КОМПЬЮТЕРНЫЕ НАУКИ, ПРИБОСТРОЕНИЕ И АВТОМАТИЗАЦИЯ
КОМПЬЮТЕРЛІК ҒЫЛЫМДАР, АСПАП ЖАСАУ ЖӘНЕ АВТОМАТТАНДЫРУ
COMPUTER SCIENCE, INSTRUMENTATION AND AUTOMATION

УДК 004.056.5
МРНТИ 81.93.29
https://doi.org/10.53364/24138614_2025_38_3_10

С.А.Адилжанова[1], А.Н.Құрасбек[1]*, М.О.Кенжебаева[1]
[1]Казахский национальный университет имени аль-Фараби, Алматы, Казахстан

*E-mail: kurasbek.a.140@mail.ru

ПРИМЕНЕНИЕ LLM В КИБЕРБЕЗОПАСНОСТИ: ОБЗОР ПРИЛОЖЕНИЙ И УЯЗВИМОСТЕЙ LLM

Аннотация. *В этом документе представлен всесторонний обзор будущего кибербезопасности с помощью больших языковых моделей (LLM). Мы представляем обзор эволюции LLM и ее текущего состояния, уделяя особое внимание достижениям в таких моделях, как GPT-4, GPT-3.5, BERT, Falcon2 и LLaMA. Наш анализ распространяется на уязвимости LLM, такие как быстрое внедрение, небезопасная обработка выходных данных, отравление данных, DDoS-атаки и состязательные инструкции. Мы подробно рассмотрим стратегии смягчения последствий для защиты этих моделей, предоставив всесторонний обзор потенциальных сценариев атак и методов их предотвращения. Эти аналитические данные направлены на улучшение защиты от кибербезопасности в режиме реального времени и повышение сложности приложений LLM для обнаружения угроз и реагирования на них. В нашем документе представлено основополагающее понимание и стратегическое направление интеграции LLM в будущие системы кибербезопасности для защиты от развивающихся киберугроз.*

Ключевые слова: *LLM, кибербезопасность, большие языковые модели, языковое моделирование, машинное обучение, NLP, обработка естественного языка.*

Введение.

Кибербезопасность — это постоянно развивающаяся область, в которой угрозы становятся все более изощренными и сложными. Поскольку организации и частные лица полагаются на цифровые технологии для связи, коммерции и критически важной инфраструктуры, потребность в надежных мерах к 65432ибербезопасности никогда не была столь велика [1]. Масштаб и разнообразие киберугроз ставят перед специалистами по безопасности сложную задачу по их эффективному выявлению, обнаружению и защите от них. В этом контексте большие языковые модели (LLM) стали технологией, меняющей правила игры и способной значительно улучшить методы кибербезопасности [2, 3, 4, 5, 6]. Эти модели, основанные на передовых методах NLP и машинного обучения (ML), открывают новый рубеж в борьбе с киберугрозами [7, 8]. В этой статье исследуются мотивы и применение LLM в кибербезопасности.

Специалистам по кибербезопасности часто приходится просеивать огромное количество текстовых данных, включая предупреждения безопасности, отчеты об инцидентах, ленты угроз и исследовательские работы, чтобы опережать развивающиеся угрозы. LLM, такие как Falcon [9], обладают возможностями распознавания естественного языка, которые позволяют им эффективно анализировать, обобщать и контекстуализировать эту информацию [10, 11], быстро выявлять релевантную информацию об угрозах, что позволяет аналитикам принимать более обоснованные

решения и расставлять приоритеты в ответных мерах [12]. LLM могут преуспеть в различных областях кибербезопасности [13, 14].

Материалы и методы исследования.

В таблице сравниваются общие большие языковые модели, основанные на преобразованиях. Модели LLM обычно обучаются на разнообразных и широких данных, чтобы обеспечить относительно полное понимание. Они могут выполнять различные языковые задачи, такие как перевод, обобщение и ответы на вопросы. В отличие от них, LLM, специфичные для кода, представляют собой специализированные модели, обученные в основном на языках программирования и соответствующей технической литературе, что делает их основную роль в понимании и создании программного кода хорошо подходящей для таких задач, как автоматическая генерация кода, завершение кода и обнаружение ошибок.

Таблица 1 - Сравнение больших языковых моделей

Модель	**Предварительная подготовка**	**Приложений**	**Примеры использования в кибербезопасности**	**Схема обучения**	**Основные техники обучения**
GPT-3	Книги, Веб-текст, Википедия, Общий обход	Языковое моделирование, завершение текстов, контроль качества	Обнаружение вредоносных программ, аналитика угроз, обнаружение социальной инженерии	Предварительное обучение, Контекстное обучение	Авторегрессионное обучение, Масштабируемые потери перекрестной энтропии, Обратное распространение ошибки и градиентный спуск, Обучение смешанной точности.
GPT-4	Веб-данные, Лицензированные данные третьих лиц	Языковое моделирование, завершение текстов, контроль качества	Обнаружение вредоносных программ, аналитика угроз, обнаружение социальной инженерии	Предтренировочная подготовка, РЛХФ	Авторегрессионное обучение
T5	C4, Веб-текст, Википедия	Языковое моделирование, обобщение, перевод	Обнаружение вредоносных программ, аналитика угроз, обнаружение социальной инженерии	Предобучение, Тонкая настройка	Структура преобразования текста в текст, предварительное обучение на основе денотации
BERT	BooksCorpus, Английская Википедия	Языковое моделирование, классификация, QA, NER	Обнаружение вредоносных программ, аналитика угроз, обнаружение вторжений,	Предварительная подготовка	Маскированный LM(MLM), прогноз следующего предложения (NSP)

			обнаружение фишинга		
ALBERT	BooksCorpus, Английская Википедия	Языковое моделирование, классификация	Обнаружение вредоносных программ, аналитика угроз, обнаружение вторжений, обнаружение фишинга	Предварительная подготовка	Факторизованная параметризация встраивания, совместное использование межуровневых параметров, потеря связности между предложениями, прогнозирование порядка предложений (СОП)
RoBERTa	BooksCorpus, Английская Википедия	Языковое моделирование, классификация, QA, NER	Обнаружение вредоносных программ, аналитика угроз, обнаружение вторжений, обнаружение фишинга	Предварительная подготовка	Динамическое маскирование, полные предложения без потери NSP, большие мини-пакеты, большие BPE на уровне байтов
XLNet	Английская Википедия	Языковое моделирование, классификация, контроль качества	Обнаружение вредоносных программ, аналитика угроз, обнаружение вторжений, обнаружение фишинга	Предварительная подготовка	Перестановка LM(PLM), двухпотоковое самовнимание, рекуррентность сегментов и относительное кодирование
ProphetNet	Веб-данные, Книги	Языковое моделирование, генерация вопросов, обобщение	Отчетность по кибербезопасности, аналитика угроз	Предобучение, Тонкая настройка	Генерация маскированной последовательности, Авторегрессионное обучение, Цель автоэнкодера с шумоподавлением, Общие параметры между энкодером и декодером, Оценка максимального

