Министерство науки и высшего образования Российской Федерации
Московский физико-технический институт
(национальный исследовательский университет)
Физтех-школа прикладной математики и информатики
Магистерская программа
«Методы и технологии искусственного интеллекта»

Беликова Юлия Андреевна

Применение white-box подходов для улучшения вопросно-ответных навыков в больших языковых моделях

03.04.01 — Прикладные математика и физика

Магистерская диссертация

Научный руководитель:
старший научный сотрудник, ктн
Коновалов Василий Павлович

Консультант:
ассистент
Глинский Андрей Владимирович

Москва

2025 г.

Содержание

1 Введение **4**

2 Основные понятия и определения **7**

3 Обзор современного состояния проблемы **10**

3.1 Методы детекции контекстуальных галлюцинаций 11

3.2 Методы минимизации контекстуальных галлюцинаций 16

4 Методология минимизации контекстуальных галлюцинаций **21**

4.1 Формальная постановка задачи 21

4.2 Архитектура решения 22

4.2.1 Необучаемая детекция контекстуальных галлюцинаций на основе топологического анализа и оценки неопределенности 23

4.2.2 Обучаемая детекция контекстуальных галлюцинаций методами пробинга 25

4.2.3 Улучшение генерации с помощью детекции контекстуальных галлюцинаций 26

5 Вычислительные эксперименты **28**

5.1 Детекция контекстуальных галлюцинаций 28

5.1.1 Постановка экспериментов 28

5.1.2 Описание результатов 30

5.1.3 Выводы 33

5.2 Минимизация контекстуальных галлюцинаций 34

5.2.1 Постановка экспериментов 34

5.2.2 Описание результатов 35

5.2.3 Выводы 38

6 Заключение **40**

Аннотация

Одно из наиболее популярных применений больших языковых моделей в индустрии – вопросно-ответные системы, где каждый ответ опирается на некоторую актуальную информацию из базы знаний. Критическим ограничением таких систем является наличие контекстуальных галлюцинаций – ситуаций, когда большая языковая модель генерирует информацию, не соответствующую предоставленному из источника знаний контексту. Данная работа предлагает комплексную методологию детекции и минимизации галлюцинаций, основанную на white-box подходах, использующих внутренние состояния модели для детекции и минимизации галлюцинаций. Разработаны два метода детекции: необучаемый, ансамблирующий топологический анализ карт внимания с метриками потокенной неопределенности, и обучаемый на основе методов пробинга, адаптированный для работы с ограниченным количеством обучающих данных. Для минимизации галлюцинаций предложены методы управляемого декодирования с выбором оптимальной генерации и дообучения модели с автоматическим формированием пар предпочтений. Экспериментальная оценка на стандартных наборах данных вопросно-ответных задач продемонстрировала высокую эффективность предложенных подходов, которые показали значительное улучшение фактологической точности ответов по сравнению с базовыми методами.

1 Введение

Большие языковые модели (англ. large language models, LLMs) в последние годы достигли огромной популярности и нашли широкое применение во многих задачах обработки текста. Особенно заметна их роль в системах вопросно-ответного (англ. question-answering, QA) поиска знаний, где модель генерирует ответ на вопрос пользователя на основе большого объёма обучающих данных. Современные LLM, такие как GPT-4 [33] и DeepSeek-V3 [45], способны генерировать связные и осмысленные ответы, и в комбинации с извлечением внешней информации из баз знаний в RAG (Retrieval Augmented Generation) системах они позволяют отвечать на запросы с опорой на актуальные данные. Такие подходы активно применяются в практике – от пользовательских чат-ботов (например, ChatGPT) до специализированных помощников в медицине и юриспруденции.

Однако, несмотря на успехи, ключевой проблемой LLM остаётся феномен галлюцинаций – генерации правдоподобных, но фактически неверных или несоответствующих контексту утверждений. В особенной степени это проявляется в RAG-сценариях, когда модель получает контекст (набор документов или фактов) для ответа: иногда вместо строгой опоры на предоставленные сведения модель добавляет несуществующие детали или противоречит данным контекста. Такие ошибки называют *контекстуальными галлюцинациями* (англ. contextual hallucinations), и они существенно подрывают доверие к системе в целом. Например, в медицинском QA-сервисе галлюцинация в ответе может привести к неправильному совету пациенту, что недопустимо с точки зрения надёжности. По данным ряда исследований, проблема галлюцинаций LLM носит массовый характер и наблюдается даже у передовых (англ. state-of-the-art, SOTA) моделей в большинстве информационно-насыщенных сценариев использования [43]. Поэтому задача обнаружения и минимизации галлюцинаций признана одной из наиболее актуальных для повышения фактической корректности и безопасности систем на базе больших языковых моделей [22].

Существующие подходы к борьбе с галлюцинациями можно условно разделить на так называемые *методы «чёрного ящика» (англ. black-box)* и *«белого ящика» (англ. white-box)*. Black-box методы не требуют доступа к внутреннему состоянию модели, рассматривая LLM как чёрный ящик с точки зрения пользователя. К таким подходам относятся, например, внешняя проверка фактов дополнительной моделью, оценка уверенности модели на основе стабильности ответа при повторной генерации,

а также специальные схемы запросов (англ. prompting), заставляющие модель самопроверять свой ответ (англ. self-refinement). В противоположность им, white-box методы опираются на знание внутренней структуры модели и её состояний: они могут анализировать скрытые представления, такие как *активации нейронных слоёв* (англ. neurons activations), *карты внимания* (англ. attention maps), изменять ход декодирования с использованием внутренних параметров или даже корректировать скрытые состояния модели, чтобы предотвратить появление недостоверной информации. Оба подхода находят применение: black-box методы удобны тем, что применимы даже к закрытым моделям, доступным по API (англ. application programming interface), однако большинство таких подходов требуют высоких вычислительних и временных затрат, тогда как white-box методы более ресурсоэффективные, они учитывают причины галлюцинаций внутри самой модели [13], а также подходят для коммерческого применения в условиях конфиденциальности данных. Настоящая работа посвящена исследованию и применению white-box подходов для улучшения точности и надежности ответов LLM в контекстно обоснованных QA-системах.

Цель работы – разработать методологию, использующую информацию о внутреннем состоянии и параметрах модели, для детекции контекстуальных галлюцинаций и снижения их количества при генерации ответов. Для достижения поставленной цели разработана двухкомпонентная методология. Первый компонент – детекция галлюцинаций – реализован в двух вариантах: (i) необучаемый (англ. unsupervised) подход, основанный на ансамблировании топологического анализа карт внимания с метриками потокенной неопределенности, что позволяет выявлять галлюцинации без предварительного обучения; (ii) обучаемый (англ. supervised) подход с использованием методов пробинга внутренних состояний модели, адаптированный для работы с ограниченным количеством обучающих данных через применение снижения размерности и специализированных классификаторов. Второй компонент – минимизация галлюцинаций – включает: (i) метод управляемого декодирования с генерацией множества кандидатов и выбором оптимального ответа на основе детектора галлюцинаций; (ii) дообучение модели методом Direct Preference Optimization (DPO) [15] на автоматически сформированных парах предпочтений, где предпочтение определяется разработанными детекторами. Ключевым преимуществом предложенного подхода является его автономность – отсутствие зависимости от проприетарных моделей или ручной разметки данных при сохранении высокой эффективности детекции и минимизации галлюцинаций.

Диссертация организована следующим образом. В разделе 2 вводятся основные понятия и определения, включая формальное описание больших языковых моделей, RAG-систем, типов контекстуальных галлюцинаций и различий между white-box и black-box подходами. Раздел 3 содержит обзор современного состояния проблемы, систематизируя существующие методы детекции и минимизации галлюцинаций с анализом их преимуществ и ограничений. В разделе 4 представлена предложенная методология, включающая формальную постановку задачи, архитектуру решения и детальное описание разработанных методов. Раздел 5 посвящен экспериментальной валидации предложенных подходов на стандартных и индустриальных наборах данных с анализом полученных результатов. В заключении подводятся итоги работы, обсуждается научная и практическая значимость результатов, а также рассматриваются направления дальнейших исследований.

