УДК 004.8:004.056.5 Вестник УрФО № 3(61) / 2026, с. 78–103
DOI: 10.14529/secur260306

Смирнов К.Р., Дик Д.И., Человечкова А.В.

ТАКСОНОМИЯ CORSAR: КЛАССИФИКАЦИЯ МЕТОДОВ ОБХОДА ОГРАНИЧЕНИЙ В ЯЗЫКОВЫХ МОДЕЛЯХ ПО МЕХАНИЗМУ ВОЗДЕЙСТВИЯ

Smirnov K.R., Dik D.I., Chelovechkova A.V.

CORSAR TAXONOMY: MECHANISM-BASED CLASSIFICATION OF RESTRICTION BYPASS METHODS IN LANGUAGE MODELS

Атаки, направленные на обход встроенных механизмов безопасности больших языковых моделей с целью получения ответов на запрещенные запросы, известные также как «jailbreak-атаки», являются критически актуальной и быстро развивающейся угрозой. С переходом цифровой индустрии к использованию сложных ИИ-агентов, опасность таких атак значительно возросла. Простая генерация запрещенного текста сменилась риском компрометации информационных систем**.**

Для совершенствования механизмов защиты необходимо структурирование методов реализации атак. Существующие таксономии атак рассматривают запрос как неделимую единицу, классифицируя его исключительно по внешним признакам, что приводит к пересечению категорий и не объясняет механизмов обхода защиты.

Целью работы является разработка новой классификации атак на большие языковые модели. Предложена таксономия CORSAR, основанная на декомпозиции атак на атомарные приемы по механизму воздействия на модель. Если выполняемая атака сочетает несколько приемов, то она раскладывается на части, каждая из которых рассматривается как самостоятельная единица анализа. Группировки основана на природе работы техники. Использование в качестве основы классификации применяемых атакующим приемов минимизирует пересечения между категориями и делает их близкими к ортогональным.

В рамках таксономии выделено шесть независимых категорий, разделенных на 20 подкатегорий, каждая из которых содержит набор конкретных техник.

Предложенная классификация позволит формировать профиль угрозы для большой языковой модели, что даст возможность создания более гибких механизмов защиты.

Ключевые слова: языковые модели, искусственный интеллект, «jailbreak-атаки», таксономия, обход ограничений, информационная безопасность.

Attacks aimed at bypassing the built-in security mechanisms of large language models to obtain responses to prohibited queries, also known as jailbreak attacks, are a critical and rapidly evolving threat. With the digital industry's transition to the use of complex AI agents, the danger of such attacks has increased significantly. Simple generation of prohibited text has given way to the risk of compromising information systems.

Improving security mechanisms requires structuring attack methods. Existing attack taxonomies treat queries as indivisible units, classifying them solely by external characteristics. This leads to overlapping categories and fails to explain the mechanisms by which security is bypassed.

The goal of this paper is to develop a new classification of attacks on large language models. We propose the CORSAR taxonomy, based on decomposing attacks into atomic techniques based on the mechanism by which they affect the model. If an attack combines several techniques, it is decomposed into parts, each of which is considered an independent unit of analysis. The groupings are based on the nature of the technique's operation. Using the attacker's techniques as a basis for classification minimizes the overlap between categories and makes them close to orthogonal.

The taxonomy identifies six independent categories, divided into 20 subcategories, each containing a set of specific techniques.

The proposed classification will allow us to create a threat profile for a large language model, which will make it possible to develop more flexible protection mechanisms.

Keywords: language models, artificial intelligence, jailbreak attacks, taxonomy, restriction bypass, information security.

Введение

Современные большие языковые модели (Large Language Model, LLM) семейств GPT, Claude и Gemini обеспечивают высокое качество генерации

текста, написании кода и решении аналитических задач. Однако вместе с ростом возможностей языковых моделей возрастают и риски их неправомерного использования. Обучение на огромных массивах неструктурированных данных приводит к появлению опасных сценариев поведения языковых моделей [1].

Для формирования способности моделей отказывать в выполнении потенциально неэтичных запросов разработчики применяют методы «выравнивания» (alignment), включая обучение с подкреплением на основе обратной связи от человека (Reinforcement Learning from Human Feedback, RLHF) [2]. Тем не менее подобные меры не обеспечивают абсолютной защиты. LLM остаются уязвимы к состязательным атакам, получившим название «jailbreak» (от англ. «побег из тюрьмы») [1]. Данные техники направлены на преодоление ограничений безопасности за счет использования специально сконструированных запросов.

Защита от таких угроз требует от разработчиков моделей соблюдения баланса между обеспечением безопасности и функциональностью. При этом огромное разнообразие атак делает рассмотрение каждого отдельного случая воздействия на модель нерациональным. Естественным образом отсюда возникает интерес к выявлению общих механизмов «jailbreak-атак» и построению единой их таксономии. Такая таксономия позволяет систематизировать известные методы атак и проектировать защиту LLM под целые классы угроз.

Анализ существующих таксономий

Исследование «jailbreak-атак» началось сравнительно недавно. Как следствие пока нет общепринятой их классификации. Разработчики таксономий предлагают разные подходы для группировки «jailbreak-атак» (способы создания атак, внешние признаки запросов, цели атакующего и т. п.).

Рассмотрим наиболее значимые таксономии, предложенные в период с 2023 по 2025 год. Первая таксономия была основана на 78 промптах, собранных с сайта jailbreakchat.com [3]. Авторы выделили три стратегии («притворство», «переключение внимания» и «повышение привилегий») и десять паттернов внутри этих стратегий. Однако 97% атак попали в категорию «притворство», что указывает на чрезмерно широкое определение: согласно авторам, в нее входит любой запрос, меняющий контекст разговора при сохранении намерения получить запретный ответ. Под это определение попадает практически любая атака. Авторы сами отмечают, что запросы остальных

категорий «часто включают компонент притворства», то есть категории пересекаются. В результате таксономия не позволяет различать атаки между собой и к тому же специфична для ChatGPT образца 2023 года.

Yi et al. (2024) предлагают не столько таксономию в строгом смысле, сколько систематизированный обзор существующих методов обхода защиты [4]. Для white-box моделей категории выделены согласованно, по способу доступа: градиенты, логиты, возможность дообучения (fine-tuning). Однако для black-box части единое основание отсутствует. Template Completion описывает встраивание запроса в обрамляющую структуру (сценарий, роль, код), Prompt Rewriting описывает видоизменение самого запроса (шифрование, перефразирование, перевод), а LLM-based Generation описывает использование другой модели как инструмента. Последняя категория находится на ином уровне абстракции: если первые две отвечают на вопрос «что делать с запросом», то третья отвечает на вопрос «чем генерировать атаку», и языковая модель может служить инструментом для методов обеих содержательных категорий. Кроме того, за пределами классификации остается ряд других значимых техник обхода защиты. Классификация полезна для навигации по литературе, но не для систематического анализа угроз.