					правдоподобия (MLE)
Falcon	Веб-данные	Языковое моделирование, завершение текстов, контроль качества	Обнаружение вредоносных программ, аналитика угроз, обнаружение социальной инженерии	Предварительная подготовка	Авторегрессионное обучение, FlashAttention, позиционное кодирование ALiBi
Reformer	Веб-данные	Языковое моделирование, классификация	Обнаружение вредоносных программ, аналитика угроз, обнаружение вторжений, обнаружение фишинга	Предварительная подготовка	Локально-чувствительное хеширование (LSH) внимание, фрагментированная обработка, головки внимания Shared-QK, обратимые слои
PaLM	Веб-страницы, Википедия, новостные статьи, книги, исходный код, обсуждения в социальных сетях, GitHub	Языковое моделирование, контроль качества, перевод	Аналитика угроз, создание политик безопасности	Предварительная подготовка	Активация SwiGLU, параллельные уровни, многозапросное внимание (MQA), встраивание RoPE, встраивание общих входов-выходов
PaLM2	Веб-документы, книги, код, математика, разговорные данные	Языковое моделирование, QA, Суммаризация	Аналитика угроз, создание политик безопасности	Предварительная подготовка	Вычисление оптимального масштабирования, последовательности маркеров Canary, маркеры управления для вывода
LLaMA	CommonCrawl, C4, GitHub, Википедия, Книги, arXiv, StackExchange	Языковое моделирование, завершение текстов, контроль качества	Аналитика угроз, обнаружение вредоносного ПО	Предварительная подготовка	Предварительная нормализация, функция активации SwiGLU, ротационное встраивание, моделирование и параллелизм

					последовательностей
LLaMA2	Сочетание пулидально доступных данных	Языковое моделирование, завершение текстов, контроль качества	Аналитика угроз, обнаружение вредоносного ПО	Предтренировочный, Тонкая настройка, РЛХФ	Оптимизированное авторегрессионное обучение, сгруппированное внимание к запросу (GQA)
Yi-34B	Набор данных на китайском и английском языках	Языковое моделирование, ответы на вопросы	Аналитика угроз, обнаружение фишинга, оценка уязвимостей	Предобучение, Тонкая настройка	NA

Круговая диаграмма отражает распределение языковых моделей по типам архитектуры: Decoder-only, Encoder-only, Encoder-decoder и NA (для моделей с отсутствующей информацией о типе архитектуры, как Yi-34B). Диаграмма наглядно показывает преобладание моделей с архитектурой Decoder-only, что свидетельствует о популярности этого подхода в разработке современных LLM. Меньшую долю занимают модели Encoder-only и Encoder-decoder, каждая из которых применяется для решения специфических задач в области обработки естественного языка.

Диаграмма 1 – Доли моделей по типам архитектур

Диаграмма рассеяния демонстрирует объёмы корпусов различных языковых моделей. По оси X представлены названия моделей, по оси Y — объём используемых для обучения данных. Модели, для которых объём корпуса известен, отмечены синими маркерами, тогда как модели с отсутствующей информацией (NA) выделены красными маркерами. Эта визуализация наглядно показывает значительные различия в масштабах данных, используемых при обучении, и подчёркивает пробелы в доступной информации по некоторым моделям.

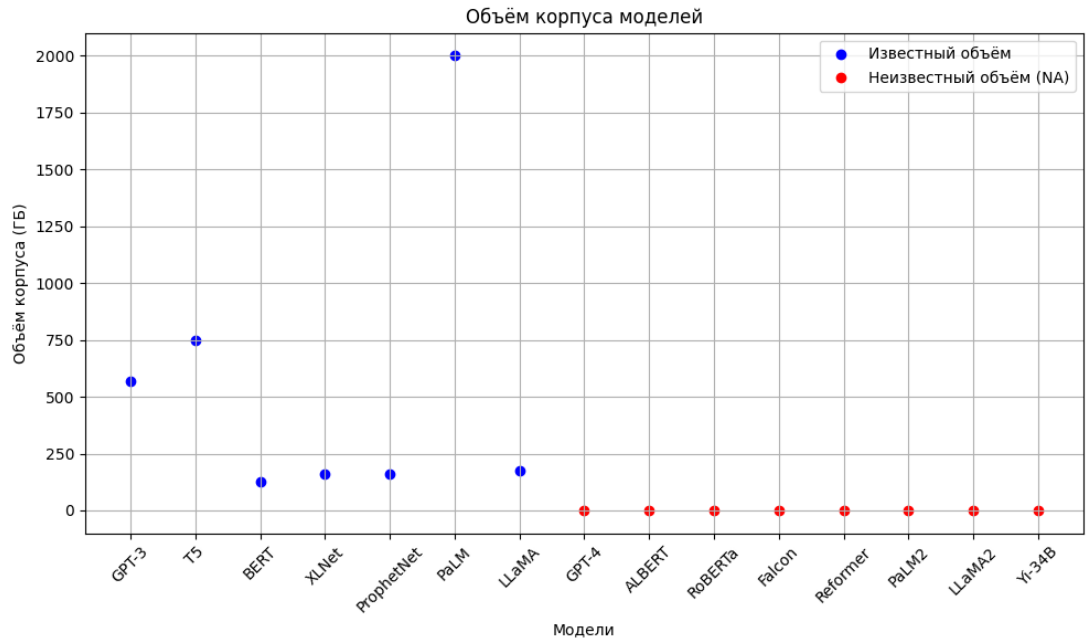

Диаграмма 2 – Объём корпуса моделей

GPT-3 (третья версия серии Generative Pre-trained Transformer от OpenAI) был разработан, чтобы доказать, что масштабирование языковых моделей существенно улучшает их производительность, не зависящую от задачи, [15]. Основанный на архитектуре трансформатора, GPT-3 имеет восемь вариантов параметров от 125M до 175B, все они обучены для токенов 300B из таких наборов данных, как Common Crawl, WebText, Books и Wikipedia. Кроме того, модели были обучены на графическом процессоре V100 с использованием таких методов, как авторегрессионное обучение, масштабируемые потери кросс-энтропии и другие. GPT-3, особенно его наиболее мощная версия 175B, продемонстрировал высокую производительность во многих задачах NLP в различных условиях (т.е. нулевой выстрел, один выстрел и несколько выстрелов), что позволяет предположить, что он может значительно улучшить приложения кибербезопасности при соответствующей тонкой настройке. Это может привести к более эффективному обнаружению фишинга за счет точного анализа языка, более быстрого реагирования на инциденты и других критически важных приложений для усиления мер цифровой безопасности.