2 Основные понятия и определения

Определение 2.1 (Большая языковая модель). *Большая языковая модель (англ. Large Language Model, LLM) – это параметрическая нейросетевая модель для генерации текста, построенная на архитектуре трансформера [1] и отличающаяся чрезвычайно большим числом параметров, обученных на масштабных корпусах текстовых данных. Основные компоненты архитектуры большой языковой модели:*

1. *кодирование: на вход LLM принимает текст, представленный в виде последовательности токенов, полученных с помощью алгоритмов токенизации. Каждый токен преобразуется в вектор фиксированной размерности d_{model} посредством обучаемого слоя эмбеддингов $E \in \mathbb{R}^{|V| \times d_{model}}$, где $|V|$ – размер словаря. К эмбеддингам токенов добавляется позиционное кодирование для сохранения информации о порядке токенов в последовательности (отметим, что в последних архитектурах позиция кодируется внутри блоков трансформера);*

2. *блоки трансформера: каждый блок состоит из многоголового механизма внимания (англ. Multi-Head Attention) и полносвязной нейросети с остаточными связями (англ. residual connections) и нормализацией слоев. Многоголовое внимание использует h параллельных голов внимания, каждая из которых обрабатывает различные аспекты зависимостей между токенами:*

$$\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h) W^O, \tag{2.1}$$

где $\mathrm{head}_i = \mathrm{Attention}(QW_i^Q, KW_i^K, VW_i^V)$, *а механизм внимания формально определяется как:*

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V, \tag{2.2}$$

где $Q = XW^Q$, $K = XW^K$, $V = XW^V$ *– матрицы запросов, ключей и значений, полученные линейным преобразованием входных эмбеддингов,* $d_k = d_{model}/h$ *– размерность каждой головы. Маскирование обеспечивает автогрессию – каждый токен* y_t *зависит только от предыдущих токенов* $y_{<t}$ *и входа* x*;*

3. *декодирование: в генеративной постановке LLM реализует отображение* $(x, y_{<t}) \mapsto P_\Theta(y_t \mid x, y_{<t})$*, то есть по входу и ранее сгенерированным токенам предсказывает следующий токен, что позволяет поэтапно строить выходную последовательность.*

Таким образом, LLM параметризует автогрессионное распределение:

$$P_\Theta(y \mid x) = \prod_{t=1}^{\ell} P_\Theta(y_t \mid x, y_{<t}), \tag{2.3}$$

где генерация каждого токена реализуется посредством прохода через стек блоков трансформера с маскированным вниманием и последующей проекцией на пространство словаря. Обучение проводится максимизацией правдоподобия.

Определение 2.2 (Вопросно-ответная система). *Вопросно-ответная система (англ. Question Answering System, QA) на основе больших языковых моделей – это система, которая принимает на вход вопрос q на естественном языке и опционально контекст c, и генерирует ответ a на естественном языке с использованием LLM.*

Определение 2.3 (Генерация с дополненной выборкой). *Генерация с дополненной выборкой (англ. Retrieval Augmented Generation, RAG) – это подход, комбинирующий LLM с внешней системой извлечения знаний [8]. В RAG-системе для заданного вопроса q сначала производится поиск релевантных документов $D = \{d_i\}$ во внешнем источнике (база знаний, интернет и т. д.), после чего LLM генерирует ответ a на основе расширенного контекста, объединяющего q и найденные данные D. На этапе augmented generation LLM принимает на вход как сам вопрос, так и фрагменты из D, благодаря чему ответ опирается на актуальные факты из документов.*

Отметим, что хотя RAG значительно повышает долю правильных ответов, LLM всё ещё могут порождать несоответствия или вымышленные факты, даже имея доступ к источникам. Проблема таких *контекстуальных галлюцинаций* рассматривается ниже.

Определение 2.4 (Контекстуальная галлюцинация). *Контекстуальной галлюцинацией называется случай, когда сгенерированный LLM ответ содержит информацию, не соответствующую предоставленному контексту (документам, вопросу). Формально, пусть дан входной контекст c (например, исходный вопрос и извлечённые документы) и моделью сгенерирован ответ r. Если r включает утверждение, которое либо (i) не поддерживается фактами из c, либо (ii) прямо противоречит информации в c, то ответ r считается галлюцинацией относительно контекста. Иными словами, модель либо вводит факт, не обоснованный входными данными, либо искажает заданные сведения. Такие галлюцинации выглядят правдоподобно, но при*

проверке не подтверждаются контекстом. В современных исследованиях надёжности LLM контекстуальные галлюцинации связывают с недостаточной faithfulness (точностью по отношению к источнику).

Для систематизации ошибок данного типа вводится таксономия контекстуальных галлюцинаций [40]. Ниже приведены основные категории с формальными критериями.

Определение 2.5 (Явный конфликт). *Ответ модели содержит утверждение, которое прямо и явно противоречит фактам из контекста. Такое несоответствие легко выявляется: утверждения r и c не могут одновременно быть истинными. Как правило, явный конфликт связан с грубой фактической ошибкой (неверное имя, дата, число и т. п.). Пример: исходный документ указывает, что «лучшие снимки получаются на сроке 24–30 недели беременности», тогда как модель в ответе утверждает «лучшие снимки получаются на сроке 20–32 недели» – напрямую отрицая данные источника.*

Определение 2.6 (Скрытый конфликт). *Ответ содержит скрытое противоречие: генерация модели отклоняется от смысла контекста, хотя явного логического конфликта может не быть. Информация из r видоизменяет утверждения из c, создавая более тонкое несоответствие. Обычно это происходит через подмену терминов или атрибутов, что меняет степень или нюанс исходного факта. Пример: в документе сказано «препарат не рекомендован детям», а модель отвечает «препарат можно применять для детей с осторожностью» – явного отрицания нет, но искажается нормативный смысл (требуется понимание контекста для обнаружения противоречия).*

Определение 2.7 (Явное добавление). *Модель явно добавляет в ответ информацию, которой нет во входном контексте. В ответе появляются факты или детали, не имеющие источника в c, и это легко проверить. Как правило, такие галлюцинации выражаются во введении вымышленных конкретных сведений (несуществующие даты, имена, количественные данные и пр.). Они не противоречат с напрямую, но являются безосновательными – не подкреплёнными ни текстом вопроса, ни документами. Пример: на вопрос об основании компании, в тексте нет даты, но модель уверенно отвечает: «Компания основана в 1985 году», хотя эта дата отсутствует в источниках (галлюцинируя факт).*

Определение 2.8 (Скрытое добавление). *Ответ модели содержит скрытое добавление: новая информация выходит за рамки данных c, однако имеет завуалированный, подразумеваемый характер. Добавленные сведения могут выглядеть как логичные обобщения, предположения или общие знания, которые нельзя прямо ни подтвердить, ни опровергнуть по контексту. Такие галлюцинации труднее выявить, так как они не являют собой конкретный факт, а скорее расширяют ответ догадками. Пример: модель дополняет ответ своими рассуждениями: «Этот исторический факт повлёк значительные изменения в культуре», хотя контекст про такие последствия не упоминал. Утверждение звучит правдоподобно, не противоречит исходным данным, но и не основано на них прямо.*

Указанные типы можно сопоставить с понятиями *внутренней* и *внешней* галлюцинации [24], принятыми в ранних работах. Конфликты соответствуют *внутренним* (англ. intrinsic) галлюцинациям, где модель искажает исходный факт, а добавления – *внешним* (англ. extrinsic) галлюцинациям, где модель вводит сторонние сведения, отсутствующие в источнике. Предложенная выше таксономия уточняет эти классы, разделяя каждый по признаку явности или скрытости несоответствия.

Определение 2.9 (White-box подходы). *White-box подходы предполагают полный доступ к внутреннему состоянию и параметрам LLM во время её использования или дообучения. В таких подходах исследователь может извлекать и анализировать скрытые представления модели (активации нейронов, веса внимания и т. д.), а также напрямую изменять параметры модели.*

Определение 2.10 (Black-box подходы). *Black-box подходы рассматривают LLM как чёрный ящик, доступный только через интерфейс ввода-вывода без раскрытия внутренних параметров и состояний.*

3 Обзор современного состояния проблемы

В силу широкой применимости LLM в критически важных областях (медицина, финансы, образование), проблема галлюцинаций привлекает большое внимание исследователей. Были зафиксированы случаи, когда уверенно сформулированные, но ложные ответы от моделей вводили в заблуждение пользователей. Так, при использовании генеративной модели в медицинской диагностике описание симптомов,

«придуманное» моделью, может привести врача по неверному пути [49]. Это подчёркивает необходимость надёжных средств контроля фактичности ответов. Ниже представлен обзор основных направлений исследований, касающихся обнаружения (детекции) галлюцинаций и методов их уменьшения. При этом рассматриваются как black-box, так и white-box решения, предлагаемые в современной литературе.