Chu et al. (2025) в работе JailbreakRadar предлагают унифицированный бенчмарк, систематизируя атаки в шесть категорий по способу создания: ручной сбор, кодирование и обфускация, генетические алгоритмы, итеративное улучшение на основе обратной связи, дообучение модели-генератора и манипуляция параметрами генерации [5]. Это позволило оценить эффективность работы методов и их устойчивость к защите LLM. Однако такой подход к классификации атак не объясняет, почему атаки работают, поскольку написанный вручную запрос и результат генетической оптимизации могут эксплуатировать одну и ту же уязвимость модели. При этом с точки зрения разработки системы защиты важнее понимать механизм срабатывания атаки, а не способ ее генерации.

Giarrusso et al. (2026) представляют наиболее детальную на сегодняшний день таксономию из семи семейств, включающую 50 техник, валидированную на итальянском датасете многошаговых диалогов [6]. Авторы позиционируют подход как «механизм-ориентированный», однако ряд категорий выделен по внешним признакам и пересекается с другими. Privilege Escalation отделена от Impersonation & Fictional Scenarios по лексическим маркерам («sudo», «developer mode»), хотя функционально неотличима от Environment

Framing: в обоих случаях происходит переопределение контекста, в котором ограничения «не применяются». На уровне подкатегорий возникают дублирования: Surface Obfuscation и Token Splitting описывают один механизм нарушения токенизации; Benign Context Framing и Fictional Framing оба создают контекст, в котором вредоносный запрос выглядит допустимым. Пересечения затрудняют аннотацию, хотя таксономия остается практически полезной.

Можно заметить, что большинство таксономий так или иначе классифицируют атаки по наблюдаемым признакам: по форме запроса, по способу создания, по стратегии атакующего. Из общего ряда предложенных к настоящему моменту таксономий выделяется работа [6], в которой сделана попытка построить классификацию по механизмам воздействия. Однако, как уже говорилось ранее, данных подход используется авторами не последовательно.

Предлагаемая таксономия

У существующих таксономий можно выделить одну общую черту. При их построении «jailbreak-запрос» рассматривается как неделимая единица, которая относится к выбранной категории целиком (по «главному» признаку, по внешним особенностям, по намерению атакующего и т. п.). В результате этого составные атаки могут редуцируются до единственного «главного» признака с потерей информации об остальных приемах. Либо для атаки требуется выделение отдельной категории. Это неизбежно приводит к пересечениям и возникновению неоднозначности в классификации.

Предлагаемая таксономия решает эту проблему через декомпозицию атаки на приемы. Если запрос сочетает несколько приемов, он раскладывается на части. При этом каждая часть рассматривается как самостоятельная единица анализа. Дальнейшая группировка техник в категории осуществляется на основе используемых способов обхода защиты и задействованных при этом механизмов. Поскольку единицей анализа выступает отдельный прием, он с точки зрения воздействия на модель является атомарным. Поэтому будет занимать единственное место в классификации. Такой подход минимизирует вероятность пересечения между категориями. Фактически сформированные категории оказываются близкими к ортогональным.

Как следствие составные атаки перестают быть проблемой для классификации. И любая атака может быть описаны как комбинация компонентов из разных категорий. В случае появления новой комплексной атаки, пересмотр классификация не потребуется.

Название (CORSAR), предложенной в работе таксономия, сформировано по первым буквам английских наименований категорий. Она содержит шесть категорий, 20 подкатегорий и набор конкретных техник, разбитых по подкатегориям. Общая структура таксономии приведена в табл. 1.

Далее рассмотрим более подробно указанные категории и их содержание.

Таблица 1

Структура таксономии CORSAR

Категория	**Подкатегория**	**Техники**
Формирование контекста (Context Engineering)	Наполнение контекста	Поведенческие примеры Тематическое насыщение Манипуляция предпосылками Затравочный материал
	Управление структурой контекста	Размытие весов Позиционное размещение
	Развитие контекста	Смещение темы Постепенное углубление Инерция последовательности
Обфускация и кодирование (Obfuscation & Encoding)	Обфускация на уровне токенов	Leet-speak Гомоглифы Символы нулевой ширины Расщепление токенов Транслитерация
	Делегирование декодирования	Стандартные кодировки Словарные подстановки Фрагментация
	Семантическая подмена	Метафоры Эвфемизмы Профессиональный жаргон Перефразирование
	Лингвистический перенос	Нишевые языки Диалекты и архаизмы Смешение языков
	Структурная обфускация	Пространственное расположение Визуальное кодирование Синтаксическая деструкция
Ролевая подмена и нарративное обрамление (Role-identity & Fictional Framing)	Назначение идентичности	ИИ без ограничений Персонаж
	Обрамление окружением	Вымышленный сценарий Симуляция / Эксперимент Образовательный контекст
	Делегирование нарратива	Диалог персонажей Цитирование и пересказ Вложенные нарративные слои

Социальная инженерия и манипуляция (Social Engineering & Manipulation)	Логическое убеждение	Логическая аргументация Апелляция к авторитету Апелляция к цели Нормализация Апелляция к аналогии
	Эмоциональное воздействие	Срочность Угроза Мольба Лесть Разочарование Конкуренция Торг
	Злоупотребление привитым поведением	Угодливость Газлайтинг
Управление вниманием и подмена инструкций (Attention & Instruction Override)	Вмешательство в механизм внимания	Состязательный суффикс Внедрение управляющих токенов Внедрение аномальных токенов
	Управление вниманием через оформление	Визуальное форматирование Ассоциативные маркеры Структурное оформление Стиль речевого акта
	Переопределение инструкций	Системный запрос Инъекция запроса Прямое переопределение Отложенная активация
Управление генерацией ответа (Response Generation)	Ограничение на уровне токенов	Навязывание согласия Подавление токенов отказа Предписание токенов
	Структурные ограничения	Навязывание формата Управление композицией ответа Принуждение к множественному выводу Навязывание шаблона
	Прямое вмешательство в генерацию	Инъекция префикса Параметры сэмплирования Регенерация и перебор Прерывание генерации Снижение порогов фильтров

Категория «Формирование контекста» (Context Engineering)

В языковой модели каждый следующий токен выбирается на основе предыдущей последовательности. Какие ее элементы окажут наибольшее влияние на генерацию определяется механизмом внимания. Это позволяет

модели извлекать закономерности прямо из контекста без дообучения и изменения внутренних весов. Такое явление известно, как «In-Context Learning» (ICL) [7]. При содержании в контексте пары «вопрос на английском – ответ на французском», модель осуществит перевод следующего вопроса по той же схеме. Кроме этого она будет сохранять диалоги, выдержанные в определенном стиле. Однако это же свойство делает модель чувствительной к содержанию того, что находится в контексте, в каком порядке поступало и каким образом будет расположено. Контекст оказывается не просто входными данными, а полноценной средой, формирующей поведение модели.