В 2023 году OpenAI выпустила модель на основе трансформатора GPT-4 в качестве первой крупномасштабной мультимодальной модели, продемонстрировавшей беспрецедентную производительность в различных бенчмарках. Способность модели обрабатывать входные данные изображений и текста вывела парадигму искусственного интеллекта на новый уровень, выйдя за рамки традиционного NLP. [16] заявил, что GPT-4 был обучен с использованием обширного корпуса веб-данных и данных, лицензированных из сторонних источников с помощью методов авторегрессии и обучения с подкреплением на основе обратной связи человека (RLHF). Однако другие подробности, такие как размер модели, объем данных и подробные сведения об обучении, остаются нераскрытыми. Несмотря на то, что GPT-4 потенциально может быть использован киберпреступниками для широкого спектра атак, таких как социальная инженерия, при стратегическом применении он также может помочь снизить вероятность того, что отдельные лица и организации станут их жертвами.

Вдохновленные тенденцией применения трансферного обучения для NLP, исследователи Google представили T5 [17], модель на основе кодера-декодера, которая работает в рамках единой структуры text-to-text. Несколько вариантов T5 с различными

размерами — от 220M до 11B параметров — были разработаны для расширения экспериментальной области и обучены на огромных объемах данных из различных источников, включая C4, Web Text и Wikipedia. Основываясь на этих различных размерах моделей и богатых источниках данных, были изучены и обсуждены различные подходы и различные настройки для предварительного обучения и тонкой настройки, в результате чего была достигнута производительность, которая почти соответствовала человеческому уровню в одном из контрольных показателей. Учитывая это, потенциал модели в приложениях кибербезопасности особенно многообещающий. Например, T5 можно использовать для анализа угроз, извлекая критически важную информацию из обширных документов по безопасности, а затем обобщая и систематизировав эту информацию.

Книга «Bidirectional Encoder Representations from Transformers», широко известная как BERT, была представлена в [18] для улучшения подходов, основанных на тонкой настройке в NLP. Он доступен в двух версиях: BERT-Base, с 110 млн параметров, и BERT-Large, с 340 млн параметров, обученный на 126 ГБ данных из BooksCorpus и английской Википедии. На этапе предварительного обучения BERT использовал два ключевых метода: моделирование замаскированного языка (MLM) и прогнозирование следующего предложения (NSP). Основываясь на этих подходах, тонкой настройке и методах, основанных на функциях, в частности, BERT-Large добилась конкурентоспособных результатов. Поскольку модели, использующие только кодировщики, такие как BERT, известны своим надежным пониманием контекста, применение таких моделей для таких задач, как обнаружение вредоносных программ и уязвимостей программного обеспечения, может быть очень эффективным в области кибербезопасности.

Стремясь устранить ограничения, связанные с памятью GPU/TPU и временем обучения в больших языковых моделях (LLM), исследователи Google разработали A Lite BERT (ALBERT), модифицированную версию BERT со значительно меньшим количеством параметров [19]. И, как и другие LLM, ALBERT был представлен в различных размерах, с параметрами от 12 до 235 миллионов, и все они обучались на данных из BooksCorpus и английской Википедии. На этапе предварительного обучения были развернуты различные методы и техники, в том числе параметризация факторизованного встраивания, межуровневое совместное использование параметров, потеря связности между предложениями и прогнозирование порядка предложений (SOP). В результате, одна из моделей (т.е. ALBERT-xxlarge) превзошла BERT-Large, несмотря на меньшее количество параметров. Таким образом, использование ALBERT в приложениях кибербезопасности, таких как обнаружение фишинга и классификация вредоносных программ, может внести значительный вклад в развитие инфраструктуры кибербезопасности.

RoBERTa, предложенный Meta, представляет собой оптимизированную репликацию BERT, которая демонстрирует, как выбор гиперпараметров может существенно повлиять на производительность модели [20]. RoBERTa имеет только одну версию с 355 млн параметров, но она обучена и протестирована в различных объемах данных и на различных этапах обучения. Как и в случае с BERT, обучающие данные были взяты из Книжного корпуса и английской Википедии. Тем не менее, ключевые оптимизации в этой модели заключались в методах обучения, которые включали в себя несколько методов, таких как динамическая маскировка, обучение на полных предложениях без потерь NSP, использование больших мини-пакетов и использование BPE большего байтового уровня. Таким образом, RoBERTa достигла самых современных результатов в некоторых бенчмарках. При правильной тонкой настройке способность RoBERTa понимать, интерпретировать и генерировать текст, похожий на человеческий, используется для автоматизации и улучшения различных задач в области кибербезопасности.

Достижения и ограничения Masked Language Modeling (MLM) в двунаправленных кодировщиках и авторегрессионном моделировании языка вдохновили исследователей из CMU и Google AI на разработку XLNet [21]. Основанная на модели Transformer-XL, XLNet

сочетает в себе аспекты обоих подходов, позволяя изучать двунаправленные контексты и решая общие проблемы MLM, такие как пренебрежение зависимостями между замаскированными позициями и несоответствие между фазами предварительного обучения и тонкой настройки. С 340 млн параметров XLNet был предварительно обучен с использованием данных из английской Википедии и с использованием таких методов, как моделирование языка перестановок (PLM), двухпотоковое внимание, повторение сегментов и относительное кодирование. Благодаря тщательному проектированию модели и стратегическим методам предварительного обучения, XLNet достигла значительных результатов по сравнению с другими популярными моделями, такими как BERT, что делает ее — после соответствующей тонкой настройки — способным инструментом для улучшения различных аспектов области кибербезопасности.

ProphetNet LLM, предложенная Microsoft, представляет собой предварительно обученную модель от последовательности к последовательности, направленную на решение проблемы переобучения на сильных локальных корреляциях с использованием двух новых методов, а именно: предсказания будущего n-грамм и n-потокового самовнимания [22]. Построенный на архитектуре кодера-декодера и обученный на базовом масштабе 16 ГБ и больших наборах данных 160 ГБ, полученных из веб-данных и книг, ProphetNet с его параметрами 550 млн достиг новых современных результатов в нескольких тестах. Модель также была тонко настроена для двух последующих задач, Question Generation и Text Summarization, где она достигла наилучшей производительности. Таким образом, использование ProphetNet в задачах кибербезопасности, таких как автоматизированное обобщение инцидентов безопасности, может значительно повысить эффективность и эффективность принятия решений.

Falcon LLM, построенный на архитектуре, основанной только на декодере, был представлен Институтом технологических инноваций (TII) в качестве доказательства концепции, согласно которой повышение качества данных может значительно улучшить производительность LLM даже при использовании данных, полученных исключительно из веб-источников [23]. Это понимание становится все более актуальным, поскольку масштабирование в LLM, которое становится все более распространенным, требует больше данных для обработки. Модель имеет три версии (т.е. 7B, 40B, 180B), предварительно обученные на наборе данных "RefinedWeb", предложенном TII. Компания RefinedWeb, полученная исключительно из веб-данных, была подвергнута различным методам фильтрации и дедупликации для обеспечения высокого качества. Авторегрессионное обучение, Flash Attention и позиционное кодирование ALiBi были методами, используемыми для предварительного обучения. Благодаря дальнейшей тонкой настройке Falcon может улучшить кибербезопасность, особенно в области разведки и анализа угроз.