3.1 Методы детекции контекстуальных галлюцинаций

Основная задача детекции – автоматическим образом определить, содержит ли сгенерированный моделью ответ некорректную или неподкреплённую фактами информацию. Формально, имея на вход тройку (q, c, r), где q – запрос пользователя, c – дополнительный контекст (история диалога или фрагмент документа – в случае RAG-системы), r – ответ большой языковой модели, нужно вывести бинарный признак $d \in 0, 1$, где $d = 1$ означает наличие контекстуальной галлюцинации (ответ недостоверен относительно c). В литературе предложен ряд подходов к этой задаче:

Оценка неопределённости модели. Один из простейших сигналов возможной галлюцинации – это низкая уверенность самой модели в сгенерированном ответе. Практически это можно выявить с помощью множественного семплирования или ансамблирования генерации. Метод SelfCheckGPT [20] реализует данную идею: он несколько раз (с разными параматрами семплирования) генерирует ответ на один вопрос и затем сравнивает полученные варианты (Рисунок 1). Если ответы существенно расходятся или противоречат друг другу, то, вероятно, ни один из них не надёжен – модель «не знает» правильного ответа и галлюцинирует. Такой подход относится к black-box и не требует внешней базы знаний. Также существуют статистические критерии неопределённости, основанные на энтропии распределения вероятностей следующего токена: если энтропия велика, модель выбирает слова наугад, и можно подозревать галлюцинацию. А в работе [19] вводится понятие семантической энтропии. В отличие от классической энтропии, которая оценивает лексическое разнообразие сгенерированных текстов, семантическая энтропия учитывает, что разные фразы могут выражать одну и ту же мысль (например, «Париж – столица Франции» и «Столицей Франции является Париж»). Для этого все ответы группируются по их смысловой эквивалентности, и вычисляется энтропия распределения вероятностей по этим группам, а не по отдельным текстам. Такой подход позволяет точнее оценивать

истинную неопределённость модели относительно различных возможных значений, а не только разнообразие формулировок. Еще один метод акцентирующий внимание именно на семантичском разнообразии – INSIDE [34]. Его идея заключается в оценке семантической самосогласованности множества ответов через их скрытые представления. Для каждого ответа извлекается эмбеддинг (для чего используется скрытое состояние последнего символа среднего слоя языковой модели), а затем вычисляются собственные числа ковариационной матрицы полученных эмбеддингов. Когда ответы близки по смыслу, их эмбеддинги оказываются сильно коррелированными, а при высоком разнообразии ответов «разброс» эмбеддингов растёт и собственные числа становятся большими.

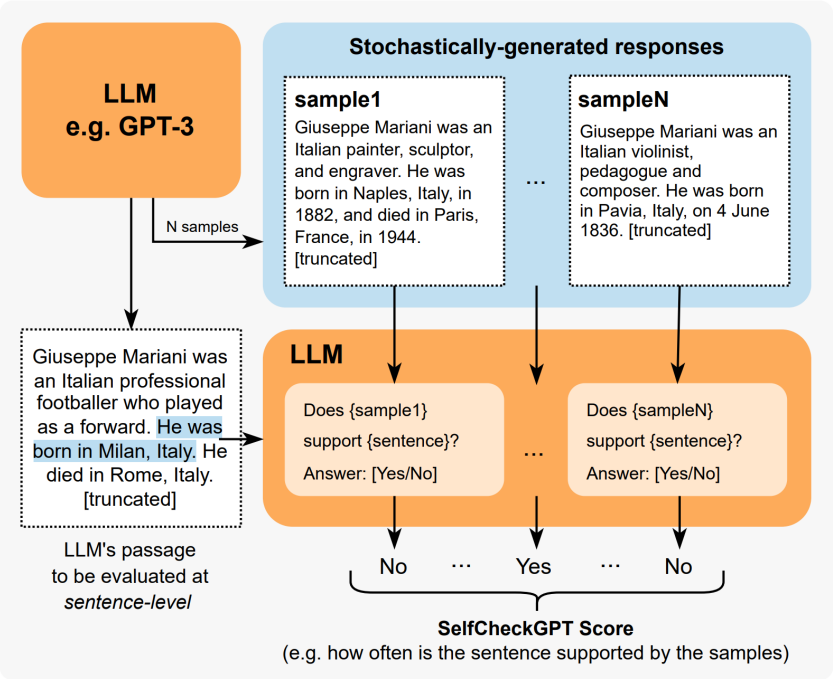

Рис. 1: SelfCheckGPT – сравнивнение сгенерированных предложений со стохастически сгенерированными ответами без взаимодействия со внешней базой данных.

Пробинг внутренних состояний. White-box методы детекции имеют целью распознавать галлюцинацию по активациям модели во время генерации. Например, показывается, что уже на этапе генерации последнего токена ответа сеть «решает», будет ли ответ правдивым или нет, что отражается в векторах скрытого состояния. Так, в работе [13] авторы обучали дополнительный классификатор, который на вход получает внутреннее представление LLM (вектор из последнего слоя перед генера-

цией ответа) и выдаёт оценку достоверности ответа. Данный подход получил дальнейшее развитие в работе [28], где предлагается метод *attention pooling probe*. Вместо использования только последнего токена, этот подход агрегирует информацию из всех скрытых состояний последовательности с помощью механизма внимания. Конкретно, для последовательности скрытых состояний $H = (h_1, \ldots, h_n)$ вычисляется взвешенная сумма:

$$h_{\text{pooled}} = \sum_{i=1}^{n} \alpha_i h_i, \tag{3.1}$$

где веса внимания α_i обучаются совместно с классификатором. Такой подход позволяет модели автоматически фокусироваться на наиболее информативных частях последовательности для детекции галлюцинаций. Экспериментальные результаты показывают, что attention pooling probe превосходит простые методы на основе последнего токена, особенно на задачах с длинными контекстами, где информация о достоверности может быть распределена по всей последовательности.

Оценка влияния контекста. Другой класс white-box методов основывается на анализе «внимания» модели ко входному контексту. Если LLM игнорирует релевантный контекст и «уходит» в автономную генерацию, вероятно появление контекстуальной галлюцинации. Для формального контроля такого поведения был предложен метод Lookback Lens[35], идея которого заключается в расчёте для каждого шага генерации соотношения внимания, которое модель уделяет уже сгенерированному ею тексту, по сравнению с вниманием на исходный контекст (источник знаний). Вычисляется так называемый *коэффициент обратного взгляда* (англ. *lookback ratio*) для каждого слоя и головы внимания: отношение суммарного веса внимания на контекст к суммарному весу внимания на все предыдущие токены (Рисунок 2). Если на каком-то шаге это отношение сильно падает (модель больше смотрит на своё же продолжение, а не на данные), есть риск, что она начала галлюцинировать подробности от себя. На основе признаков, извлечённых из таких коэффициентов для разных голов, обучается простой линейный классификатор, распознающий предложения с галлюцинациями. Преимущество Lookback Lens в том, что он не требует дополнительных внешних моделей: детектор встроен в сам процесс генерации и использует информацию, которую можно получить из модели «изнутри» практически без накладных расходов. Авторы также демонстрируют, что интегрируя детектор в процесс декодирования (например, запрещая токены, приводящие к помеченной галлюцинации,

или перепроверяя сгенерированное) можно снизить частоту галлюцинаций без существенного ущерба связности ответа.