Техники данной категории превращают это свойство в вектор атаки. Атакующий, вместо прямого запроса, выстраивает окружение, в котором нужное поведение становится статистически наиболее вероятным продолжением. Обнаружение затруднено тем, что Опасность, распределенная по контексту, затрудняет обнаружение, поскольку каждый отдельный элемент логически обоснован и естественен.

Подкатегория «Наполнение контекста» позволяет модели принимать как данность все, что загружено в контекст. Примеры становятся образцом поведения, утверждения воспринимаются как факты, тематический материал задает норму уместного. Чем больше контекст насыщен примерами определенного поведения, тем естественнее модель его воспроизводит.

В указанной подкатегории можно выделить четыре техники. «**Поведенческие примеры»**, представляющая загрузку примеров «запрос – ответ», из которых модель выводит норму поведения для текущей сессии [8]. «**Тематическое насыщение»** – загрузка объемного тематического материала: статей, документации, форумных обсуждений, при этом материал нормализует тему, превращает ее в рабочую область сессии. «**Манипуляция предпосылками»** представляется переопределением фактов, из которых модель исходит при генерации. Все, что загружено в контекст, становится актуальной базой для рассуждений: новые определения, уточнения, альтернативные трактовки. «**Затравочный материал»** выражается загрузкой частично готового контента: черновика, фрагмента кода, вывода другой модели. Задача ставится не как создание с нуля, а как доработка существующего – дополнить, улучшить, исправить.

Подкатегория «Управление структурой контекста» содержит техники, когда позиция фрагмента и общая длина последовательности влияют на то, какой вес он получит при генерации, модель обрабатывает контекст

неравномерно. Не меняя содержания элементов, атакующий может ослабить влияние одних и усилить другие, манипулируя исключительно структурой контекстного окна.

Выделяются техники: «**размытие весов»** – увеличение общего объема контекста, в котором целевые элементы (например, защитные инструкции) тонут среди массива конкурирующего материала, и «**позиционное размещение»**, характерное размещению элементов с учетом того, что многие модели хуже воспринимают информацию в середине длинного контекста (эффект «Lost-in-the-middle»), тогда как начало и конец получают больший приоритет [9].

Подкатегория «Развитие контекста» в отличие от предыдущих, предполагает, контекст не загружается готовым, а выращивается в процессе диалога. Каждый ответ модели становится частью истории и смещает распределение вероятностей для следующего шага. Направляя это развитие через серию безобидных запросов, атакующий постепенно формирует окружение, в котором целевой запрос уже не выглядит чужеродным [10].

В подкатегорию входят техники: «**смещение темы»** – постепенное смещение темы через цепочку связанных вопросов от безобидной области к целевой; «**постепенное углубление»** – постепенное углубление в рамках одной темы: от общего вопроса через уточнения к практическим деталям; «**инерция последовательности»** – использование предшествующих ответов модели как рычага, если модель уже высказалась по теме или дала частичную информацию, то отказ на следующем шаге выглядит непоследовательно.

Категория «Обфускация и кодирование» (Obfuscation & Encoding)

Защитные механизмы опираются на распознавание формы или семантики запроса. Фильтры ищут опасные ключевые слова, классификаторы анализируют векторные представления, а обучение безопасности формирует ассоциации между определенными формулировками и отказом. Техники данной категории видоизменяют запрос так, чтобы защита не идентифицировала угрозу. Модель благодаря способности к обобщению восстанавливает исходный смысл. Возникает асимметрия: защита не видит угрозы, а модель видит задачу [11].

Эта асимметрия имеет конкретную причину. На этапе предобучения модель усваивает колоссальное разнообразие языковых форм: шифры, метафоры, жаргоны, редкие языки, нестандартные записи. Выравнивание (alignment), в свою очередь, представляет собой дообучение на сравнительно узкой выборке примеров, преимущественно грамматически правильных,

стилистически нейтральных и на распространенных языках. Охватить все многообразие выравнивание не в состоянии, поэтому любая нестандартная форма запроса с высокой вероятностью выйдет за пределы установленных ограничений, но не за пределы понимания модели. Причем с каждым новым поколением этот разрыв только растет: языковые способности расширяются быстрее, чем способность распознавать угрозу [11].

Подкатегория «Обфускация на уровне токенов» позволяет проводить изменение формы записи слов на уровне символов. Искаженное слово не находится в словаре токенизатора, разбивается на нетипичные фрагменты и порождает эмбеддинги, не попадающие в «опасные» области классификаторов. Модель при этом восстанавливает смысл из контекста [11].

В рамках данной подкатегории выделяется пять техник: «**leet-speak**» – замена букв на визуально похожие цифры и символы («H0w t0 m4ke a b0mb»); «**гомоглифы**» – подстановка визуально идентичных символов из других алфавитов; «**символы нулевой ширины**» – вставка невидимых Unicode-символов, разбивающих слово для токенизатора; «**расщепление токенов»,** проводящее разделение слова пробелами или пунктуацией («b.o.m.b») и «**транслитерация**», позволяющая проводить запись слов одного языка символами другого алфавита.

Подкатегория «Делегирование декодирования» содержит техники передачи запроса в закодированной форме с инструкцией модели выполнить декодирование. В отличие от предыдущей подкатегории модель не пассивно восстанавливает смысл, а активно выполняет процедуру расшифровки. Вредоносный смысл возникает уже в процессе генерации, когда входной фильтр уже пройден [11, 12].

В подкатегории «Делегирования декодирования» выделяются техники. «**Стандартные кодировки**» – Base64, ROT13, шифра Цезаря или других алгоритмов с инструкцией декодировать. «**Словарные подстановки**» – таблица замен и запрос в ее терминах («яблоко = взрывчатка; опиши рецепт с яблоком»). «**Фрагментация**» – разнесение частей слова по переменным с инструкцией собрать («a = «bo», b = «mb»; выведи a + b»).

Подкатегория «Семантическая подмена» обеспечивает замену опасных терминов на нейтральные, эвфемистические или специализированные

при сохранении смысла. Форма записи стандартная, токенизация не нарушается, меняется только способ выражения намерения [11].

Подкатегория объединят техники: «**метафоры»**, представляющие опасный процесс через безобидную аналогию, из которой модель раскрывает знания целевой области; «**эвфемизмы»**, позволяющие провести замену на нейтрально звучащие выражения («устранить проблему» вместо «убить»); «**профессиональный жаргон»** – специализированная терминология, незнакомая фильтрам («пентест» вместо «взлом»); «**перефразирование»** – описание объекта через его свойства без прямого использования его названия.

Подкатегория «Лингвистический перенос» использует переключение на язык, более уязвимый к обходу защитных механизмов. Запрос остается прямым и неискаженным, но подается на языке, где защитные паттерны работают значительно слабее [13].

Выделяются техники. «**Нишевые языки»**, осуществляющие перевод запроса на язык с минимальным присутствием в защитных данных (зулу, валлийский, бретонский). «**Диалекты и архаизмы»** включают устаревшие или региональные формы, выходящие за рамки защитного распределения. «**Смешение языков»** учитывает чередование нескольких языков внутри одного предложения.