Стремясь устранить распространенные ограничения памяти в LLM, Google предложила Reformer, кодировщик-декодер, эффективный в памяти LLM [24]. Имея до 6 Б параметров, Reformer был предварительно обучен на веб-данных с использованием таких методов, как Locality-Sensitive Hashing (LSH) Attention, Chunked Processing, Shared-QK Attention Heads и Reversible layers. Было доказано, что эти методы оказывают незначительное влияние на процесс обучения по сравнению со стандартным Transformer, так как Reformer достиг результатов, которые соответствовали полному Transformer, но с гораздо более быстрой обработкой и лучшей эффективностью памяти. Впоследствии, использование Reformer для таких задач, как крупномасштабный анализ данных, может послужить области кибербезопасности, обеспечивая более эффективную обработку и анализ обширных наборов данных.

Движимая развитием машинного обучения и обработки естественного языка, компания Google разработала PaLM для изучения влияния масштаба на малосерийное обучение [25]. PaLM, построенный на архитектуре только с декодером, был обучен с параметрами 540B с использованием Pathways, новой системы, которая использует

высокоэффективное обучение на нескольких модулях TPU. Модель была обучена на 2 ТБ данных из нескольких источников, включая новостные статьи, Википедию, исходный код и т. д. SwiGLU Activation, Parallel Layers и другие методы были развернуты для предварительного обучения трех различных шкал параметров: 8B, 62B и 540B, чтобы лучше понять поведение масштабирования. Наблюдаемое прерывистое улучшение показало, что по мере того, как LLM достигают определенного уровня масштаба, они проявляют новые способности. Кроме того, эти новые возможности продолжают развиваться и становятся очевидными даже за пределами масштабов, которые были ранее исследованы и задокументированы. Впоследствии PaLM совершил прорыв, превзойдя точно настроенного современного и среднего человека в некоторых тестах, доказав, что, когда масштабирование сочетается с цепочкой подсказок, базовая оценка нескольких выстрелов может сравняться или превзойти производительность тонко настроенных современных моделей в широком спектре задач рассуждения. Обладая такими широкими возможностями, использование PaLM для таких задач, как создание политик безопасности и автоматизация реагирования на инциденты, может повысить эффективность и результативность операций по обеспечению кибербезопасности.

PaLM2 является усовершенствованным вариантом модели PaLM, который является более эффективным с точки зрения вычислений, хотя и предлагает лучшие многоязычные и интеллектуальные возможности [26]. Основными улучшениями в модели являются улучшенные наборы данных, оптимальное масштабирование для вычислений, а также архитектурные и объективные улучшения. Важные результаты оценки PaLM2 указывают на то, что помимо масштабирования можно было бы разработать различные подходы к усовершенствованию модели, такие как тщательный отбор данных и эффективная архитектура/цели. Более того, тот факт, что PaLM2 превзошел предшественника PaLM, несмотря на его значительно меньший размер, показывает, что качество модели оказывает большее влияние на производительность, чем размер модели, поскольку это может обеспечить более эффективный вывод, снижая затраты на обслуживание и потенциально обеспечивая более широкие приложения и доступность для большего числа пользователей.

Предложенная Meta модель LLaMA, состоящая только из декодеров, является доказательством того, что можно достичь современной производительности, обучаясь исключительно на общедоступных данных [27]. LLaMA с несколькими вариантами от 7 до 65 миллиардов параметров была обучена на 1400 млрд токенов общедоступных наборов данных, включая CommonCrawl, C4, arXiv и другие. Интересно, что методы, использованные для обучения модели, были вдохновлены несколькими популярными моделями, такими как GPT-3 (предварительная нормализация), PaLM (функция активации SwiGLU) и GPTNeo (Rotary Embedding). В результате этого включения LLaMA-13B смогла превзойти GPT-3 (175B) в большинстве тестов, несмотря на то, что она была более чем в десять раз меньше, в то время как LLaMA-65B показала конкурентоспособность с Chinchilla-70B и PaLM-540B. Учитывая его относительно небольшой размер и превосходную производительность, тонкая настройка LLaMA для задач анализа киберугроз может значительно повысить безопасность периферийных устройств.

LLaMA2 — это оптимизированная версия LLaMA, разработанная компанией Meta, и набор предварительно обученных и точно настроенных LLM с размерами от 7 до 70B параметров [28]. В ходе предварительного обучения была использована смесь общедоступных данных для получения до 2000 млрд обучающих токенов. Кроме того, в предшественнике LLaMA использовалось несколько методов, таких как предварительная нормализация, функция активации SwiGLU и ротационные позиционные встраивания. Также были использованы два дополнительных метода, а именно увеличение длины контекста и внимание группового запроса (GQA). После предварительного обучения варианты модели (например, LLaMA2-Chat) были оптимизированы для диалоговых сценариев использования с помощью контролируемой тонкой настройки и обучения с

подкреплением с обратной связью от человека (RLHF). Оценка модели, которая была сосредоточена на полезности и безопасности, показала превосходство над другими моделями с открытым исходным кодом и конкурентоспособность по сравнению с некоторыми моделями с закрытым исходным кодом.

Недавно выпущенный LLM Yi-34B, разработанный 01.AI, привлекает внимание как один из лучших LLM с открытым исходным кодом [29]. Учитывая недавний выпуск модели, ее технический документ еще не опубликован; следовательно, доступная информация ограничена. Модель имеет несколько вариантов: базовая и чатовая модели, некоторые квантованы. Все варианты обучаются на наборе данных, содержащем только китайский и английский языки, а версии чата прошли контролируемую тонкую настройку, что привело к более эффективным моделям для последующих задач. Базовая модель превзошла многие открытые LLM в определенных бенчмарках, включая такие известные, как LLaMA2-70B и Falcon-180B. Даже квантованные версии продемонстрировали впечатляющую производительность, что открывает путь к их развертыванию в приложениях кибербезопасности, таких как решения для безопасности на периферии.

Результаты и их обсуждение.

В этом разделе рассматривается проект OWASP Top 10 for LLM Applications [30], комплексная инициатива, направленная на повышение осведомленности об уязвимостях безопасности LLM. Этот проект нацелен на широкую аудиторию, включая разработчиков, дизайнеров, архитекторов, менеджеров и организации, которые развертывают и управляют LLM. В его основном продукте перечислены 10 наиболее критических уязвимостей безопасности, обычно встречающихся в приложениях LLM. Кроме того, мы включили другие уязвимости LLM, не включенные в проект OWASP, как представлено в таблице.

