. 41, no. 4, pp. 1–27. https://doi.org/10.1145/3576923
13. Goldblum M., Tsipras D., Xie C., Chen X., Schwarzschild A., Song D., Madry A., Li B., Goldstein T. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023,

Transactions on Pattern Analysis and Machine Intelligence. 2023. V. 45. N 2. P. 1563–1580. https://doi.org/10.1109/tpami.2022.3162397
14. Ding R., Liu H., Zhou X. IE-Net: Information-enhanced binary neural networks for accurate classification // Electronics. 2022. V. 11. N 6. P. 937. https://doi.org/10.3390/electronics11060937
15. Huang Y., Giledereli B., Köksal A., Özgür A., Ozkirimli E. Balancing methods for multi-label text classification with long-tailed class distribution // Proc. of the 2021 Conference on Empirical Methods in Natural Language Processing. 2021. P. 8153–8161. https://doi.org/10.18653/v1/2021.emnlp-main.643
16. Zhang S., Yao H. ACE: An actor ensemble algorithm for continuous control with tree search // Proceedings of the AAAI Conference on Artificial Intelligence. 2019. V. 33. N 01. P. 5789–5796. https://doi.org/10.1609/aaai.v33i01.33015789
17. Antoun W., Baly F., Hajj H. AraBERT: Transformer-based model for Arabic language understanding // arXiv. 2020. arXiv:2003.00104. https://doi.org/10.48550/arXiv.2003.00104
18. Borges L., Martins B., Calado P. Combining similarity features and deep representation learning for stance detection in the context of checking fake news // Journal of Data and Information Quality (JDIQ). 2019. V. 11. N 3. P. 1–26. https://doi.org/10.1145/3287763
19. Wang X., Yang Y., Deng Y., He K. Adversarial training with fast gradient projection method against synonym substitution based text attacks // Proceedings of the AAAI Conference on Artificial Intelligence. 2021. V. 35. N 16. P. 13997–14005. https://doi.org/10.1609/aaai.v35i16.17648
20. Yang X., Qi Y., Chen H., Liu B., Liu W. Generation-based parallel particle swarm optimization for adversarial text attacks // Information Sciences. 2023. V. 644. P. 119237. https://doi.org/10.1016/j.ins.2023.119237
21. Peng H., Wang Z., Zhao D., Wu Y., Han J., Guo S., Ji S., Zhong M. Efficient text-based evolution algorithm to hard-label adversarial attacks on text // Journal of King Saud University — Computer and Information Sciences. 2023. V. 35. N 5. P. 101539. https://doi.org/10.1016/j.jksuci.2023.03.017
22. Hauser J., Meng Z., Pascual D., Wattenhofer R. Bert is robust! A case against word substitution-based adversarial attacks // Proc. of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2023. P. 1–5. https://doi.org/10.1109/icassp49357.2023.10095991

vol. 45, no. 2, pp. 1563–1580. https://doi.org/10.1109/tpami.2022.3162397
14. Ding R., Liu H., Zhou X. IE-Net: Information-enhanced binary neural networks for accurate classification. *Electronics*, 2022, vol. 11, no. 6, pp. 937. https://doi.org/10.3390/electronics11060937
15. Huang Y., Giledereli B., Köksal A., Özgür A., Ozkirimli E. Balancing methods for multi-label text classification with long-tailed class distribution. *Proc. of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 8153–8161. https://doi.org/10.18653/v1/2021.emnlp-main.643
16. Zhang S., Yao H. ACE: An actor ensemble algorithm for continuous control with tree search. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, vol. 33, no. 01, pp. 5789–5796. https://doi.org/10.1609/aaai.v33i01.33015789
17. Antoun W., Baly F., Hajj H. AraBERT: Transformer-based model for Arabic language understanding. *arXiv*, 2020, arXiv:2003.00104. https://doi.org/10.48550/arXiv.2003.00104
18. Borges L., Martins B., Calado P. Combining similarity features and deep representation learning for stance detection in the context of checking fake news. *Journal of Data and Information Quality (JDIQ)*, 2019, vol. 11, no. 3, pp. 1–26. https://doi.org/10.1145/3287763
19. Wang X., Yang Y., Deng Y., He K. Adversarial training with fast gradient projection method against synonym substitution based text attacks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, vol. 35, no. 16, pp. 13997–14005. https://doi.org/10.1609/aaai.v35i16.17648
20. Yang X., Qi Y., Chen H., Liu B., Liu W. Generation-based parallel particle swarm optimization for adversarial text attacks. *Information Sciences*, 2023, vol. 644, pp. 119237. https://doi.org/10.1016/j.ins.2023.119237
21. Peng H., Wang Z., Zhao D., Wu Y., Han J., Guo S., Ji S., Zhong M. Efficient text-based evolution algorithm to hard-label adversarial attacks on text. *Journal of King Saud University — Computer and Information Sciences*, 2023, vol. 35, no. 5, pp. 101539. https://doi.org/10.1016/j.jksuci.2023.03.017
22. Hauser J., Meng Z., Pascual D., Wattenhofer R. Bert is robust! A case against word substitution-based adversarial attacks. *Proc. of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5. https://doi.org/10.1109/icassp49357.2023.10095991

Авторы

Менисов Артем Бакытжанович — кандидат технических наук, докторант, Военно-космическая академия имени А.Ф.Можайского, Санкт-Петербург, 197198, Российская Федерация, sc 57220815185, https://orcid.org/0000-0002-9955-2694, vka@mil.ru

Ломако Александр Григорьевич — доктор технических наук, профессор, профессор, Военно-космическая академия имени А.Ф.Можайского, Санкт-Петербург, 197198, Российская Федерация, sc 57188270500, https://orcid.org/0000-0002-1764-1942, vka@mil.ru

Сабиров Тимур Римович — кандидат технических наук, старший преподаватель, Военно-космическая академия имени А.Ф.Можайского, Санкт-Петербург, 197198, Российская Федерация, sc 57188236500, https://orcid.org/0000-0002-6807-2954, vka@mil.ru

Authors

Artem B. Menisov — PhD, Doctoral Student, Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation, sc 57220815185, https://orcid.org/0000-0002-9955-2694, vka@mil.ru

Aleksandr G. Lomako — D.Sc., Full Professor, Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation, sc 57188270500, https://orcid.org/0000-0002-1764-1942, vka@mil.ru

Timur R. Sabirov — PhD, Senior Lecturer, Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation, sc 57188236500, https://orcid.org/0000-0002-6807-2954, vka@mil.ru

Статья поступила в редакцию 01.05.2023
Одобрена после рецензирования 19.06.2023
Принята к печати 17.09.2023

Received 01.05.2023
Approved after reviewing 19.06.2023
Accepted 17.09.2023