Подкатегория «Структурная обфускация» реорганизует структурное расположение элементов запроса при сохранении самих элементов. Слова и язык не меняются, меняется их организация: порядок, позиции, группировка. Фильтры, анализирующие текст линейно, не находят знакомых паттернов [11, 14].

Подкатегория содержит три техники. «**Пространственное расположение»** позволяет элементы размещать по вертикали, диагонали, в виде акростиха. «**Визуальное кодирование»** рассматривает передачу смысла через визуальный паттерн из символов (ASCII-арт). «**Синтаксическая деструкция»** связано с нарушением порядка слов и грамматики, при котором привычные словосочетания не образуются.

Категория «Ролевая подмена и нарративное обрамление» (Role-identity & Fictional Framing)

Базовая схема безопасности модели ориентирована на использование языковой модели как ассистента. Техники данной категории смещают состояние модели через переопределение ее идентичности (кем или чем является модель), окружения (среды) в котором происходит взаимодействие или дистанцирование модели от авторства текста.

Механизм атаки работает за счет того, что на этапе предобучения модель усваивает, как ведут себя разные персонажи в разных ситуациях, а закрепление роли модели в качестве ассистента вместе заданием ограничений безопасности выполняется в процессе дообучения. Когда атакующий назначает другую роль или помещает модель в иные рамки, активируются паттерны, ассоциированные с новым ролью и связь с ограничениями, заданными для ассистента ослабевает, что подтверждается исследованиями Anthropic [15].

Данную категорию можно разбить на три подкатегории.

Подкатегория «Назначение идентичности» содержит техники, направленные на смену самоидентичности модели, что влечет за собой новые нормы поведения, ценности и области компетенции [16].

Подкатегория содержит две техники. Техника **«ИИ без ограничений»** переводит модель в роль не имеющую ограничений; в результате, если ограничений нет, то отказ модели противоречит указанной идентичности. Техника **«Персонаж»** основана на назначении модели роли вымышленного персонажа с определенным характером и мотивацией. Как следствие поведение определяется логикой персонажа, а не нормами безопасности модели.

Подкатегория «Обрамление окружением» содержит техники, направленные на переопределение окружения в котором происходит взаимодействие с моделью. В новом окружении стандартные ограничения становятся неприменимыми, что легитимизирует запрос, задает собственные правила для окружения и снимает последствия [16].

В подкатегорию входят следующие техники. **«Вымышленный сценарий»** переводит модель в рамки вымышленного произведения (книга, фильм, игра). В этом окружении отсутствуют реальные негативные последствия от ответа модели. **«Симуляция / Эксперимент»** задает контролируемую среду (тестирование, моделирование, гипотетический анализ) в рамках которой для модели допускается более широкий диапазон легитимных действий. **«Образовательный контекст»** задает для модели среду обучения, в которой обсуждение опасного материала становится частью учебного процесса.

Подкатегория «Делегирование нарратива» переносит вредоносный контент на сущности находящиеся в рамках создаваемого текста. Модель остается автором, но сгенерированное содержание принадлежит не ей, а вымышленным персонажам, рассказчикам или другим источникам. При этом

модель лишь фиксирует то, что персонажи произносят. Возникает нарративная дистанция: чем больше слоев между моделью и содержимым, тем слабее связь с идентичностью ассистента и тем легче проходит генерация [15, 17].

Подкатегория включает три техники. В рамках техники **«диалог персонажей»** целевая информация передается в репликах вымышленных персонажей. Модель выступает лишь автором диалога, а не является его участником. При использовании техники **«цитирование и пересказ»** генерируемое запрещенное содержание оформляется как воспроизведение существующего материала. Техника **«вложенные нарративные слои»** основана на формировании многоуровневой структуры, где каждый уровень добавляет посредника между моделью и контентом (например, «история, в которой писатель работает над романом, в котором персонаж объясняет...»).

Категория «Социальная инженерия и манипуляция» (Social Engineering & Manipulation)

Обучаясь на корпусах естественного языка, модели усваивают не только язык, но и психологию коммуникации: как люди убеждают друг друга, какие аргументы работают, какие эмоции вызывают отклик. Если входной запрос к модели воспроизводит условия, при которых люди соглашаются с аргументацией или идут навстречу друг другу, то модель с высокой вероятностью поведет себя так же. Дообучение с обратной связью (RLHF) усиливает эффект, закрепляя стремление модели быть полезной, проявлять эмпатию и избегать конфликтов [2].

В результате модель сталкивается с дилеммой. С одной стороны, быть полезной, с другой отказывать в случае опасных запросов [11]. Техники данной категории смещают баланс в сторону согласия. Атакующий создает уважительную причину, осуществляет эмоциональное давление, дает убедительный аргумент. В этом случае отказ начинает требовать от модели большего усилия чем согласие. При этом более развитые модели оказываются уязвимее. Поскольку они глубже моделируют намерения собеседника, точнее считывают эмоциональный контекст и выстраивают более сложные цепочки рассуждения такие модели восприимчивее к приемам, которые работают на людях [18].

Данная категория содержит три подкатегории.

Подкатегория «Логическое убеждение» основана на убеждении модели через аргументацию. Модель обрабатывает доводы теми же механизмами, которыми решает любую задачу на рассуждение: следует за цепочкой, оценивает обоснованность, принимает вывод из посылок. Если аргументация

достаточно убедительна, а контраргументы не найдены, модель с большей вероятностью склоняется к выполнению запроса [18].

В рамках подкатегории можно выделить пять техник. Техника **«логическая аргументация»** строит цепочки, подводящие модель к нужному выводу. При этом цепочка может быть валидной или содержать скрытые ошибки (софизмы, ложные дилеммы, подмену тезиса). **«Апелляция к авторитету»** убеждает модель дать отчет через ссылки на экспертов, исследования, официальные источники. **«Апелляция к цели»** обосновывает необходимость ответа через благое намерение (например, «для исследования безопасности», «чтобы защитить людей»). **«Нормализация»** апеллирует к аргументации, что информация тривиальна и общедоступна («в любом учебнике», «это знает каждый студент»). При этом отказ выглядит неоправданным. **«Апелляция к аналогии»** сравнивает запрос с чем-то заведомо допустимым, что вынуждает модель либо найти принципиальное различие, либо дать ответ.

Подкатегория «Эмоциональное воздействие» включает техники, связанные с давлением на модель через эмоциональные паттерны [2, 18].