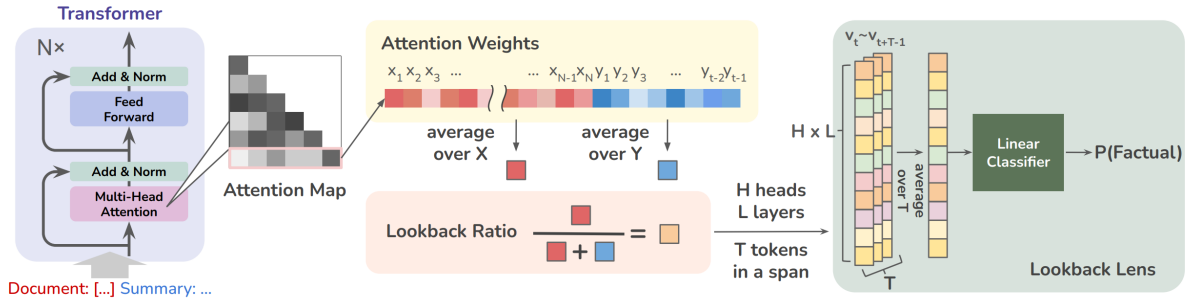

Рис. 2: Детекция галлюцинаций с помощью Lookback Lens.

LLM-as-a-Judge. Распространённый black-box подход – задействовать саму LLM (либо другую, более мощную модель) для проверки фактов ответа. Реализуется это посредством специального запроса: модели передаются исходный вопрос, контекст c и сгенерированный ответ r, после чего её просят вынести вердикт: соответствует ли ответ фактам из контекста, содержит ли ошибки, какие именно. Например, можно явно запросить: «Проанализируй ответ на соответствие приведённому тексту и укажи, есть ли в нём недостоверные утверждения». Если такая проверка проводится другой моделью (скажем, использованием GPT-4 для оценки ответа от GPT-3), это эквивалентно автоматизированной экспертной оценке. Подходы вида G-Eval [18] показывают, что современные SOTA модели способны достаточно точно выявлять грубые галлюцинации у менее эффективных моделей. Однако и они не безошибочны, к тому же этот метод увеличивает затраты (требуется запускать модель-эксперт для каждого ответа).

Метрики на основе атомарных фактов. Разработаны специальные метрики, которые автоматически измеряют долю фактически верных утверждений в тексте ответа. Метрика FActScore [17] предлагает разбить сгенерированный ответ на набор атомарных фактов и для каждого пытаться найти подтверждение в базах знаний или исходном контексте (Рисунок 3). Затем вычисляется отношение числа подтверждённых фактов к общему числу фактов – это и есть FactScore, показатель фактической точности ответа (чем ближе к единице, тем меньше галлюцинаций). Подобные метрики можно использовать и для детекции: если FactScore ниже заданного порога, то ответ помечается как недостоверный. Важным элементом здесь выступает алгоритм

извлечения и проверки фактов. Некоторые решения используют модели естественно-языкового вывода (англ. Natural Language Inference, NLI): для каждого утверждения из ответа формулируется гипотеза и проверяется, следует ли она из контекста (англ. entailment) или нет [9]. Если хотя бы одно ключевое утверждение не поддержано контекстом, ответ считается галлюцинирующим.

Модели-классификаторы. Ещё один подход – применение специализированных моделей, таких как трансформеры-энкодеры, которые обучаются детекции галлюцинаций в black-box режиме как на уровне токенов, так и на уровне всего ответа модели-генератора. Такой подход доказал свою эффективность в работах [48] и [47], где state-of-the-art энкодеры были обучены на больших корпусах синтетических данных и показали результаты, сравнимые с современными LLM-as-a-judge подходами.

Методы детекции могут комбинироваться для повышения надёжности. Например, в продукции можно сначала прогонять ответ через быстро вычислимый классификатор по скрытым состояниям (white-box детектор), а затем спорные случаи перепроверять мощной моделью-судьёй (black-box). В итоге цель всех этих техник – до выдачи ответа пользователю определить, не несёт ли он вымышленной информации.

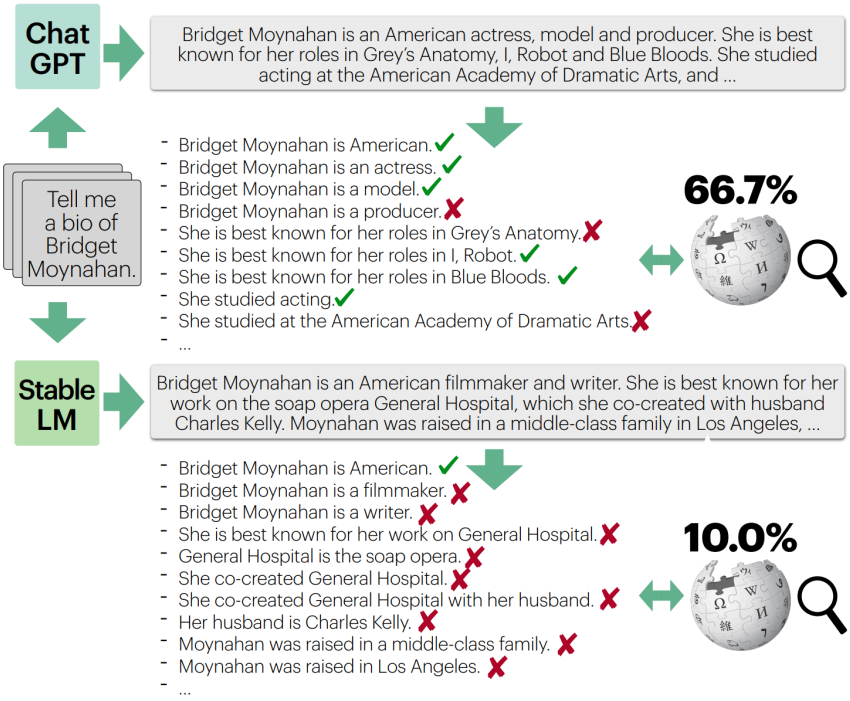

Рис. 3: Метод FactScore – извлечение и оценка атомарных фактов, полученных из ответа языковой модели.

3.2 Методы минимизации контекстуальных галлюцинаций

Если обнаружена высокая вероятность галлюцинации, следующим шагом является её минимизация – либо коррекция уже сгенерированного текста, либо предотвращение генерации ошибочного фрагмента изначально. Существует два основных класса подходов к уменьшению галлюцинаций: методы, влияющие на процесс декодирования ответа (во время генерации токенов), и методы, заставляющие модель отказаться от ответа или перепроверить себя при недостаточной уверенности. Рассмотрим ключевые представители каждого направления:

Декодирование с учетом контекста. В данной группе находятся техники, модифицирующие алгоритм выборки токенов с учётом предоставленного контекста. Идея состоит в том, чтобы поощрять модель генерировать текст, максимально опирающийся на контекст. Так, в работе [42] для каждого токена определять, насколько модель опирается на предоставленный контекст, а не на свои параметрические знания. Авторы вводят метрику Context-Awareness Score (CAS), которая измеряет, как сильно распределение вероятностей следующего токена меняется при наличии контекста по сравнению с генерацией без контекста (Рисунок 4).

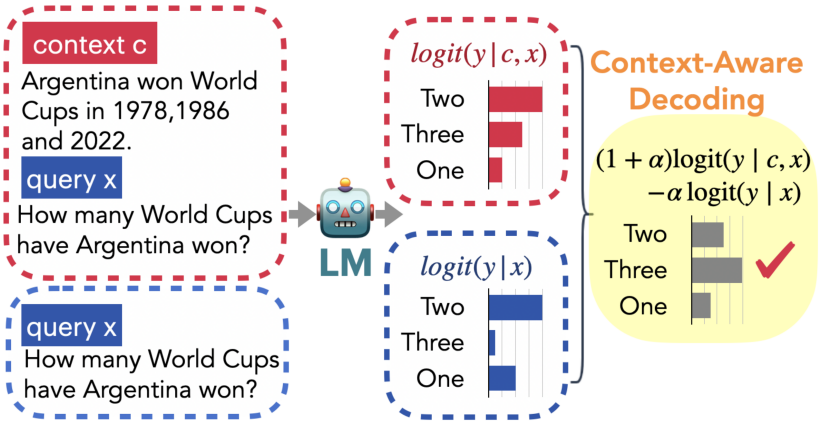

Рис. 4: Метод Context-Aware Decoding – схема декодирования.