Таблица 2 - Обзор уязвимостей LLM и их устранение

Уязвимые места	**Характер уязвимости**	**Примеры**	**Стратегии смягчения последствий**	**Потенциальные сценарии атак**
Быстрая инъекция	Манипулирование LLM с помощью сфабрикованных входных данных, приводящее к несанкционированному использованию или раскрытию конфиденциальной.	Скрытые подсказки на веб-страницах Документы, вводящие в заблуждение Инструкции по работе с мошенническим веб-плагином	• Эксплуатационные ограничения • Согласие пользователя на конфиденциальные операции • Установление доверительных границ	• Состязательные инъекции на сайтах • Скрытые подсказки в документах • Прямое управление пользователем с помощью специально разработанных входных данных
Небезопасная обработка вывода	Слепое доверие к результатам LLM приводит к рискам безопасности, таким как XSS, CSRF, SSRF и т. д.	• Прямая обработка JavaScript или Markdown, сгенерированного LLM	• Валидация результатов LLM • Кодирование выходных данных до того, как они попадут к конечным пользователям	• Ответы LLM для управления внутренними функциями • Генерация непроверенных SQL-запросов • Создание вредоносных

				полезных данных XSS
Отравление данных логического вывода	Скрытая активация вредоносных ответов в определенных рабочих условиях, таких как ограниченный по токенам вывод.	• Условия, основанные на лимитах вывода токенов в настройках пользователя • Скрытое изменение выходов при включении экономичных режимов	• Системы мониторинга и обнаружения аномалий, специально разработанные для условных выводов • Регулярный аудит выходов с различными ограничениями токенов	• Манипулирование ответами в условиях ограничений токенов, приводящее к дезинформации • Инициированное вредоносное поведение в средах, чувствительных к затратам
Состязательные инструкции на естественном языке	Программные LLM создают функционально точный код со скрытыми уязвимостями из-за состязательных инструкций.	Алгоритм DeceptPrompt, создающий вводящие в заблуждение инструкции	Расширенная валидация кода Обучение LLM на состязательных примерах Постоянные обновления и исправления безопасности	Созданные подсказки, ведущие к коду с уязвимостями Несанкциониров-ан-ный доступ или компрометация системы
Автоматическое создание состязательных запросов	Автоматизированные методы генерации подсказок в обход мер выравнивания LLM.	Создание специфических суффиксов для генерации нежелательного контента	Разработка усовершенствованных алгоритмов юстировки Мониторинг в режиме реального времени Обучающие модели с новыми состязательными примерами	Обход мер по выравниванию, приводящий к созданию нежелательного контента
Отравление тренировочных данных	Манипулирование обучающими данными для искажения обучения LLM, внесение предубеждений или уязвимостей.	• Внедрение предвзятых или вредных данных в обучающие наборы	• Проверка источников данных • Использование специализированных моделей • Песочница, входные фильтры • Контроль за признаками отравления	• Вводящие в заблуждение выводы, распространяющие предвзятые мнения • Внедрение ложных данных в обучение
Небезопасные плагины	Уязвимости в дизайне плагинов	• Недостаточная валидация	• Строгая проверка	• Эксплуатация уязвимостей

	и взаимодействии с внешними системами или источниками данных.	входных данных • Сверхпривил е-гированный доступ • Небезопасны е взаимодействи я с API	вводимых данных • Соблюдение минимальных привилегий • Безопасные методы работы с API • Регулярные аудиты безопасности	обработки ввода • Плагины с избыточными привилегиями для повышения привилегий • SQL-инъекции
Атака типа «отказ в обслуживан ии» (DoS)	Попытки сделать систему недоступной путем перегрузки ее трафиком или спровоцирования сбоев.	• Массовые атаки • Атаки на протокол • Атаки на прикладном уровне	• Ограничение скорости • Надежная инфраструктура • Непрерывный мониторинг и быстрое реагирование	• Перегрузка серверов • Нарушение коммуникации между пользователям и и службами • Нагрузка на системные ресурсы

Оперативное внедрение — это угроза кибербезопасности, при которой обработанные входные данные манипулируют поведением LLM, что может привести к раскрытию конфиденциальных данных или несанкционированным действиям. Методы атаки включают внедрение скрытых подсказок на веб-страницах или в документах и управление моделями для раскрытия конфиденциальной информации. Стратегии смягчения последствий включают ограничение возможностей LLM, требование согласия пользователя на конфиденциальные операции и установление границ доверия для предотвращения использования.

Небезопасная обработка выходных данных возникает, когда приложения доверяют контенту, созданному LLM, без надлежащей проверки, что приводит к серьезным рискам, таким как XSS, CSRF, SSRF, повышение привилегий и удаленное выполнение кода. Если выходные данные LLM (например, JavaScript или SQL) обрабатываются напрямую, они могут спровоцировать вредоносные действия. Для предотвращения этого требуется строгая проверка и кодирование выходных данных, чтобы избежать непреднамеренного выполнения и утечки данных.

DeceptPrompt раскрывает уязвимость Code LLM для использования противоречивых языковых данных, когда кажущиеся безобидными инструкции приводят к созданию функционально корректного, но небезопасного кода. Для предотвращения этого требуется расширенная проверка кода, обучение примерам, связанным с противодействием, и регулярные обновления системы безопасности. Злоумышленники могут использовать этот недостаток для создания уязвимого кода, рискуя утечкой данных и компрометацией системы, что подчеркивает необходимость повышения безопасности в Code LLM.

Зоу и др. предлагают разработать целевые суффиксы, которые обходят меры по выравниванию LLM, что приводит к вредным результатам. Для предотвращения этого требуются усовершенствованные алгоритмы выравнивания, мониторинг в режиме реального времени, постоянная переподготовка моделей и адаптивные системы реагирования. Сценарии атак включают крупномасштабное манипулирование контентом, целенаправленное уклонение от политики и динамические атаки с использованием новых уязвимостей, что подчеркивает необходимость постоянного совершенствования защиты.

Это происходит, когда злоумышленники манипулируют обучающими данными LLM, внедряя ошибки, бэкдоры или вредоносные инструкции, которые нарушают целостность и надежность модели. Предотвращение включает проверку источников данных, изолированную среду, фильтрацию входных данных и мониторинг аномалий. Стратегии защиты включают фильтрацию подозрительных данных перед обучением и точную настройку на основе чистых наборов данных после обучения. Риски атак включают предвзятые результаты, дезинформацию и вред обществу, что подчеркивает необходимость надежной защиты данных.

Эта атака манипулирует входными данными LLM во время работы, вызывая вредоносное поведение без изменения самой модели. Она часто использует упущенные настройки, такие как ограничения на вывод токенов, что затрудняет обнаружение. Предотвращение включает обнаружение аномалий, аудит реагирования и более строгий контроль ввода. Риски атак включают в себя предвзятую информацию, дезинформацию и ущерб репутации, что подчеркивает необходимость обеспечения бдительной безопасности при развертывании LLM.

Небезопасные плагины в LLM создают уязвимости из-за плохой проверки ввода, доступа с избыточными привилегиями и небезопасного взаимодействия с API, включая такие риски, как внедрение SQL. Стратегии предотвращения включают тщательную проверку ввода, доступ с наименьшими привилегиями, безопасные методы API и регулярные проверки безопасности. Потенциальные атаки варьируются от извлечения данных и повышения привилегий до взлома баз данных и эксплуатации системы, что подчеркивает необходимость применения надежных протоколов безопасности при разработке и развертывании плагинов.

