УДК 004.89

ИНСТРУМЕНТАЛЬНОЕ ОБЕСПЕЧЕНИЕ ОЦЕНКИ УСТОЙЧИВОСТИ ТРАНСФОРМЕРОВ К АТАКАМ

Артамонов Н.М., Старший оператор научной роты, III курс
Военная академия связи имени Маршала Советского Союза С. М. Буденного,
г. Санкт-Петербург

Современные трансформерные архитектуры, включая BERT, RoBERTa, GPT, T5 и их производные, стали основой почти всех флагманских решений в задачах обработки естественного языка. Их способности к обобщению, контекстуализации и многоязычной адаптации сделали их незаменимыми как в исследовательских, так и в промышленных приложениях. Однако с ростом их мощности и масштабов возросло и внимание к вопросам устойчивости: выяснилось, что трансформеры, несмотря на кажущееся понимание контекста, могут быть крайне чувствительными к небольшим изменениям входа, намеренным или случайным [1].

Это включает опечатки, перестановки слов, грамматические нарушения, но особенно – адверсариальные атаки, специально спроектированные для дестабилизации модели. В таких условиях ключевым направлением стало инструментальное обеспечение оценки устойчивости трансформеров к атакам – то есть создание, развитие и стандартизация фреймворков, библиотек, метрик и процедур, позволяющих систематически и воспроизводимо тестировать трансформерные модели на устойчивость к разнообразным формам агрессивного воздействия [2].

Проблема уязвимости трансформеров к атакам является особенно сложной, поскольку архитектурно они сильно зависят от токенизации, позиции слов, распределения внимания и конфигурации слоёв нормализации. Даже единичная замена символа или смена одного синонима может привести к радикально иному выходу модели – как в классификации, так и в генерации. При этом внутренние представления модели могут оставаться близкими, что затрудняет выявление нестабильности на уровне латентного пространства. Именно поэтому простые эвристики и ручное тестирование оказываются недостаточными: требуется инструментальная база, способная работать как в white-box режиме (с доступом к градиентам и внутренним параметрам модели), так и в black-box условиях, когда модель предоставляется в виде API без возможности внутреннего вмешательства.

Современные библиотеки, направленные на тестирование трансформеров, можно разделить на несколько классов. Наиболее зрелыми являются TextAttack и OpenAttack – это фреймворки, которые предоставляют широкий спектр методов генерации атак, от лексических (например, замена слов на синонимы, антонимы, гипонимы) до контекстуальных и градиентных (таких как HotFlip, DeepWordBug, BERT-Attack). Они позволяют моделировать атаки с разным уровнем агрессии, семантической близости, ограничения на длину, количество изменений и даже с учётом грамматической правдоподобности. Встроенные метрики – такие как успешность атаки, степень модификации, уверенность модели, изменение логитов и расстояния в эмбеддингах – позволяют объективно сравнивать поведение трансформеров разных архитектур.

Инструментальная поддержка особенно важна в задачах генерации текста, где трансформеры используются в диалоговых системах, переводе, суммаризации и создании юридически значимых текстов. Генеративные модели, такие как GPT и T5, могут быть атакованы так называемыми «prompt-injection» техниками – когда в подсказку внедряются скрытые инструкции, радикально меняющие поведение модели. Для таких случаев применяются тестовые пайплайны с автоматической генерацией атакующих промптов, оценкой контекстной устойчивости и анализом отклонений в смысловом ядре ответа. Это требует сложной инфраструктуры, включающей автоматическое сравнение с эталонными ответами, фильтрацию токсичных и несвязных высказываний, а также лингвистический анализ с привлечением предикативных шаблонов.

Кроме атак в пространстве символов и слов, актуальными становятся и атаки в эмбеддинговом пространстве – когда небольшие модификации в векторах входа приводят к значительным изменениям в выходе. Такие атаки особенно опасны в моделях, где вход преобразуется изначально в фиксированные представления (например, при использовании BERT в zero-shot классификации). Для анализа устойчивости в этих случаях применяются дифференцируемые методы, использующие обратное распространение ошибки для оптимизации вектора атаки – и требуется инструментальная поддержка градиентного анализа, автоматической генерации «направленных» искажений и построения карты устойчивости модели.

Для комплексной оценки устойчивости трансформеров важна не только генерация атак, но и интеграция тестирования в пайплайны обучения и развертывания. Здесь на помощь приходят библиотеки типа Robustness Gym, DeepChecks и NLP TestBench, которые позволяют вставлять модульные проверки устойчивости на каждом этапе: от дообучения модели на новых данных до валидации перед выводом в продакшн. Такие тесты могут быть автоматизированы и параметризированы, обеспечивая, например, оценку модели на

корпусе переформулированных запросов, на синтетически зашумлённых примерах, на межъязычных переносах или в условиях код-свитча (перемешивания языков в одном предложении) [3].