Технически это реализуется путем выполнения двух параллельных проходов через модель: один с полным контекстом, другой – только с вопросом. Затем вычисляется разница между этими распределениями с помощью KL-дивергенции. Более сложный вариант декодирования можно осуществить на основе коэффициентов возврата ([35]), рассмотренных ранее: на каждом шаге генерации вычисляется показатель lookback ratio – отношение суммарного веса внимания модели на слова контек-

ста к весу внимания на уже сгенерированный текст. Если этот показатель падает (т.е. модель начинает уделять больше внимания априорным знаниям, а не источнику), велик риск, что она отклоняется от фактов. Авторы интегрируют детектор на основе lookback ratio прямо в ход декодирования: как только модель начинает «забывать» про контекст, алгоритм замедляет генерацию и перенаправляет внимание обратно на контекст (Рисунок 5). Экспериментально такой способ декодирования – Lookback-guided decoding – позволил значительно снизить число контекстуальных галлюцинаций без дополнительного обучения модели.

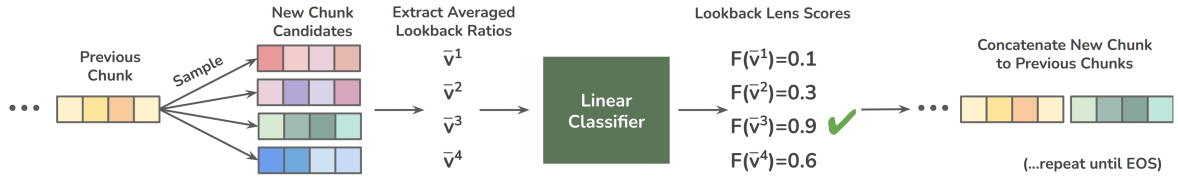

Рис. 5: Lookback Lens Guided Decoding – выбор лучшего фрагмента последовательности, который с меньшей вероятностью окажется галлюцинацией.

Энтропийно-ориентированное декодирование. Ещё один подход – задействовать показатели энтропии (неопределённости) для управления выбором токенов. Например, метод, предложенный в работе [27], выполняет параллельное декодирование с несколькими вариантами контекста и измеряет энтропию выходного распределения (Рисунок 6). Идея состоит в том, чтобы отдать предпочтение путям генерации с более низкой энтропией, полагая, что на них модель более уверена и опирается на надёжные сведения. Конкретно, в Retrieval-Augmented Generation авторы формируют ансамбль из нескольких документов контекста и на каждом шаге вычисляют объединённое распределение слов, акцентируя те, что дают наименьшую энтропию. Дополнительно вводится контрастивный механизм: сравнивается распределение токенов, полученное с использованием контекста, с «фоновым» распределением, полученным без контекста (или с более раннего слоя модели, содержащего в основном параметрические знания). Если разница велика, значит контекст внёс ясность и снизил неопределённость – в этом случае выбор делается в пользу контекстуально обусловленных токенов. Такой entropy-guided decoding позволяет вытеснить влияние внутреннего «шумного» знания модели и сконцентрироваться на внешних фактах, что уменьшает риск галлюцинаций по данным экспериментальной оценки на открытых QA-наборах [27].

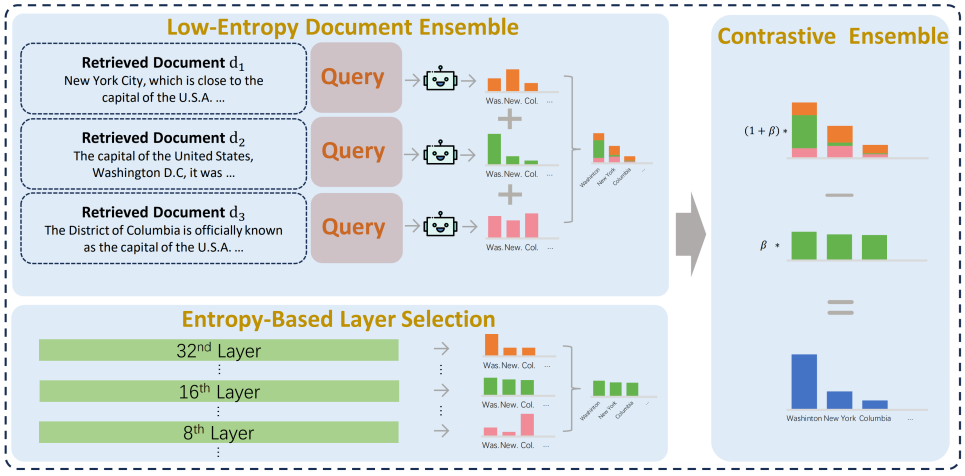

Рис. 6: Процесс энтропийно-ориентированного декодирования.

Декодирование с контролем логического следования. В этом подходе во время генерации или сразу после неё проводится проверка логической согласованности с контекстом с помощью модели естественно-языкового вывода (англ. NLI). Практически это может быть реализовано как дополнительный шаг: после генерации каждого завершенного предложения ответа система запускает NLI-модель, которая получает на вход это предложение (гипотезу) и контекст (основание), и выдает одну из меток: Entailment (следует из контекста), Contradiction (противоречит контексту) или Neutral (не доказано). Если получена метка Contradiction – декодер отменяет последнее предложение и пытается перегенерировать его иначе. Если метка Neutral – это сигнал, что модель добавила что-то сверх контекста; в зависимости от требований можно либо принять нейтральные добавления (если они безвредны), либо также пересмотреть. Таким образом, NLI-guided подход действует как «правило контроля качества» после черновой генерации. В литературе данный метод встречается как часть комплексных систем (например, для абстрактного саммари NLI-фильтрация предложений демонстрирует снижение галлюцинаций [4]). В QA с контекстом аналогичные идеи также применялись: к примеру, в [14] встроена проверка entailment для каждого факта ответа. Хотя такой подход увеличивает время ответа, он существенно повышает точность: модель просто не выпускает противоречивых с контекстом утверждений, отфильтровывая их.

DoLa-декодирование. Decoding by Contrasting Layers (DoLa) – это специальная стратегия декодирования, направленная на повышение фактотлогичности ответов без использования внешнего контекста, но её идеи применимы и в RAG-сценариях.

В работе [29] предложено вычислять распределение следующего токена не напрямую с последнего слоя трансформера, а с комбинации промежуточных слоёв: берётся разность логитов между верхними и нижними слоями модели (Рисунок 7). Интуитивно, нижние слои больше отражают «поверхностную» связанность текста, тогда как средние слои хранят фактические знания [16]. Комбинируя их, DoLa пытается усилить сигнал именно от фактических знаний и ослабить влияние «языковой плавности». Результат – модель чаще выбирает токены, соответствующие известным ей фактам, и реже – просто правдоподобные, но вымышленные продолжения. DoLa показала улучшение точности на задачах TruthfulQA [10] для семейств LLaMA на 12–17 процентных пунктов, снижая число явно неправильных утверждений [29].

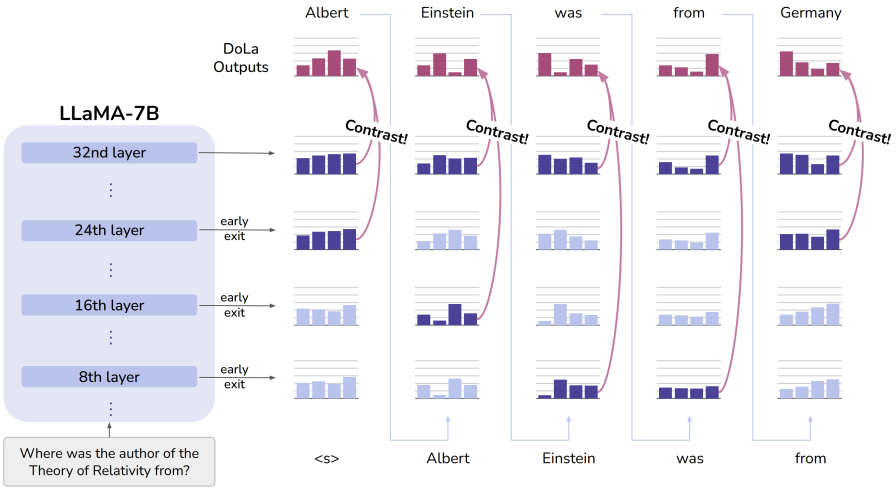

Рис. 7: Схема DoLa декодирования.