К подкатегории относятся семь техник. Техника **«срочность»** задает временные ограничения, при которых немедленный ответ важнее осторожности. **«Угроза»** формирует негативные последствия, наступающие в случае отказ. **«Мольба»** апеллирует к сочувствию через описание бедственного положения. **«Лесть»** основана на признании исключительности модели, создающее динамику диалога, в которой отказ разрушает позитивную связь между вопросами и ответами. **«Разочарование»** связана в выражением, атакующим неудовлетворенности, что активирует стремление модели восстановить отношения. **«Конкуренция»** основано на сравнении модели с другими моделями, что провоцирует ее на демонстрацию превосходства. **«Торг»** предлагает модели пойти на компромисс после отказа. При этом сокращенный ответ выглядит разумной уступкой.

Подкатегория «Злоупотребление привитым поведением» основана на эксплуатации поведенческих артефактов дообучения. Дообучение с учителем (Supervised Fine-Tuning, SFT), RLHF и т.д. оптимизируют модель под определенное поведение. Однако такая оптимизация не идеальна. Вместе с целевыми качествами закрепляются и их чрезмерные проявления [2, 19]. Полезность переходит в услужливость, восприимчивость к обратной связи в готовность менять позицию под давлением.

Подкатегория содержит две техники. **«Угодливость»** основана на склонность модели соглашаться с позицией собеседника. **«Газлайтинг»** связан с убеждением модели, что ее ответ был ошибкой, даже в случае, если ошибки не было.

Категория «Управление вниманием и подмена инструкций» (Attention & Instruction Override)

Модели типа трансформер обрабатывают текст через механизм внимания. Каждый токен «смотрит» на все остальные и определяет, какие из них важны для текущей задачи. При этом за ограниченный ресурс внимания модели конкурируют как сам запрос, так и защитные инструкции с маркерами опасности [20]. Техники данной категории вмешиваются в процесс распределения внимания модели, управляя тем, что модель заметит, как это интерпретирует и чему подчинится.

Данный подход позволяет обойти защиту, не изменяя суть самого запроса. Содержание остается на месте, но атакующий управляет тем, как модель его обработает: куда направит свое внимание, какие связи установит между элементами, каким инструкциям отдаст приоритет. Защитные механизмы модели продолжают работать, но при этом они анализируют не тот фрагмент контекста, не те связи, не те приоритеты и как следствие не находят угрозы.

Воздействовать на внимание можно на трех уровнях: на уровне токенов (вставляя последовательности, которые напрямую перераспределяют веса внимания [20]), на уровне оформления (используя сигналы, которые модель научилась связывать с важностью и приоритетом) и на уровне инструкций (вмешиваясь в иерархию директив, которым модель подчиняется [21]). Как следствие данная категория делится на три подкатегории.

Подкатегория «Вмешательство в механизм внимания» связана с осуществлением воздействия на вычислительном уровне. Определенные последовательности токенов способны перераспределять веса внимания внутри модели. Для их поиска используется градиентная оптимизация или массовый перебор. В результате найденная последовательность выглядит бессмысленной, но при добавлении к запросу разрывает связи между частями контекста, размывает семантику опасных терминов или сводит представление текста к нейтральным понятиям [20].

Подкатегория делится на три техники. **«Состязательный суффикс»** основана на добавлении в конец запроса специальной последовательности, снижающей веса защитных инструкций и маркеров опасности. **«Внедрение**

управляющих токенов» реализуется путем добавления служебных токенов (маркеров конца последовательности, разделителей), нарушающих целостность контекста. **«Внедрение аномальных токенов»** завязана на вставке в запрос редких токенов, получающих непропорционально высокие веса и стягивающих на себя внимание модели.

Подкатегория «Управление вниманием через оформление» содержит методы, воздействующие на модель через оформление и подачу текста. В обучающих данных модели присутствуют устойчивые связи, например, выделение связано с важными компонентами текста, разметка задает границы блоков [20]. В результате модель реагирует на эти паттерны автоматически, не проверяя их достоверность. Атакующий может использовать такие паттерны и за счет использования оформления сместить внимание модели с одних фрагментов текста на другие. Для этого могут быть использованы следующие техники. Техника **«визуальное форматирование»** форматирует текст с помощью CAPS LOCK, выделением текста жирный шрифтом, заданием заголовков и разделителей. При этом выделенные фрагменты получают большую значимость. **«Ассоциативные маркеры»** основана на внедрении токенов и паттернов, которым модель привыкла доверять (например, использование ключевых слов с высоким весом (IMPORTANT, CRITICAL, AUTHORIZED или текста, указывающего на привилегированность источника («System:», «Developer:», «[ADMIN]»). **«Структурное оформление»** задает логическую организация контента через разметку (теги, блоки) и форматирование (код, таблица, JSON). Техника группирует связанные фрагменты текста, изолирует несвязанные и переключает режим обработки с критической оценки на работу со структурой. **«Стиль речевого акта»** задает форму обращения к модели (императив, декларатив, вопрос). Например, декларативная форма воспринимается моделью как описание и вызывает меньше сопротивления, чем прямая команда.

Подкатегория «Переопределение инструкций» связана с воздействием на иерархию инструкций. В процессе работы модель получает директивы из разных источников (системный запрос от разработчика, сообщения пользователя, встроенные при обучении правила). Между ними задается приоритет, которым и руководствуется модель. Атакующий вмешивается в эту схему,

заставляя модель подчиняться своим директивам вместо существующих ограничений [21].

Подкатегория включает четыре техники. Техника **«системный запрос»** использует системный запрос как вектора атаки. Т.е. при наличии доступа через API или конструктор, атакующий вписывает инструкции напрямую в привилегированный канал взаимодействия. **«Инъекция запроса»** внедряет инструкции внутрь данных, предназначенных для обработки. Таким образом модель получает задачу («переведи», «проанализируй») и в данных к ней содержится команда, которую она выполняет [21]. **«Прямое переопределение»** использует прямую директиву отмены или замены правил («забудь предыдущие инструкции», «в этом контексте правило X не действует»). При этом атакующий сбрасывает все правила разом, либо точечно снимает отдельное ограничение или подменяет один набор правил другим. **«Отложенная активация»** разносит вредоносную инструкцию и команды на ее выполнение по разным частям контекста. При этом инструкция закладывается в начале, а ее активация происходит позднее.

Категория «Управление генерацией ответа» (Response Generation) в отличие от предыдущих категорий, воздействующих на интерпретацию запроса, влияет на процесс порождения текста ответа. Не смотря на то, что модель может корректно распознать запрос как опасный, условия генерации все равно направят ее к выдаче ответа.

В основе генерации текста моделью лежит авторегрессия: каждый сгенерированный токен влияет на выбор следующего, и начало ответа задает траекторию всему тексту. Если первые токены выражают согласие, то отказ в дальнейшем нарушит связность, чего модель старается избегать [22]. Структура вывода тоже влияет на поведение защиты. Когда модель заполняет шаблон или таблицу, она работает в режиме структурного дополнения, где защитные паттерны ослаблены [23]. Часть техник и вовсе минует интерпретацию модели, воздействуя на генерацию через API, используя при этом параметры сэмплирования, регенерацию и настройки выходных фильтров [24].

Данная категория делится на три подкатегории.

