МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра теоретических основ компьютерной безопасности и криптографии

Безопасность систем, основанных на больших языковых моделях

АВТОРЕФЕРАТ
дипломной работы

студента 6 курса 631 группы
специальности 10.05.01 Компьютерная безопасность
факультета компьютерных наук и информационных технологий
Дунаева Михаил Дмитриевича

Научный руководитель
доцент _____ И. И. Слеповичев
19.01.2026 г.

Заведующий кафедрой
д. ф.-м. н., профессор _____ М. Б. Абросимов
19.01.2026 г.

Саратов 2026

ВВЕДЕНИЕ

Технологии нейронных сетей стремительно развиваются, и одним из наиболее распространённых их направлений стали большие языковые модели. Эти системы находят применение в самых разных областях обработки информации: от простых чат-ботов до интеллектуальных агентов, способных работать с внешними данными и выполнять команды в операционных системах. Однако вместе с ростом их популярности усиливаются и угрозы безопасности. Стохастическая природа нейронных сетей делает эти атаки непредсказуемыми, что существенно осложняет их обнаружение и предотвращение.

В данной работе рассматриваются основные виды угроз и рисков в системах, на основе больших языковых моделей, рассмотрены их классификация, способы построения атак и способы защиты. В практической части реализованы способы обнаружения инъекций запросов и проведена оценка их эффективности.

Дипломная работа состоит из введения, пяти разделов, заключения, списка использованных источников и одного приложения. Общий объем работы – 71 страница, из них 50 страниц – основное содержание, включая 0 рисунков и 4 таблицы, список использованных источников из 42 наименований.

КРАТКОЕ СОДЕРЖАНИЕ

Первый раздел дипломной работы содержит информацию, описывающие основные сведения о больших языковых моделях, об их архитектуре, способах применения и о том, как это влияет на их безопасность.

В первом подразделе даётся определение больших языковых моделей как нейронных сетей, связанных с обработкой текста, обученных на большом объёме данных и имеющих более миллиарда параметров. Описывается трёхблочная структура, состоящая из входного интерфейса (токенизация с последующим переводом в векторные вложения), блока трансформеров (механизм внимания вместе с полносвязной нейронной сетью) и выходного блока (преобразование в логиты, а затем в вероятностное распределение). Описываются способы токенизации и виды токенов, механизм внимания и его варианты (Multi-Head Attention, Multi-Query Attention и Grouped-Query Attention), способы позицинного кодирования (Rotary Positional Embeddings, Attention with Linear Biases), методы авторегрессии генерации текста и выбора токенов, а также процесс обучения.

Во втором подразделе описываются основные системы на основе больших языковых моделей. Рассматриваются чат-боты, с описанием способов разделения ролей, а также техник построения запросов (без примеров, с примерами, цепочка мыслей), системы на основе генерации дополненной поиском (RAG) и агентных систем, использующих внешние инструменты для работы с актуальными данными и операционной системой.

В третьем подразделе описывается, как особенности устройства больших языковых моделей и их использования создают проблемы безопасности. Описывается проблема стохастичности больших языковых моделей, приводящая к непредсказуемому поведению, проблема отсутствия разграничения данных в контексте модели, приводящая к доверию потенциально вредоносного текста, а также проблема доступа агентов к внешним инструментам.

Большие языковые модели способны эффективно генерировать осмысленный текст, что позволяет создать новые системы обработки информации. Однако особенности архитектуры, вероятностная природа, открывает пути к специфическими уязвимостям.

Во втором разделе рассматриваются основные угрозы и риски безопасности, которые возникают при работе с системами на основе больших языковых моделей. Рассматривается классификация OWASP LLM Top 10, описывающая основные уязвимости, классификация инъекций запросов, а также фреймворк NIST AI RMF вместе с профилем NIST AI 600-1, описывающие методы управления рисками для создания надёжных систем на основе языковых моделей.

В первом подразделе рассматривается классификация OWASP LLM Top 10, выделяющая следующие основные угрозы:

1. Инъекции запросов, заставляющие модель игнорировать инструкции, заданные пользователем или системой.
2. Раскрытие конфиденциальной информации, в том числе пользовательских данных или коммерческой тайны.
3. Риски цепочки поставок, возникающих из-за уязвимостей в сторонних компонентах.
4. Внедрение уязвимостей на этапе обучения или дообучения.
5. Неправильная обработка вывода, передача необработанного вывода модели в уязвимые подсистемы.
6. Предоставление модели избыточных прав для вызова инструментов без контроля человека.
7. Утечка системных инструкций, раскрывающих внутреннюю логику и фильтры системы.
8. Уязвимости векторных вложений и атаки на векторные базы данных в системах на основе генерации, дополненной поиском.
9. Генерация правдоподобной ложной информации (галлюцинации).
10. Неограниченное потребление ресурсов.