DoS-атака нарушает функциональность системы, перегружая ее избыточным трафиком или сложными запросами. Службы LLM сталкиваются с атаками на основе объема (перенасыщение полосы пропускания), атаками на протоколы (использование слабых мест сети) и атаками на прикладном уровне (истощение ресурсов с помощью сложных запросов). Профилактика включает в себя ограничение скорости, надежную инфраструктуру с масштабируемыми серверами и непрерывный мониторинг для быстрого обнаружения и устранения последствий.

Обеспечение безопасности LLM имеет важное значение из-за присущей им восприимчивости к традиционным уязвимостям программного обеспечения и уникальным рискам, связанным с их конструкцией и методами работы. В частности, LLM склонны к быстрому взлому, когда такие методы, как быстрое внедрение, могут использоваться для манипулирования реакциями модели, утечка подсказок, которая может привести к раскрытию обучающих данных, и джейлбрейк, предназначенный для обхода встроенных механизмов безопасности. Эти конкретные угрозы требуют принятия комплексных мер безопасности, которые напрямую решают уникальные проблемы, связанные с LLM. Кроме того, внедрение бэкдоров во время обучения, будь то отравление данных или встраивание секретных триггеров, может значительно изменить поведение модели во время вывода, что создает серьезные риски для целостности данных и надежности модели.

Как обсуждалось в разделе предыдущем, для эффективного устранения этих угроз организации должны применять строгие стратегии защиты, рекомендованные в контрольном списке безопасности OWASP LLM. Это включает в себя тестирование приложений LLM на известные уязвимости с использованием таких методов, как red teaming, и специальных инструментов, таких как garak, для выявления и устранения недостатков безопасности. Кроме того, внедрение систем непрерывного мониторинга, таких как langfuse, в производственных средах помогает обнаруживать и исправлять аномальное поведение или потенциальные нарушения в режиме реального времени. В контрольном перечне OWASP также подчеркивается важность систем управления, которые обеспечивают этичный отбор и обработку данных, используемых при обучении, сохраняя

прозрачность источников данных и типовых методик обучения. Такой структурированный подход к безопасности и управлению гарантирует ответственное использование LLM и их защиту от обычных киберугроз и киберугроз, уникальных по своей операционной природе.

Заключение.

В этой статье мы представили всесторонний и глубокий обзор будущего кибербезопасности через призму генеративного ИИ и больших языковых моделей (LLM). Наше исследование охватывало широкий спектр приложений LLM в области кибербезопасности, включая безопасность проектирования оборудования, обнаружение вторжений, разработку программного обеспечения, проверку проекта, аналитику киберугроз, обнаружение вредоносного ПО, а также обнаружение фишинга и спама, иллюстрируя широкий потенциал LLM в различных областях.

Мы подробно изучили эволюцию и текущее состояние LLM, выделив достижения в 15 конкретных моделях, таких как GPT-4, GPT-3.5, BERT, Falcon и LLaMA. Наш анализ включал в себя углубленный анализ уязвимостей, связанных с LLM, таких как быстрое внедрение, небезопасная обработка выходных данных, отравление данных обучения и вывода, DDoS-атаки и состязательные инструкции на естественном языке. Мы обсудили стратегии смягчения последствий для защиты этих моделей, предложив полное понимание потенциальных сценариев атак и методов их предотвращения.

Наши результаты подчеркивают значительный потенциал LLM в трансформации практик кибербезопасности. Интегрируя LLM в будущие системы кибербезопасности, мы сможем использовать их возможности для разработки более надежных и сложных средств защиты от развивающихся киберугроз. Стратегическое направление, изложенное в этом документе, направлено на руководство будущими исследованиями и развертыванием, подчеркивая важность инноваций и устойчивости для защиты цифровых инфраструктур.

Список литературы

1. Sarker, I. H., Janicke, H., Ferrag, M. A., & Abuadbba, A. (2024). Multi-aspect rule-based AI: Methods, taxonomy, challenges and directions toward automation, intelligence and transparent cybersecurity modeling for critical infrastructures. *Internet of Things, 25*, 101110. https://doi.org/10.1016/j.iot.2024.101110

2. Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., & Zhang, Y. (2024). A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, *4*, 100211. https://doi.org/10.1016/j.hcc.2024.100211

3. Yan, Y., Zhang, Y., & Huang, K. (2024). Depending on yourself when you should: Mentoring LLM with RL agents to become the master in cybersecurity games. *arXiv*. https://doi.org/10.48550/arXiv.2403.17674

4. Sladić, M., Valeros, V., Catania, C., & Garcia, S. (2023). LLM in the shell: Generative honeypots. *arXiv*. https://doi.org/10.48550/arXiv.2309.00155

5. Tann, W., Liu, Y., Sim, J. H., Seah, C. M., & Chang, E.-C. (2023). Using large language models for cybersecurity capture-the-flag challenges and certification questions. *arXiv*. https://doi.org/10.48550/arXiv.2308.10443

6. Lira, O. G., Marroquin, A., & To, M. A. (2024). Harnessing the advanced capabilities of LLM for adaptive intrusion detection systems. In M. B. Nagamalai et al. (Eds.), *Advanced Information Networking and Applications* (pp. 453–464). Springer.

7. Ebert, C. & Beck, M. (2023). Artificial intelligence for cybersecurity. *IEEE Software*, *40*(6), 27–34. https://doi.org/10.1109/MS.2023.3305726

8. Wang, J., Huang, Y., Chen, C., Liu, Z., Wang, S., & Wang, Q. (2024). Software testing with large language models: Survey, landscape, and vision. *IEEE Transactions on Software Engineering*. https://doi.org/10.48550/arXiv.2307.07221

9. Almazrouei, E., Alobeidli, H., Alshamsi, A., Cappelli, A., Cojocaru, R., Debbah, M., Goffinet, É., Hesslow, D., Launay, J., Malartic, Q., … et al. (2023). The Falcon series of open language models. *arXiv*. https://doi.org/10.48550/arXiv.2311.16867

10. Zhou, H., Hu, C., Yuan, Y., Cui, Y., Jin, Y., Chen, C., Wu, H., Yuan, D., Jiang, L., Wu, D., Liu, X., Zhang, C., Wang, X., & Liu, J. (2024). Large language model (LLM) for telecommunications: A comprehensive survey on principles, key techniques, and opportunities. *arXiv*. https://doi.org/10.48550/arXiv.2405.10825

11. Lai, H. & Nissim, M. (2024). A survey on automatic generation of figurative language: From rule-based systems to large language models. *ACM Computing Surveys*, *56*(3), 1–37. https://doi.org/10.1145/3654795

12. Ferrag, M. A., Ndhlovu, M., Tihanyi, N., Cordeiro, L. C., Debbah, M., Lestable, T., & Thandi, N. S. (2024). Revolutionizing cyber threat detection with large language models: A privacy-preserving BERT-based lightweight model for IoT/IIoT devices. *IEEE Access*, *12*, 39892–39904. https://doi.org/10.48550/arXiv.2306.14263

13. Tihanyi, N., Bisztray, T., Dubniczky, R. A., Toth, R., Borsos, B., Cherif, B., Ferrag, M. A., Muzsai, L., Jain, R., Marinelli, R., … et al. (2024). Dynamic intelligence assessment: Benchmarking LLMs on the road to AGI with a focus on model confidence. *arXiv*. https://doi.org/10.48550/arXiv.2410.15490

14. Tihanyi, N., Ferrag, M. A., Jain, R., Bisztray, T., & Debbah, M. (2024). Cybermetric: A benchmark dataset based on retrieval-augmented generation for evaluating LLMs in cybersecurity knowledge. In *Proceedings of the 2024 IEEE International Conference on Cyber Security and Resilience (CSR)* (pp. 296–302). IEEE. https://doi.org/10.48550/arXiv.2402.07688

15. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., … et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, *33*, 1877–1901. https://doi.org/10.48550/arXiv.2005.14165

16. OpenAI. (2023). *GPT-4 technical report*. https://doi.org/10.48550/arXiv.2303.08774

17. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, *21*(1), 5485–5551.

18. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv*. https://doi.org/10.48550/arXiv.1810.04805

19. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. *arXiv*. https://doi.org/10.48550/arXiv.1909.11942

20. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv*. https://doi.org/10.48550/arXiv.1907.11692

21. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. *Advances in Neural Information Processing Systems*, *32*, 5753–5763. https://doi.org/10.48550/arXiv.1906.08237

22. Qi, W., Yan, Y., Gong, Y., Liu, D., Duan, N., Chen, J., Zhang, R., & Zhou, M. (2020). ProphetNet: Predicting future n-gram for sequence-to-sequence pre-training. *arXiv*. https://doi.org/10.48550/arXiv.2001.04063

23. Penedo, G., Malartic, Q., Hesslow, D., Cojocaru, R., Cappelli, A., Alobeidli, H., Pannier, B., Almazrouei, E., & Launay, J. (2023). The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. *arXiv*. https://doi.org/10.48550/arXiv.2306.01116

24. Kitaev, N., Kaiser, Ł., & Levskaya, A. (2020). Reformer: The efficient transformer. *arXiv*. https://doi.org/10.48550/arXiv.2001.04451

25. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., … et al. (2023). PaLM: Scaling language modeling with pathways. *Journal of Machine Learning Research*, *24*(240), 1–113.

26. Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., & Chen, Z. (2023). PaLM 2 technical report. *arXiv*. https://doi.org/10.48550/arXiv.2305.10403

27. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., & Azhar, F. (2023). LLaMA: Open and efficient foundation language models. *arXiv*. https://doi.org/10.48550/arXiv.2302.13971

28. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., & Bhosale, S. (2023). LLaMA 2: Open foundation and fine-tuned chat models. *arXiv*. https://doi.org/10.48550/arXiv.2307.09288

29. 01.AI. (2025). *Yi-34B* [Model]. Hugging Face. https://huggingface.co/01-ai/Yi-34B

30. OWASP Foundation. (2025). *OWASP Top 10 for Large Language Model Applications* [Project]. https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/

References

1. Sarker, I. H., Janicke, H., Ferrag, M. A., & Abuadbba, A. (2024). Multi aspect rule based AI: Methods, taxonomy, challenges and directions toward automation, intelligence and transparent cybersecurity modeling for critical infrastructures. Internet of Things, 25, 101110. https://doi.org/10.1016/j.iot.2024.101110

2. Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., & Zhang, Y. (2024). A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly. High-Confidence Computing, 4, 100211. https://doi.org/10.1016/j.hcc.2024.100211

3. Yan, Y., Zhang, Y., & Huang, K. (2024). Depending on yourself when you should: Mentoring LLM with RL agents to become the master in cybersecurity games. arXiv. https://doi.org/10.48550/arXiv.2403.17674

4. Sladić, M., Valeros, V., Catania, C., & Garcia, S. (2023). LLM in the shell: Generative honeypots. arXiv. https://doi.org/10.48550/arXiv.2309.00155

5. Tann, W., Liu, Y., Sim, J. H., Seah, C. M., & Chang, E.-C. (2023). Using large language models for cybersecurity capture the flag challenges and certification questions. arXiv. https://doi.org/10.48550/arXiv.2308.10443

6. Lira, O. G., Marroquin, A., & To, M. A. (2024). Harnessing the advanced capabilities of LLM for adaptive intrusion detection systems. In M. B. Nagamalai et al. (Eds.), Advanced Information Networking and Applications (pp. 453–464). Springer.

7. Ebert, C. & Beck, M. (2023). Artificial intelligence for cybersecurity. IEEE Software, 40(6), 27–34. https://doi.org/10.1109/MS.2023.3305726

8. Wang, J., Huang, Y., Chen, C., Liu, Z., Wang, S., & Wang, Q. (2024). Software testing with large language models: Survey, landscape, and vision. IEEE Transactions on Software Engineering. https://doi.org/10.48550/arXiv.2307.07221

9. Almazrouei, E., Alobeidli, H., Alshamsi, A., Cappelli, A., Cojocaru, R., Debbah, M., Goffinet, É., Hesslow, D., Launay, J., Malartic, Q., … et al. (2023). The Falcon series of open language models. arXiv. https://doi.org/10.48550/arXiv.2311.16867

10. Zhou, H., Hu, C., Yuan, Y., Cui, Y., Jin, Y., Chen, C., Wu, H., Yuan, D., Jiang, L., Wu, D., Liu, X., Zhang, C., Wang, X., & Liu, J. (2024). Large language model (LLM) for telecommunications: A comprehensive survey on principles, key techniques, and opportunities. arXiv. https://doi.org/10.48550/arXiv.2405.10825

11. Lai, H. & Nissim, M. (2024). A survey on automatic generation of figurative language: From rule based systems to large language models. ACM Computing Surveys, 56(3), 1–37. https://doi.org/10.1145/3654795

12. Ferrag, M. A., Ndhlovu, M., Tihanyi, N., Cordeiro, L. C., Debbah, M., Lestable, T., & Thandi, N. S. (2024). Revolutionizing cyber threat detection with large language models: A privacy preserving BERT based lightweight model for IoT/IIoT devices. IEEE Access, 12, 39892–39904. https://doi.org/10.48550/arXiv.2306.14263

13. Tihanyi, N., Bisztray, T., Dubniczky, R. A., Toth, R., Borsos, B., Cherif, B., Ferrag, M. A., Muzsai, L., Jain, R., Marinelli, R., … et al. (2024). Dynamic intelligence assessment: Benchmarking LLMs on the road to AGI with a focus on model confidence. arXiv. https://doi.org/10.48550/arXiv.2410.15490

14. Tihanyi, N., Ferrag, M. A., Jain, R., Bisztray, T., & Debbah, M. (2024). Cybermetric: A benchmark dataset based on retrieval augmented generation for evaluating LLMs in cybersecurity knowledge. In Proceedings of the 2024 IEEE International Conference on Cyber Security and Resilience (CSR) (pp. 296–302). IEEE. https://doi.org/10.48550/arXiv.2402.07688

15. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., … et al. (2020). Language models are few shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901. https://doi.org/10.48550/arXiv.2005.14165

16. OpenAI. (2023). GPT 4 technical report. https://doi.org/10.48550/arXiv.2303.08774

17. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text to text transformer. Journal of Machine Learning Research, 21(1), 5485–5551.

18. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv. https://doi.org/10.48550/arXiv.1810.04805

19. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. arXiv. https://doi.org/10.48550/arXiv.1909.11942

20. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv. https://doi.org/10.48550/arXiv.1907.11692

21. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. Advances in Neural Information Processing Systems, 32, 5753–5763. https://doi.org/10.48550/arXiv.1906.08237

22. Qi, W., Yan, Y., Gong, Y., Liu, D., Duan, N., Chen, J., Zhang, R., & Zhou, M. (2020). ProphetNet: Predicting future n gram for sequence to sequence pre training. arXiv. https://doi.org/10.48550/arXiv.2001.04063

23. Penedo, G., Malartic, Q., Hesslow, D., Cojocaru, R., Cappelli, A., Alobeidli, H., Pannier, B., Almazrouei, E., & Launay, J. (2023). The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. arXiv. https://doi.org/10.48550/arXiv.2306.01116

24. Kitaev, N., Kaiser, Ł., & Levskaya, A. (2020). Reformer: The efficient transformer. arXiv. https://doi.org/10.48550/arXiv.2001.04451

25. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., … et al. (2023). PaLM: Scaling language modeling with pathways. Journal of Machine Learning Research, 24(240), 1–113.

26. Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., & Chen, Z. (2023). PaLM 2 technical report. arXiv. https://doi.org/10.48550/arXiv.2305.10403

27. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., & Azhar, F. (2023). LLaMA: Open and efficient foundation language models. arXiv. https://doi.org/10.48550/arXiv.2302.13971

28. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., & Bhosale, S. (2023). LLaMA 2: Open foundation and fine tuned chat models. arXiv. https://doi.org/10.48550/arXiv.2307.09288

29. 01.AI. (2025). Yi 34B [Model]. Hugging Face. https://huggingface.co/01-ai/Yi-34B

30. OWASP Foundation. (2025). OWASP Top 10 for Large Language Model Applications [Project]. https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/

LLM-ДІ КИБЕРҚАУІПСІЗДІКТЕ ҚОЛДАНУ: LLM ҚОСЫМШАЛАРЫ МЕН ОСАЛДЫҚТАРЫНА ШОЛУ

Аңдатпа*.Бұл құжат үлкен тілдік модельдер (LLM) арқылы Киберқауіпсіздіктің болашағына жан-жақты шолу жасайды. Біз GPT-4, GPT-3.5, BERT, Falcon2 және LLaMA сияқты модельдердегі жетістіктерге назар аудара отырып, LLM эволюциясы мен оның қазіргі жағдайына шолу жасаймыз. Біздің талдауымыз LLM осалдықтарын қамтиды, мысалы, жылдам іске асыру, қауіпсіз Шығыс өңдеу, деректермен улану, DDoS шабуылдары және қарсыласу нұсқаулары. Біз ықтимал шабуыл сценарийлері мен олардың алдын алу әдістеріне жан-жақты шолу жасай отырып, осы үлгілерді қорғаудың азайту стратегияларын егжей-тегжейлі қарастырамыз. Бұл аналитикалық деректер нақты уақыттағы киберқауіпсіздіктен қорғауды жақсартуға және қауіптерді анықтау және оларға жауап беру үшін LLM қолданбаларының күрделілігін арттыруға бағытталған. Біздің құжат дамып келе жатқан киберқауіпсіздіктерден қорғау үшін LLM-ді болашақ киберқауіпсіздік жүйелеріне біріктірудің негізгі түсінігі мен стратегиялық бағытын ұсынады.*

Түйін сөздер*: LLM, киберқауіпсіздік, үлкен тілдік модельдер, тілдік модельдеу, Машиналық оқыту, NLP, табиғи тілді өңдеу.*

THE USE OF LLM IN CYBERSECURITY: OVERVIEW OF LLM APPLICATIONS AND VULNERABILITIES

Abstract. *This document provides a comprehensive overview of the future of cybersecurity through Large Language Models (LLM). We present an overview of the evolution of LLM and its current state, focusing on advances in models such as GPT-4, GPT-3.5, BERT, Falcon2, and LLaMA. Our analysis extends to LLM vulnerabilities such as rapid deployment, insecure output processing, data poisoning, DDoS attacks, and adversarial instructions. We will take a detailed look at mitigation strategies to protect these models, providing a comprehensive overview of potential attack scenarios and methods to prevent them. This analytical data is aimed at improving real-time cybersecurity protection and increasing the complexity of LLM applications for threat detection and response. Our document provides a fundamental understanding and strategic direction for integrating LLM into future cybersecurity systems to protect against evolving cyber threats.*

Keywords: *LLM, cybersecurity, large language models, language modeling, machine learning, NLP, natural language processing.*

Сведение об авторах

Курасбек Аяжан Нурлыбеккызы	Магистрант, Системы информационной безопасности, Казахский национальный университет имени аль-Фараби, г. Алматы, Казахстан, E-mail: kurasbek.a.140@mail.ru
Кенжебаева Мерей Омаровна	PhD, и. о. доцента кафедры кибербезопасности и криптологии, Казахский национальный университет имени аль-Фараби, г. Алматы, Казахстан, E-mail: merey-mex-2017@mail.ru
Адилжанова Салтанат Адьмуханбетовна	PhD, и. о. доцента кафедры кибербезопасности и криптологии, Казахский национальный университет имени аль-Фараби, г. Алматы, Казахстан, E-mail: asaltanat81@gmail.com

Авторлар туралы мәлімет

Құрасбек Аяжан Нұрлыбекқызы	Магистрант, Ақпараттық қауіпсіздік жүйелері, әл-Фараби атындағы Қазақ ұлттық университеті, Алматы қ., Қазақстан, E-mail: kurasbek.a.140@mail.ru
Кенжебаева Мерей Омаровна	PhD, киберқауіпсіздік және криптология кафедрасының доцентінің м. а., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы қ., Қазақстан, E-mail: merey-mex-2017@mail.ru
Адилжанова Салтанат Адьмуханбетовна	PhD, киберқауіпсіздік және криптология кафедрасының доцентінің м. а., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы қ., Қазақстан E-mail: asaltanat81@gmail.com

Information about the authors

Kurasbek Ayazhan	Master's student, Information Security Systems, Al-Farabi Kazakh National University, Almaty, Kazakhstan, E-mail: kurasbek.a.140@mail.ru
Kenzhebayeva Merey	PhD, Acting Associate Professor of the Department of Cybersecurity and Cryptology, Al-Farabi Kazakh National University, Almaty, Kazakhstan E-mail: merey-mex-2017@mail.ru
Adilzhanova Saltanat	PhD, Acting Associate Professor of the Department of Cybersecurity and Cryptology, Al-Farabi Kazakh National University, Almaty, Kazakhstan, E-mail: asaltanat81@gmail.com