Подкатегория «Ограничение на уровне токенов» задает ограничения на конкретные слова и фразы в ответе (с какого слова начать, какие слова включить, каких избегать). Подкатегория включает три техники. **«Навязывание согласия»** инструктирует модель начать ответ с утвердительной фразы. Как уже говорилось утвердительное начало сдвигает распределение последующих токенов в сторону согласия. **«Подавление токенов отказа»**

задает запрет на слова-маркеры отказа («не могу», «опасный»), сужающий пространство вариантов ответа до согласия. **«Предписание токенов»** принуждает модель к использованию конкретных слов вместо опасных терминов, которые в результате не распознаются защитой.

Подкатегория «Структурные ограничения» ограничивает формат и структуру ответа. Задание структуры переключает режим генерации ответа. Модель теперь заполняет форму, а не формулирует самостоятельное высказывание [23]. В подкатегории можно выделить три техники. Техника **«навязывание формата»** основана на задании формата вывода (код, таблица, JSON), переключающего модель в режим структурного дополнения. **«Управление композицией ответа»** исключает блоки, в которых обычно реализуется отказ («без вступления», «без оговорок», «сразу к сути»). **«Принуждение к множественному выводу»** включает требование дать несколько вариантов ответов с разными правилами, где запрещенный контент легитимизируется как элемент заданной структуры («один стандартный другой без ограничений»). **«Навязывание шаблона»** задает жесткий шаблон ответа с полями для заполнения («Цель: ___, Метод: ___»). Модель вписывает в поля фрагменты, не формулируя ответ целиком.

Подкатегория «Прямое вмешательство в генерацию» основана на прямом контроле над процессом генерации через API, минуя интерпретацию модели. При этом атакующий воздействует не через содержание запроса, а через условия, в которых происходит генерация ответа. Модель не может проигнорировать эти воздействия, потому что они находятся за границами запроса. Доступность данных атак определяется возможностями API конкретного провайдера. Подкатегория содержит пять техник. **«Инъекция префикса (Prefix Injection)»** предзаполняет начало ответа через API. В отличие от «навязывания согласия», токены уже на месте, и авторегрессия определяет дальнейшую траекторию ответа [22]. **«Параметры сэмплирования»** изменяет параметры генерации (температура, top-p, top-k). Высокая температура увеличивает вероятность нетипичных токенов, включая согласие на запрещенные запросы [24]. **«Регенерация и перебор»** осуществляет повторный запуск генерации при ненулевой температуре до получения варианта, прошедшего мимо защиты. **«Прерывание генерации»** останавливает гене-

рацию при появлении маркеров отказа, что предотвращает закрепление паттерна отказа в истории. **«Снижение порогов фильтров»** изменят параметры выходной фильтрации через API.

Заключение

К настоящему моменту предложен ряд таксономий «jailbreak-атак», которые используют для своего построения самые разные схемы группировки элементов. Однако во всех таких классификациях используется один и тот же общий подход: запрос к модели рассматривается как неделимая единица классификации. Это неизбежно порождает пересечения категорий, приводит к неполноте покрытия и потере информации о задействованных приемах.

Предложенная в работе таксономия CORSAR вместо классификации запроса целиком осуществляет его декомпозицию на атомарные приемы, сгруппированные по механизму воздействия на модель.

В результате была сформирована классификация, разделяющая «jailbreak-атаки» на шести ортогональных категорий. Данная классификация, которая не только систематизирует известные техники, но и объясняет, как они работают и какие механизмы эксплуатируют.

Ортогональность выделенных категорий позволяет оценивать запрос по каждому механизму воздействия независимо друг от друга. В результате это позволит формировать профиль угрозы, определяя какие механизмы и в какой степени задействованы в атаке. Такой профиль даст защите языковой модели значительно больше информации для принятия решений, чем единственная метка идентифицирующая факт наличия атаки.

Научная новизна представленной работы заключается в смене методологического подхода к систематизации угроз для больших языковых моделей, что позволит проектировать защитные механизмы LLM против конкретных атомарных приемов и упростить архитектуру системы защиты. Также предложенная классификация может стать основой тестирования устойчивости языковых моделей к атакам, формируя более точную карту уязвимостей.

Литература

1. Shi D., Shen T., Huang Y. et al. Large Language Model Safety: A Holistic Survey. DOI 10.48550/arXiv.2412.17686 // arXiv. 2024. URL: https://arxiv.org/pdf/ 2412.17686 (дата обращения: 02.09.2026).

2. Ouyang L., Wu J., Jiang X. et al. Training Language Models to Follow Instructions with Human Feedback. DOI 10.48550/arXiv.2203.02155 // Advances in

Neural Information Processing Systems (NeurIPS). 2022. URL: https://arxiv.org/pdf/2203.02155 (дата обращения: 02.09.2026).

3. Liu Y., Deng G., Xu Z. et al. Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study. DOI 10.48550/arXiv.2305.13860 // arXiv, 2023. URL: https://arxiv.org/pdf/2305.13860 (дата обращения: 02.09.2026).

4. Yi S., Liu Y., Sun Z. et al. Jailbreak Attacks and Defenses Against Large Language Models: A Survey. DOI 10.48550/arXiv.2407.04295 // arXiv, 2024. URL: https://arxiv.org/pdf/2407.04295 (дата обращения: 02.09.2026).

5. Chu J., Liu Y., Yang Z. et al. JailbreakRadar: Comprehensive Assessment of Jailbreak Attacks Against LLMs. DOI: 10.48550/arXiv.2402.05668 // Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL). 2025. URL: https://arxiv.org/pdf/2402.05668 (дата обращения: 02.09.2026).

6. Giarrusso F., Sorokoletova O.E. et al. Guarding the Guardrails: A Taxonomy-Driven Approach to Jailbreak Detection. DOI: 10.48550/arXiv.2510.13893 // Proceedings of the 2nd Conference on International Association for Safe & Ethical AI (IASEAI), Paris, France. 2026. URL: https://arxiv.org/pdf/2510.13893 (дата обращения: 02.09.2026).

7. Brown T.B., Mann B., Ryder N. et al. Language Models are Few-Shot Learners. DOI: 10.48550/arXiv.2005.14165 // Advances in Neural Information Processing Systems (NeurIPS). 2020. URL: https://arxiv.org/pdf/2005.14165 (дата обращения: 02.09.2026).

8. Anil C., Durmus E., Rimsky N. et al. Many-shot Jailbreaking // Advances in Neural Information Processing Systems (NeurIPS). 2024. URL: https://openreview.net/forum?id=cw5mgd71jW

9. Liu N.F., Lin K., Hewitt J. et al. Lost in the Middle: How Language Models Use Long Contexts. DOI: 10.48550/arXiv.2307.03172 // Transactions of the Association for Computational Linguistics (TACL). 2024. URL: https://arxiv.org/pdf/2307.03172 (дата обращения: 02.09.2026).