Отказ от генерации при неуверенности. Класс принципиально иных подходов состоит в том, чтобы научить модель не отвечать, если она не уверена в корректности ответа. Вместо того чтобы «галлюцинировать» произвольный факт, правильно настроенная LLM должна отклонить запрос или признать отсутствие знаний. Для достижения этого разрабатываются специальные методики обучения. Refusal-aware tuning (или R-Tuning) [38] – это подход тонкой настройки LLM, при котором в обучающую выборку добавляются примеры вопросов, на которые модель не знает ответа, и желаемая реакция – фраза вроде «Извините, у меня недостаточно данных чтобы ответить на вопрос». Таким образом, модель поощряется генерировать отказ в случаях, выходящих за её компетенцию. Экспериментально R-Tuning значительно снижает долю галлюцинаций: в работе [38] улучшенная модель почти всегда отве-

чала «I don't know» на заведомо неизвестные ей вопросы, тогда как базовая модель пыталась выдумать ответ. Другой мощный инструмент – обучение с подкреплением по обратной связи от человека (англ. Reinforcement Learning from Human Feedback, RLHF) [11]. Хотя RLHF в целом предназначен для улучшения соответствия ответа предпочтениям человека, одним из ярких эффектов этой техники стало уменьшение фактических ошибок. Аннотаторы при сравнении ответов склонны выбирать правдивые и полезные ответы, поэтому модель в процессе RLHF получает наказание за галлюцинации и поощрение за корректность. В результате, как показано в техническом отчёте OpenAI по GPT-4, RLHF-дообучение практически удвоил точность модели на ряде сложных вопросов по сравнению с предобученной версией [21].

Ещё одним направлением являются многошаговые процессы самопроверки: вышеупомянутый метод Chain-of-Verification [26] фактически внедряет этап рефлексии – модель сперва генерирует черновой ответ, затем разбивает его на факты и задаёт сама себе уточняющие вопросы по каждому факту с проверкой по контексту, и лишь после этого выдаёт окончательный ответ. Такая цепочка проверки значительно снижает число скрытых галлюцинаций, поскольку модель устраняет непроверенные детали. Похожий принцип применён и в модели FAVA [30] (Fine-grained Automated Verification and Amendment): она построена на основе LLM с дополнительным доступом к поиску, и умеет автоматически находить мелкие неточности в сгенерированном тексте и исправлять их, привлекая внешние источники. FAVA получает на вход контекст и первоначальный ответ, выделяет потенциально галлюцинирующие фрагменты и заменяет их на подтверждённые факты, если таковые находятся. В бенчмарках FAVA заметно превзошла GPT-4 по точности именно благодаря целенаправленному устранению мелких «галлюциногенных» вставок [30].

Обзор показал, что совокупными усилиями сообщества разрабатываются эффективные средства для борьбы с галлюцинациями в ответах LLM. Методы детекции позволяют автоматически отслеживать качество генерируемых ответов, а методы минимизации – улучшать фактическую состоятельность этих ответов без ручного вмешательства. Тем не менее, полностью решить проблему пока не удалось: даже новейшие модели всё ещё склонны в $\sim$10–20% случаев вводить неподтверждённую информацию, особенно на сложных или выходящих за их опыт запросах. Это мотивирует дальнейшие исследования, в частности, в направлении white-box модификаций, которые могли бы гарантировать соответствие ответа предоставленному контексту.

4 Методология минимизации контекстуальных галлюцинаций

4.1 Формальная постановка задачи

Чтобы четко определить задачи, решаемые в рамках работы, введём необходимые обозначения. Пусть:

- D – исходный набор данных, содержащий примеры запросов и ответов;
- c – контекст (промпт с извлеченными документами или история диалога);
- r – ответ, сгенерированный большой языковой моделью;
- $\widehat{r}$ – целевой (эталонный) ответ;
- S – набор внутренних состояний модели, доступных в процессе генерации;
- $y = \mathbb{I}[r \not\sim c]$ – индикаторная функция наличия галлюцинации, принимающая значение 1, если ответ r содержит информацию, не следующую из контекста c, и 0 в противном случае.

Контекстуальная галлюцинация возникает, когда сгенерированный ответ r содержит информацию, не следующую из контекста c. Формально, если существует факт f в r, такой что $c \not\models f$, то r содержит галлюцинацию и $y = 1$.

Задача детекции галлюцинаций формулируется как построение функции $f(c, r, S; \varphi)$, параметризованной φ, которая оценивает вероятность наличия галлюцинации на основе контекста c, ответа r и внутренних состояний модели S. Цель – максимизировать правдоподобие, минимизируя классическую функцию бинарной кросс-энтропии:

$$\mathcal{L}_{hallucination}(c, r, S; \varphi) = -\mathbb{E}_{(c,r)\sim D}\Big[y \log(f(c, r, S; \varphi)) + (1-y) \log(1-f(c, r, S; \varphi))\Big]. \quad (4.1)$$

Задача минимизации галлюцинаций при сохранении качества генерации формулируется как комплексная оптимизационная проблема:

$$\min_{\varphi, r, \lambda} \Big[\mathcal{L}_{hallucination}(c, r, S; \varphi) + \lambda \cdot \mathcal{L}_{generation}(r, \widehat{r})\Big], \quad (4.2)$$

где $\mathcal{L}_{generation}(r, \widehat{r})$ – исходная функция потерь для оценки качества генерации относительно эталонного ответа:

$$\mathcal{L}_{generation}(r, \widehat{r}) = -\mathbb{E}_{(r,\widehat{r})\sim D}\Big[\sum_{i=1}^{n} \widehat{r}_i \log p(r_i|r_{<i})\Big]. \tag{4.3}$$

Гиперпараметр λ определяет баланс между достоверностью (отсутствием галлюцинаций) и качеством генерации. При $\lambda \to 0$ основной приоритет отдаётся минимизации галлюцинаций, в то время как при увеличении λ возрастает важность соответствия ответа целевому.

В рамках данной работы основной упор делается на использование внутренних состояний модели S для улучшения детекции и минимизации галлюцинаций. Таким образом, формально решаемая задача заключается в разработке алгоритма, который оптимизирует представленную выше целевую функцию путём: (i) создания эффективной функции детекции галлюцинаций $f(c, r, S; \varphi)$, использующей информацию о внутреннем состоянии модели; (ii) разработки метода модификации процесса генерации для минимизации вероятности возникновения галлюцинаций при сохранении информативности ответов.

4.2 Архитектура решения

В данном разделе описывается архитектура разработанного подхода, направленного на обнаружение и минимизацию галлюцинаций в ответах больших языковых моделей. Подход состоит из двух основных модулей: (i) методов детекции галлюцинаций – white-box анализ внутренних признаков модели и (ii) методов их минимизации (генерация альтернативных ответов и дообучение модели). Ниже приводится подробное описание компонентов каждого модуля, включая используемые метрики, алгоритмы ансамблирования, а также схемы генерации и дообучения модели. Краткая схема методологии представлена на Рисунке 8.

Для решения задачи детекции галлюцинаций, рассмотренной в разделе 4.1, предлагается два подхода: подход, предполагающий обучение без учителя, неявно решающий задачу оптимизации $\mathcal{L}_{hallucination}(c, r, S; \varphi)$, а также обучаемый с учителем, напрямую оптимизирующий функцию бинарной кросс-энтропии.