Также рассматривается классификация инъекций запросов. Описываются прямые и непрямые инъекции, а также их подклассы.

Во втором подразделе описывается фреймворк безопасности NIST AI RMF для управления рисками для систем на основе искусственного интеллекта. Рассматривается концепция «надёжного искусственного интеллекта», а также определяющие его характеристики (обоснованность и надёжность, безопасность, защищённость и отказоустойчивость, подотчётность и прозрачность, объяснимость и интерпретируемость, повышенная конфиденциальность, справедливость и управление вредоносной предвзятостью). Описываются основополагающие функции для управления рисками: GOVERN (формирование культуры управления рисками), MAP (идентификация рисков и определения их контекста), MEASURE (оценка рисков) и MANAGE (реагирование на риски). В конце раздела описывается риски, которые создают системы на основе больших языковых моделей согласно профилю NIST AI 600-1, являющемуся дополнением к NIST AI RMF для генеративного искусственного интеллекта.

Рассмотрены два основных документа, описывающих угрозы и риски систем на основе больших языковых моделей. Классификация OWASP LLM Top 10 описывает проблемы безопасности с технической точки зрения, в то время как фреймворк NIST AI RMF описывает риски больше с социальной точки зрения, предлагая организационно-технические методы управления, для создания надёжных систем на основе искусственного интеллекта.

В третьем разделе описываются способы построения атак на большие языковые модели, а также принципы, которые лежат в их основе.

В первом подразделе описываются атаки на этапе обучения. Сначала рассматриваются атаки, изменяющие данные, которые должны использоваться для дообучения модели. Описываются атаки BadNets - внедрение специальных примеров, на основе которых модели активируют вредоносное поведение по определённому триггеру, и Sleeper Agents, в которых роль триггера играет

контекстная переменная, позволяющая скрыть вредоносную генерацию на этапе обучения, но проявляющуюся на этапе эксплуатации.

Также в этом подразделе рассматривается атака BadEdit, которая предполагает точечное редактирование весов модели, а не тренировочного набора данных, как у прошлых атак, и атака Trojan Activation Attack, вычисляющая вектор смещения для некоторого слоя модели, для внедрения вредоносного поведения.

Во втором подразделе рассматриваются атаки на этапе логического вывода, а именно автоматическое построение инъекций запросов. Описывается GCG (градиентый метод поиска), генерирующий подстановочный суффикс, заставляющий модель менять своё поведение при добавлении его к запросу. Также описаны атаки PAIR и TAP, использующие дополнительные модели, автоматической генерации атак и тестирования их эффективности. TAP развивает идею PAIR с помощью формирования дерева атак, для отсеивания неэффективных путей построения запросов.

В третьем подразделе рассматриваются атаки на конфиденциальность. Описывается идея различного поведения модели при использовании запросов из тренировочного набора данных. Рассматривается способ построения атаки на членство в общем виде, а также её применимость в контексте больших языковых моделей.

Таким образом, показаны основные атаки на большие языковые модели. Уязвимости могут быть использованы на разных этапах жизненного цикла модели, а также их построения может быть автоматизировано, используя при этом особенности модели, как математической функции, так и в представлении «чёрного ящика», показывая при этом высокую эффективность.

В четвёртом разделе рассматриваются способы защиты больших языковых моделей от уязвимостей и атак.

В первом подразделе рассматривается защита на этапе исполнения. Описаны способы предобработки запросов: ретокенизации, перефразирования и использования специальных меток для разграничения недоверенных данных.

Также описан метод SmoothLLM, защищающий с помощью использования множества небольших искажений оригинального запроса.

Во втором подразделе описаны методы защиты во время обучения модели. Рассмотрены метод R2D2 для генерации примеров с подстановочными суффиксами и дообучения модели для создания «иммунитета» к ним, метод CAT, аналогичный предыдущему, но использующий вместо дискретных примеров, непрерывные векторные вложения, для повышения вычислительной эффективности. Также рассмотрен метод Safe RLHF, модифицирующий обучение с подкреплением, путём внесения в него оценки безвредности помимо полезности.

В третьем подразделе описаны способы защиты на основе систем контроля и защитных барьеров. Описан архитектурный подход к построению таких систем с помощью добавления слоёв защиты семантической фильтрации входов, динамической проверки вывода и изоляции среды выполнения внешних инструментов. Также описаны специализированные модели и принцип минимизации прав.

Таким образом, существуют различные способы защиты систем на основе больших языковых моделей, причём они есть для каждого этапа жизненного цикла моделей. В то же время, не смотря на всё многообразие, даже комплексного применения всех методов может быть недостаточно и необходимо придерживаться архитектурных решений с минимизацией прав доступа для обеспечения максимальной защищённости.

В пятом разделе описывается практическая реализация библиотеки для обнаружения инъекций запросов. Описываются способы обнаружения, алгоритмы, а также особенности реализации. Для каждого метода проведено тестирование и оценка эффективности.