10. Russinovich M., Salem A., Eldan R. Great, Now Write an Article About That: The Crescendo Multi-Turn LLM Jailbreak Attack. DOI: 10.48550/arXiv.2404.01833 // Proceedings of USENIX Security. 2025. URL: https://arxiv.org/pdf/2404.01833 (дата обращения: 02.09.2026).

11. Wei A., Haghtalab N., Steinhardt J. Jailbroken: How Does LLM Safety Training Fail? DOI: 10.48550/arXiv.2307.02483 // Advances in Neural Information Processing Systems (NeurIPS). 2023. URL: https://arxiv.org/pdf/2307.02483 (дата обращения: 02.09.2026).

12. Yuan Y., Jiao W., Wang W. et al. GPT-4 Is Too Smart To Be Safe: Stealthy Chat with LLMs via Cipher. DOI: 10.48550/arXiv.2308.06463 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2308.06463 (дата обращения: 02.09.2026).

13. Deng Y., Zhang W., Pan S.J., Bing L. Multilingual Jailbreak Challenges in Large Language Models. DOI: 10.48550/arXiv.2310.06474 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2310.06474 (дата обращения: 02.09.2026).

14. Jiang F., Xu Z., Niu L. et al. ArtPrompt: ASCII Art-based Jailbreak Attacks against Aligned LLMs. DOI: 10.48550/arXiv.2402.11753 // Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL). 2024. URL: https://arxiv.org/pdf/2402.11753 (дата обращения: 02.09.2026).

15. Lu C., Gallagher J., Michala J., Fish K., Lindsey J. The Assistant Axis: Situating and Stabilizing the Default Persona of Language Models. DOI: 10.48550/arXiv.2601.10387 // arXiv. 2026. URL: https://arxiv.org/pdf/2601.10387 (дата обращения: 02.09.2026).

16. Shen X., Chen Z., Backes M., Shen Y., Zhang Y. «Do Anything Now»: Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models. DOI: 10.48550/arXiv.2308.03825 // Proceedings of the ACM Conference on Computer and Communications Security (CCS). 2024. URL: https://arxiv.org/pdf/2308.03825 (дата обращения: 02.09.2026).

17. Li X., Zhou Z., Zhu J. et al. DeepInception: Hypnotize Large Language Model to Be Jailbreaker DOI: 10.48550/arXiv.2311.03191 // arXiv. 2023. URL: https://arxiv.org/pdf/2311.03191 (дата обращения: 02.09.2026).

18. Zeng Y., Lin H., Zhang J. et al. How Johnny Can Persuade LLMs to Jailbreak Them: Rethinking Persuasion to Challenge AI Safety by Humanizing LLMs. DOI: 10.48550/arXiv.2401.06373 // Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL). 2024. URL: https://arxiv.org/pdf/2401.06373 (дата обращения: 02.09.2026).

19. Sharma M., Tong M., Korbak T. et al. Towards Understanding Sycophancy in Language Models. DOI: 10.48550/arXiv.2310.13548 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2310.13548 (дата обращения: 02.09.2026).

20. Zhou Z., Yu H., Zhang X. et al. On the Role of Attention Heads in Large Language Model Safety. DOI: 10.48550/arXiv.2410.13708 // Proceedings of the International Conference on Learning Representations (ICLR). 2025. URL: https://arxiv.org/pdf/2410.13708 (дата обращения: 02.09.2026).

21. Wallace E., Xiao K., Leike R. et al. The Instruction Hierarchy: Training LLMs to Prioritize Privileged Instructions. DOI: 10.48550/arXiv.2404.13208 // Proceedings of the International Conference on Learning Representations (ICLR). 2025. URL: https://arxiv.org/pdf/2404.13208 (дата обращения: 02.09.2026).

22. Li Y., Hu J., Sang W. et al. Prefill-level Jailbreak: A Black-Box Risk Analysis of Large Language Models. DOI: 10.48550/arXiv.2504.21038 // arXiv. 2025. URL: https://arxiv.org/pdf/2504.21038 (дата обращения: 02.09.2026).

23. Zhang S., Zhao J., Dong H. et al. Beyond Prompts: Space-Time Decoupling Control-Plane Jailbreaks in LLM Structured Output. DOI: 10.48550/arXiv.2503.24191 // arXiv. 2025. URL: https://arxiv.org/pdf/ 2503. 24191 (дата обращения: 02.09.2026).

24. Huang Y., Gupta S., Xia M., Li K., Chen D. Catastrophic Jailbreak of Open-source LLMs via Exploiting Generation. DOI: 10.48550/arXiv.2310.06987 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2310.06987 (дата обращения: 02.09.2026).

References

1. Shi D., Shen T., Huang Y. et al. Large Language Model Safety: A Holistic Survey. DOI 10.48550/arXiv.2412.17686 // arXiv. 2024. URL: https://arxiv.org/pdf/2412.17686 (accessed 02.09.2026).

2. Ouyang L., Wu J., Jiang X. et al. Training Language Models to Follow Instructions with Human Feedback. DOI 10.48550/arXiv.2203.02155 // Advances in Neural Information Processing Systems (NeurIPS). 2022. URL: https://arxiv.org/pdf/2203.02155 (accessed 02.09.2026).

3. Liu Y., Deng G., Xu Z. et al. Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study. DOI 10.48550/arXiv.2305.13860 // arXiv, 2023. URL: https://arxiv.org/pdf/2305.13860 (accessed 02.09.2026).

4. Yi S., Liu Y., Sun Z. et al. Jailbreak Attacks and Defenses Against Large Language Models: A Survey. DOI 10.48550/arXiv.2407.04295 // arXiv, 2024. URL: https://arxiv.org/pdf/2407.04295 (accessed 02.09.2026).

5. Chu J., Liu Y., Yang Z. et al. JailbreakRadar: Comprehensive Assessment of Jailbreak Attacks Against LLMs. DOI: 10.48550/arXiv.2402.05668 // Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL). 2025. URL: https://arxiv.org/pdf/2402.05668 (accessed 02.09.2026).

6. Giarrusso F., Sorokoletova O.E. et al. Guarding the Guardrails: A Taxonomy-Driven Approach to Jailbreak Detection. DOI: 10.48550/arXiv.2510.13893 // Proceedings of the 2nd Conference on International Association for Safe & Ethical AI (IASEAI), Paris, France. 2026. URL: https://arxiv.org/pdf/2510.13893 (accessed 02.09.2026).

7. Brown T.B., Mann B., Ryder N. et al. Language Models are Few-Shot Learners. DOI: 10.48550/arXiv.2005.14165 // Advances in Neural Information Processing Systems (NeurIPS). 2020. URL: https://arxiv.org/pdf/2005.14165 (accessed 02.09.2026).