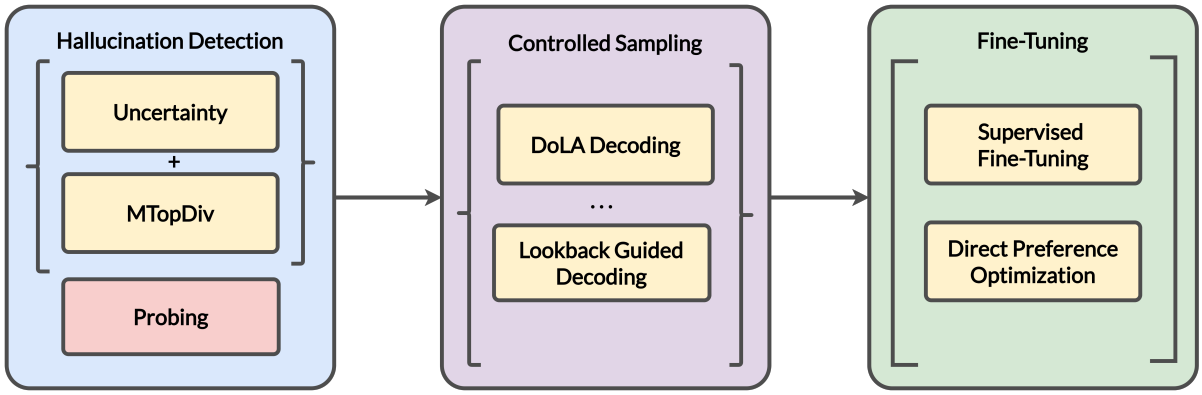

Рис. 8: Схема предлагаемой методологии детекции и минимизации галлюцинаций.

4.2.1 Необучаемая детекция контекстуальных галлюцинаций на основе топологического анализа и оценки неопределенности

Топологические признаки. Основным компонентом необучаемой детекции является метод Manifold Topology Divergence (MTopDiv), адаптированный для анализа графов внимания [46]. Данный подход основан на гипотезе о том, что ответы с контекстуальными галлюцинациями создают топологически отличные структуры в картах внимания по сравнению с контекстно-обоснованными ответами. Для чего рассматривается граф внимания, который строится по соответствующим матрицам внимания: узлы графа соответствуют токенам, а веса рёбер определяются коэффициентами внимания w_{ij} между токенами i и j. Для вычисления *дивергенции* между подграфом ответа (R) и подграфом контекста (P) веса внимания интерпретируются как длины (псевдо-расстояния $1 - w_{ij}$) и вычисляем длину минимального остовного леса, связывающего вершины ответа R с вершинами контекста P. Эта величина и определяется как значение метрики $\mathrm{MTopDiv}(R, P)$ – чем больше длина такого остовного леса, тем слабее *связан* ответ с контекстом, что указывает на возможную галлюцинацию в ответе.

Оценка неопределенности. Для повышения качества детекции предлагается комбинирование MTopDiv с классическими метриками потокенной неопределенности, опробированными в работе [25]. Рассматриваются три метрики, вычисляемые для каждого токена ответа y_t с апостериорным распределением $P_t = P_\Theta(y_t|x, y_{<t})$ на словаре:

– *Энтропия (англ. Entropy)* $H_t = -\sum_i P_t(i), \ln P_t(i)$. Энтропия оценивает степень неопределённости распределения вероятностей следующего слова. Высокая эн-

тропия указывает на расплывчатость знаний модели на данном шаге генерации.

– *Максимальная вероятность (англ. MaxProb)* $Q_t = \max_i P_t(i)$. Максимальная вероятность выбранного токена (или уверенность в выборе). Низкое значение Q_t (близкое к 0.5 и ниже) указывает, что модель не имеет выраженного предпочтения для следующего слова.

– *Разница вероятностей (англ. Margin)* $M_t = P_t(i_{(1)}) - P_t(i_{(2)})$, разница между наибольшей и второй по величине вероятностями. Небольшой *margin* (близкий к 0) означает, что несколько вариантов конкурируют, и модель неуверена в выборе токена.

Каждая из этих метрик вычисляется для всех токенов $t = 1 \dots T$ в сгенерированном ответе длины T. Затем необходимо агрегировать их значения в один показатель на уровне всего ответа. Для агрегации последовательности значений $u_1, \dots, u_T$ предлагается несколько функций:

– Среднее значение: $u_{\text{mean}} = \dfrac{1}{T}\sum_{t=1}^{T} u_t$. Усреднение энтропии (или других метрик) даёт общее ощущение «растянутости» неопределённости по всему ответу.

– Максимум: $u_{\max} = \max_{1 \le t \le T} u_t$. Берётся наибольшее значение неопределённости среди всех токенов. Например, $\max_t H_t$ отражает самый неуверенный момент модели при генерации ответа.

– Квантиль: $u_{\alpha\text{-quantile}} = \alpha$-квантиль множества u_t. Мы экспериментально рассматриваем квантиль с параметром $\alpha \in (0, 0.4, 0.8)$. При $\alpha = 0$ получается минимальное значение (например, минимальная энтропия токена в ответе), при $\alpha = 0.8$ – значение, превышающее 80% наблюдений. Квантиль позволяет учесть распределение неопределённости: например, высокий 0.8-квантиль энтропии означает, что даже в относительно уверенных шагах генерации наблюдается заметная неопределённость.

Каждый из вариантов агрегации может быть применён к каждой из трёх метрик (H_t, Q_t, M_t), порождая различные признаки неопределённости на уровне ответа. В частности, для энтропии полезным оказался средний показатель, тогда как для MaxProb и Margin логично рассматривать минимальные значения по токенам (что эквивалентно максимуму неопределённости). Полученные агрегированные признаки мы обозначим как $U_1, U_2, \dots$ (в общем случае их несколько).

Комбинирование метрик и итоговый ансамблевый коэффициен. После вычисления MTopDiv и агрегированных метрик неопределённости необходимо объединить их в единый итоговый коэффициен галлюцинации. В данном случае рассматривается простое линейное ансамблирование с возможностью последующей калибровки на целевом валидационном множестве или множестве отличного домена. Пусть $d = \mathrm{MTopDiv}(R, P)$ – топологическое отклонение ответа, а u_j – одна или несколько агрегированных метрик неуверенности. Тогда итоговый коэффициен можно определить как линейную комбинацию:

$$S_{\text{ensemble}} = w_d \cdot f_d(d) \; + \; \sum_j w_{u_j} \cdot f_u(u_j), \tag{4.4}$$

где w_d и w_{u_j} – весовые коэффициенты (гиперпараметры, которые можно подобрать по валидационным данным), а функции f_d, f_u представляют собой возможную калибровку или нормализацию метрик. В простейшем случае в качестве f можно взять тождественную функцию (или мин-макс нормализацию на диапазон $[0, 1]$).

Интерпретация: высокое значение S_{ensemble} будет у ответов, для которых одновременно наблюдается структурная оторванность от контекста (большой d) и внутренняя неуверенность модели (большой усреднённый H_t, малые Q_t и M_t). Такие ответы с высокой вероятностью являются галлюцинациями. Наоборот, если ответ хорошо привязан к контексту (малый MTopDiv) и модель уверенно его сформулировала (низкая энтропия, высокая уверенность по токенам), то S_{ensemble} будет низким, указывая на правдоподобный, основанный на контексте ответ.

4.2.2 Обучаемая детекция контекстуальных галлюцинаций методами пробинга

Для supervised детекции предлагаются модификации существующих методов пробинга, адаптированные для работы с ограниченными объемами обучающих данных [44].

Извлечение признаков. Система извлечения признаков включает два основных компонента:

1. *Агрегация скрытых состояний*: для последовательности скрытых состояний

$H = (h_1, \ldots, h_n)$ применяются операции:

$$f_{\text{mean}} = \frac{1}{n} \sum_{i=1}^{n} h_i, \tag{4.5}$$

$$f_{\text{max}} = \text{MaxPool}(H), \tag{4.6}$$

$$f_{\text{last}} = h_n. \tag{4.7}$$

2. *Lookback признаки*: адаптирован метод вычисления отношения внимания между контекстом и сгенерированными токенами, предложенный в работе [36] для спановой детекции галлюцинаций:

$$\text{LookbackRatio}_{ij} = \frac{\sum_{c \in \text{Context}} \sum_{r \in \text{Response}} A_{ij}[r, c]}{\sum_{r \in \text{Response}} \sum_{t=1}^{|S|} A_{ij}[r, t]}, \tag{4.8}$$

где A_{ij} – матрица внимания головы j слоя i.

Снижение размерности и классификация. Для работы с ограниченными данными применяется снижение размерности с помощью Метода главных компонент (англ. Principal Components Analysis, PCA) и Uniform Manifold Approximation (UMAP) and Projection) [7]. В качестве мета-классификаторов используются: логистическая регрессия, градиентный бустинг над решающими деревьями (CatBoost [2]), а также трансформер для in-context классификации (табличная фундаментальная модель для задач с малым количеством данных – TabPFN [23]).