Это позволяет формировать метрики устойчивости не как разовое значение, а как динамическую характеристику модели, отслеживаемую во времени.

Особенно перспективным направлением становится формализация метрик устойчивости трансформеров. Помимо традиционных показателей, таких как accuracy under attack, всё чаще используются метрики, заимствованные из теории информации и метрической геометрии: энтропия выхода, доверительный разрыв, KL-дивергенция между распределениями логитов до и после атаки, спектральные характеристики матриц внимания. Эти показатели можно визуализировать в процессе атаки, формируя карты чувствительности, показывающие, какие слои, головы внимания или токены особенно уязвимы. Инструментальная реализация таких визуализаций требует интеграции с библиотеками типа Captum, BertViz, LIT (Language Interpretability Tool) – они позволяют разработчику не просто видеть, что модель изменила поведение, но и понять, где именно это происходит внутри архитектуры.

Кроме оценки самой модели, инструментальная база должна включать механизмы тестирования защитных механизмов. Это касается fine-tuning с аугментацией, применения робастных потерь, маскировки чувствительных токенов, предобучения на шумных данных. Только наличие инструментов, способных сравнивать поведение защищённой и незащищённой модели в идентичных условиях, позволяет делать выводы об эффективности применённых техник. Фреймворки типа TextFlint позволяют реализовать такие сценарии, вводя согласованные серии мутаций и фиксируя реакцию нескольких моделей на одинаковых примерах. Это делает возможным построение робастностных бенчмарков, сопоставимых по строгости с GLUE, SuperGLUE или XTREME, но ориентированных не на точность, а на устойчивость [4].

Наконец, важной составляющей инструментального обеспечения устойчивости трансформеров становится соблюдение нормативных требований и прозрачность анализа. В условиях появления регулирования ИИ, включая европейский AI Act и инициативы в области цифровой ответственности, важно не только протестировать модель, но и создать отчёт, пригодный для аудита. Это требует генерации формализованных отчётов с указанием типа атак, сценариев, метрик, предельных параметров, оценки повторяемости и интерпретации результатов. Такие отчёты могут быть интегрированы в систему MLflow, Data Version Control или другие компоненты MLOps, создавая полноценный модуль доверенного тестирования [5].

Таким образом, инструментальное обеспечение оценки устойчивости трансформеров к атакам является сложным, многоуровневым и быстро развивающимся направлением, в котором переплетаются инженерия, теория информации, вычислительная лингвистика и нормативная совместимость [6].

Оно требует не только знания уязвимостей архитектур, но и создания масштабируемых, стандартизированных и воспроизводимых процедур тестирования, способных выявить слабые места модели до её вывода в продуктивную среду. Только так можно обеспечить действительно надёжную, понятную и этически допустимую эксплуатацию трансформерных моделей в условиях открытой, нестабильной и потенциально враждебной информационной среды.

Список литературы:

1. Hsieh, Y.-L. Robustness Analysis of BERT-based Question Answering Models to Adversarial Paragraphs / Y.-L. Hsieh, M. Cheng, D. Juan, W. Wei, M. Hsieh, C.-J. Hsieh // Proceedings of the 2nd Workshop on Machine Reading for Question Answering. – Stroudsburg : ACL, 2019. – P. 299-304. – DOI 10.18653/v1/D19-5824.

2. Pruthi, D. Combating Adversarial Misspellings with Robust Word Recognition / D. Pruthi, B. Dhingra, Z. C. Lipton // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. – Stroudsburg : ACL, 2019. – P. 5582-5591. – DOI 10.18653/v1/P19-1561.

3. Wang, A. Dynabench : Rethinking Benchmarking in NLP / A. Wang, M. Kiela, D. Goyal [et al.] // Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics. – Stroudsburg : ACL, 2021. – P. 4110-4124. – DOI 10.18653/v1/2021.naacl-main.324.

4. Mao, Y. Robustness Gym : Unifying the NLP Evaluation Landscape / Y. Mao, L. Mathias, R. Hou [et al.] // Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics. – Stroudsburg : ACL, 2021. – P. 4768-4781. – DOI 10.18653/v1/2021.naacl-main.368.

5. Nie, Y. Adversarial NLI : A New Benchmark for Natural Language Understanding / Y. Nie, A. Williams, E. Dinan [et al.] // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. – Stroudsburg : ACL, 2020. – P. 4885-4901. – DOI 10.18653/v1/2020.acl-main.441.

6. McCoy, R. T. Right for the Wrong Reasons : Diagnosing Syntactic Heuristics in Natural Language Inference / R. T. McCoy, E. Pavlick, T. Linzen // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. – Stroudsburg : ACL, 2019. – P. 3428-3448. – DOI 10.18653/v1/P19-1334.