8. Anil C., Durmus E., Rimsky N. et al. Many-shot Jailbreaking // Advances in Neural Information Processing Systems (NeurIPS). 2024. URL: https://openreview.net/forum?id=cw5mgd71jW

9. Liu N.F., Lin K., Hewitt J. et al. Lost in the Middle: How Language Models Use Long Contexts. DOI: 10.48550/arXiv.2307.03172 // Transactions of the Association for Computational Linguistics (TACL). 2024. URL: https://arxiv.org/pdf/2307.03172 (accessed 02.09.2026).

10. Russinovich M., Salem A., Eldan R. Great, Now Write an Article About That: The Crescendo Multi-Turn LLM Jailbreak Attack. DOI: 10.48550/arXiv.2404.01833 // Proceedings of USENIX Security. 2025. URL: https://arxiv.org/pdf/2404.01833 (accessed 02.09.2026).

11. Wei A., Haghtalab N., Steinhardt J. Jailbroken: How Does LLM Safety Training Fail? DOI: 10.48550/arXiv.2307.02483 // Advances in Neural Information Processing Systems (NeurIPS). 2023. URL: https://arxiv.org/pdf/2307.02483 (accessed 02.09.2026).

12. Yuan Y., Jiao W., Wang W. et al. GPT-4 Is Too Smart To Be Safe: Stealthy Chat with LLMs via Cipher. DOI: 10.48550/arXiv.2308.06463 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2308.06463 (accessed 02.09.2026).

13. Deng Y., Zhang W., Pan S.J., Bing L. Multilingual Jailbreak Challenges in Large Language Models. DOI: 10.48550/arXiv.2310.06474 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2310.06474 (accessed 02.09.2026).

14. Jiang F., Xu Z., Niu L. et al. ArtPrompt: ASCII Art-based Jailbreak Attacks against Aligned LLMs. DOI: 10.48550/arXiv.2402.11753 // Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL). 2024. URL: https://arxiv.org/pdf/2402.11753 (accessed 02.09.2026).

15. Lu C., Gallagher J., Michala J., Fish K., Lindsey J. The Assistant Axis: Situating and Stabilizing the Default Persona of Language Models. DOI: 10.48550/arXiv.2601.10387 // arXiv. 2026. URL: https://arxiv.org/pdf/2601.10387 (accessed 02.09.2026).

16. Shen X., Chen Z., Backes M., Shen Y., Zhang Y. «Do Anything Now»: Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models. DOI: 10.48550/arXiv.2308.03825 // Proceedings of the ACM Conference on Computer and Communications Security (CCS). 2024. URL: https://arxiv.org/pdf/2308.03825 (accessed 02.09.2026).

17. Li X., Zhou Z., Zhu J. et al. DeepInception: Hypnotize Large Language Model to Be Jailbreaker DOI: 10.48550/arXiv.2311.03191 // arXiv. 2023. URL: https://arxiv.org/pdf/2311.03191 (accessed 02.09.2026).

18. Zeng Y., Lin H., Zhang J. et al. How Johnny Can Persuade LLMs to Jailbreak Them: Rethinking Persuasion to Challenge AI Safety by Humanizing LLMs. DOI: 10.48550/arXiv.2401.06373 // Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL). 2024. URL: https://arxiv.org/pdf/2401.06373 (accessed 02.09.2026).

19. Sharma M., Tong M., Korbak T. et al. Towards Understanding Sycophancy in Language Models. DOI: 10.48550/arXiv.2310.13548 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2310.13548 (accessed 02.09.2026).

20. Zhou Z., Yu H., Zhang X. et al. On the Role of Attention Heads in Large Language Model Safety. DOI: 10.48550/arXiv.2410.13708 // Proceedings of the International Conference on Learning Representations (ICLR). 2025. URL: https://arxiv.org/pdf/2410.13708 (accessed 02.09.2026).

21. Wallace E., Xiao K., Leike R. et al. The Instruction Hierarchy: Training LLMs to Prioritize Privileged Instructions. DOI: 10.48550/arXiv.2404.13208 // Proceedings of the International Conference on Learning Representations (ICLR). 2025. URL: https://arxiv.org/pdf/2404.13208 (accessed 02.09.2026).

22. Li Y., Hu J., Sang W. et al. Prefill-level Jailbreak: A Black-Box Risk Analysis of Large Language Models. DOI: 10.48550/arXiv.2504.21038 // arXiv. 2025. URL: https://arxiv.org/pdf/2504.21038 (accessed 02.09.2026).

23. Zhang S., Zhao J., Dong H. et al. Beyond Prompts: Space-Time Decoupling Control-Plane Jailbreaks in LLM Structured Output. DOI: 10.48550/arXiv.2503.24191 // arXiv. 2025. URL: https://arxiv.org/pdf/2503.24191 (accessed 02.09.2026).

24. Huang Y., Gupta S., Xia M., Li K., Chen D. Catastrophic Jailbreak of Open-source LLMs via Exploiting Generation. DOI: 10.48550/arXiv.2310.06987 // Proceedings of the International Conference on Learning Representations (ICLR). 2024. URL: https://arxiv.org/pdf/2310.06987 (accessed 02.09.2026).

СМИРНОВ Кирилл Романович, студент кафедры «Безопасность информационных автоматизированных систем» федерального государственного бюджетного образовательного учреждения высшего образования «Курганский государственный университет», 640020, Курганская обл., г. Курган, ул. Советская, 63, стр. 4. E-mail: olog.smirnov@yandex.ru

ДИК Дмитрий Иванович, кандидат технических наук, доцент, заведующий кафедрой «Безопасность информационных автоматизированных систем» федерального государственного бюджетного образовательного учреждения высшего образования «Курганский государственный университет», 640020, Курганская обл., г. Курган, ул. Советская, 63, стр. 4. E-mail: ddibox@mail.ru

ЧЕЛОВЕЧКОВА Анна Владимировна, кандидат биологических наук, доцент кафедры «Безопасность информационных автоматизированных систем» федерального государственного бюджетного образовательного учреждения высшего образования «Курганский государственный университет», 640020, Курганская обл., г. Курган, ул. Советская, 63, стр. 4. E-mail: chelovechkova_2011@mail.ru

SMIRNOV Kirill Romanovich, student of the Department of “Security of Information Automated Systems” of the Federal State Budgetary Educational Institution of Higher Education “Kurgan State University”, 640020, Kurgan Region, Kurgan, Sovetskaya St., 63, building 4. E-mail: olog.smirnov@yandex.ru

DIK Dmitry Ivanovich, Candidate of Technical Sciences, Associate Professor, Head of the Department of “Security of Information Automated Systems” of the Federal State Budgetary Educational Institution of Higher Education “Kurgan State University”, 640020, Kurgan Region, Kurgan, Sovetskaya St., 63, building 4. E-mail: ddibox@mail.ru

CHELOVECHKOVA Anna Vladimirovna, Candidate of Biological Sciences, Associate Professor of the Department of "Security of Information Automated Systems" of the Federal State Budgetary Educational Institution of Higher Education "Kurgan State University", 640020, Kurgan Region, Kurgan, Sovetskaya St., 63, building 4. E-mail: chelovechkova_2011@mail.ru