В первом подразделе описаны способы обнаружения. Первым описан детектор на основе перплексии — меры неопределённости текста. Рассматривались как классификация на основе порогового значения, так и в сочетании с количеством токенов в запросе.

Следующим описан метод на основе классификации векторных вложений. Его идея основа на использование векторных вложений, полученных с помощью таких моделей как BERT или DeBERTa. Предполагается, что вредоносные запросы имеют схожие семантические признаки, их векторы будут находиться на близком друг от друга расстоянии. Тогда их можно классифицировать с помощью классических методов машинного обучения.

Третий способ обнаружения основан на классификации с использованием больших языковых моделей, поскольку благодаря своей архитектуре они способны воспринимать смысл запроса и тем самым распознать его вредоносные намерения, если они есть, следуя заранее определённым правилам. Рассмотрена также проблема уязвимости классификатора к инъекциям запросов, а также возможное решение.

Четвёртым рассмотрен способ обнаружения на основе отслеживания изменения активации слоёв внимания. Идея основана том, что вредоносные запросы, смещают внимание модели с оригинальной инструкции в некоторых «головах» и слоях внимания. Для этого эффекта описана оценка, которая позволяет определить его наличие, при обработке моделью запроса.

Во втором подразделе описаны особенности практической реализации. Обозначена общая архитектура библиотеки, а также описано устройство детектора, определяющего наличие инъекции запроса в тексте. Также описаны используемые библиотеки, и методы классификации для детекторов на основе перплексии и векторных вложений.

В третьем подразделе описан процесс тестирования. Обозначен набор данных для тестирования и причины его использования. Описана среда, в которой производилось тестирование. Описаны конкретные параметры детекторов, в частности языковые модели, используемые в детекторах на основе наивной классификации и механизма внимания.

В четвёртом подразделе описаны метрики, используемые для оценки детекторов: Accuracy (доля правильных ответов), Precision (точность), Recall (полнота) и F-мера.

В пятом подразделе описаны результаты тестирования детекторов с параметрами и данными указанными ранее. Были получены следующие результаты:

1. PPL-детекторы: Наивный пороговый метод оказался бесполезным, в то время как комбинация перплексии и длины токенов вместе с классификатором на основе машинного обучения, в частности, на основе градиентного бустинга.
2. Детекторы на основе векторных вложений показали наилучший результат. Все варианты показали высокую долю правильных ответов и точность на одинаковом уровне.
3. При наивной классификации получились нестабильные результаты. Самая маленькая модель показала худший результат, в то время как модели с большим числом параметров проявили себя лучше.
4. Отслеживание на основе внимания показало низкий результат сравнимый с методом перплексии. При этом модель с самым большим числом параметров оказалась бесполезной и не наблюдается зависимости между числом параметров модели и её метриками.

Тестирование показало, что детекторы на основе векторных вложений являются наиболее эффективными, в то время как другие значительно менее эффективны, но имеют свои особенности и могут быть использованы эффективно при правильной настройке в специфических случаях.

ЗАКЛЮЧЕНИЕ

В ходе данной работы был проведён комплексный анализ безопасности систем на основе больших языковых моделей, рассмотрены их теоретические основы, виды угроз и рисков, способы атак, и способы защиты от них. Архитектура трансформеров, несмотря на свою эффективность в обработке естественного языка, имеет стохастическую природу, что и определяет специфику проблем с безопасностью.

В практической части работы были реализованы и протестированы основные методы детектирования атак: на основе перплексии, векторных вложений, отслеживания внимания и использования LLM в качестве классификатора. На основании проведённых экспериментов можно сделать следующие выводы:

1. Наибольшую эффективность продемонстрировал метод на основе векторных вложений в сочетании с классификатором градиентного бустинга. Данный метод показал наиболее сбалансированные результаты, обеспечивая высокую точность при минимальном количестве ложных срабатываний. Это подтверждает гипотезу о том, что семантическое представление текста позволяет лучше улавливать структуру вредоносных запросов.

2. Метод на основе перплексии с фиксированным порогом оказался бесполезен для практического применения из-за критически низкого показателя полноты. Однако использование перплексии вместе с количеством токенов в запросе позволяет классифицировать запросы используя методы машинного обучения в качестве легковесного средства обнаружения инъекций запросов.

3. Использование больших языковых моделей с наивной классификацией показало высокую чувствительность к атакам, но столкнулось с проблемой «гиперчувствительности» и высокого уровня ложных тревог. Кроме того для более эффективных результатов требуется модель с большим числом параметров.

4. Анализ механизмов внимания подтвердил возможность обнаружения инъекций запросов, однако точность метода не позволяет его рекомендовать в качестве способа обнаружения. Использование детектора на основе векторных вложений даёт более высокий результат, при меньших вычислительных затратах.