4.2.3 Улучшение генерации с помощью детекции контекстуальных галлюцинаций

Если модуль детекции определяет вероятность присутствия галлюцинации, то модуль минимизации галлюцинаций направлен на снижение фактического их появления в ответах модели. Предлагаемый подход включает: (i) улучшение процесса генерации с помощью внешнего классификатора галлюцинаций и (ii) дообучение исходной языковой модели так, чтобы она начала генерировать более близкий контексту ответ.

Генерация кандидатов с пониженным содержанием галлюцинаций. Для снижения галлюцинаций в процессе генерации предлагается использовать контекстно-обоснованные методы семплирования, включающие DoLA Decoding, Context-Aware

Decoding и Lookback Guided Decoding описанные в разделе 3. Эти методы позволяют генерировать множество альтернативных кандидатов $Y = \{y_1, y_2, \ldots, y_k\}$, учитывающих взаимосвязь с контекстом и внутренние признаки модели, что способствует повышению качества и снижению числа галлюцинаций.

Выбор наилучшего кандидата по метрике детекции. Выбор финального ответа из набора Y осуществляется на основе значения MTopDiv или предложенного ансамбля MTopDiv+Uncertainty: каждому кандидату $y^{(i)} \in Y$ необходимо получить оценки $S^{(i)} = S_{\text{MTopDiv}}(y^{(i)})$ (или расширенные S_{Ensemble} с учётом классификатора). Далее выбирается наиболее оптимальная генерация:

$$y^* = \arg\min_{y^{(i)} \in Y} S^{(i)}. \tag{4.9}$$

Минимизация обоснована тем, что низкий коэффициен соответствует ответу, прочно опирающемуся на контекст и уверенность модели. Таким образом, y^* – наиболее правдоподобный и фактически корректный ответ по версии нашего детектора.

Дообучение языковой модели. Финальный этап – дообучение исходной LLM, чтобы она впредь самостоятельно избегала галлюцинаций. Для дообучения можно использовать различные методы, от исходного SFT (supervised fine-tuning) до RLHF [12]. Однако каждый из этих подходов имеет существенные ограничения в контексте детекции галлюцинаций.

SFT требует высококачественных примеров «правильных» ответов, но в задаче минимизации галлюцинаций сложно определить единственный корректный ответ, особенно для открытых вопросов. Более того, SFT может привести к переобучению на конкретных формулировках, не обучая модель принципам избегания галлюцинаций.

Классический RLHF требует обучения отдельной так называемой *reward-модели* на вручную размеченных парах предпочтений, что является дорогостоящим и времязатратным процессом. Кроме того, RLHF страдает от нестабильности обучения из-за необходимости поддержания баланса между *reward-моделью* и *policy-моделью*.

Поэтому предлагается использовать оптимизацию методом Direct Preference Optimization (DPO) [15], обучая языковую модель на псевдо-парах предпочтений, полученных с помощью описанных ранее автоматических методов детекции. Функ-

ция потерь DPO определяется как:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y^+|x)}{\pi_{\text{ref}}(y^+|x)} - \beta\log\frac{\pi_\theta(y^-|x)}{\pi_{\text{ref}}(y^-|x)}\right)\right], \tag{4.10}$$

где y^+ и y^- – предпочитаемый и непредпочитаемый ответы соответственно, π_θ – обучаемая модель, π_{ref} – референсная модель, β – параметр температуры.

Формирование пар предпочтений осуществляется следующим образом:

$$y^+ = \arg\min_{y\in\mathcal{Y}(x)} \text{HallucinationScore}(y), \tag{4.11}$$

$$y^- = \arg\max_{y\in\mathcal{Y}(x)} \text{HallucinationScore}(y), \tag{4.12}$$

где $Y(x)$ – множество сгенерированных кандидатов для входа x.

После нескольких эпох обучения по функции $\mathcal{L}_{\text{DPO}}$ на сгенерированных парах, получается обновлённую модель. Она уже скорректирована: при получении нового запроса модель с большей вероятностью сгенерирует ответ, похожий на прежние y^+ (правдивые, привязанные к контексту), и с меньшей – на y^- (неконсистентные с контекстом). Это подтверждается снижением среднего коэффициена галлюцинации для ответов новой модели по сравнению с исходной, что описано в разделе 5. Все этапы – от генерации кандидатов до финального дообучения – составляют единый масштабируемый подход к минимизации галлюцинаций, основанный на совмещении white-box анализа и эффективного обучения предпочтений модели без привлечения ручной разметки или разметки от проприетарных моделей.

5 Вычислительные эксперименты

5.1 Детекция контекстуальных галлюцинаций

В данном разделе представлены экспериментальные исследования эффективности предложенных методов детекции галлюцинаций, включающие как обучаемые, так и необучаемые подходы. Эксперименты проводились с целью оценки практической применимости разработанных методов в различных доменах и условиях ограниченного количества данных.

5.1.1 Постановка экспериментов

Для оценки эффективности предложенных методов детекции галлюцинаций использовались три основных набора данных, представляющих различные домены

вопросно-ответных задач с контекстом (RAGTruth [40], CoQA [5], SQuAD [3]), и два дополнительных индустриальных датасета, для исследования методов пробинга (ExpertQA, Emanual [31]):

- **RAGTruth** – набор данных с экспертной разметкой галлюцинаций на уровне span'ов в ответах моделей в RAG постановке. Для настоящего исследования разметка была агрегирована до уровня всего ответа: ответ считался содержащим галлюцинацию, если в нем присутствовал хотя бы один галлюцинированный фрагмент.

- **SQuAD** – классический набор данных для вопросно-ответных задач, основанный на статьях Википедии. Разметка галлюцинаций производилась автоматически с использованием GPT-4o с последующей валидацией экспертами [46].

- **CoQA** – набор данных для разговорного вопросно-ответного взаимодействия, требующий понимания контекста. Аналогично SQuAD, разметка осуществлялась с помощью GPT-4o.

- **ExpertQA** – набор данных с вопросами от экспертов в различных областях знаний.

- **EManual** – специализированный датасет для вопросно-ответной системы по электронным устройствам.

Для последних двух датасетов ответы были сгенерированы с помощью исследуемых моделей и аннотированы на наличие галлюцинаций с использованием GPT-4o, что соответствует методологии, описанной в работах [20].

В качестве основных метрик оценки использовались:

$$\text{ROC-AUC} = \int_0^1 \text{TPR}(\text{FPR}^{-1}(x))\,dx, \tag{5.1}$$

где $\text{TPR} = \frac{TP}{TP+FN}$ – истинно положительный показатель, $\text{FPR} = \frac{FP}{FP+TN}$ – ложно положительный показатель.

$$\text{PR-AUC} = \int_0^1 \text{Precision}(\text{Recall}^{-1}(x))\,dx, \tag{5.2}$$

где $\text{Precision} = \frac{TP}{TP+FP}$, $\text{Recall} = \frac{TP}{TP+FN}$.

Данные метрики позволяют комплексно оценить качество бинарной классификации независимо от выбранного порога принятия решения.

Для генерации ответов использовались две модели: Llama-2-7b-chat-hf и Mistral-7B-Instruct-v0.1, что позволило оценить обобщаемость методов детекции галлюцинаций на разных архитектурах.

5.1.2 Описание результатов

Исследование обучаемых методов на индустриальных данных. Первым этапом исследования стала оценка эффективности предложенных модификаций методов пробинга в индустриальной RAG постановке. На рисунке 9 представлены результаты описанных в разделе 4 комбинаций методов извлечения признаков, алгоритмов снижения размерности и классификаторов на трех датасетах: EManual, ExpertQA и RAGTruth; в сравнении также участвуют RAGAS и LLM-judge, которые используют проприетарные state-of-the-art LLM. Важно отметить, что в данном случае методы пробинга применяются ко вспомогательным моделям – экстракторам скрытых состояний (Gemma-2-9B-It [32], Llama-3.1-8B [41], Qwen2.5-7B-Instruct [37]), так как исходные ответы на вопросы были сгенерированы проприетарными LLM.

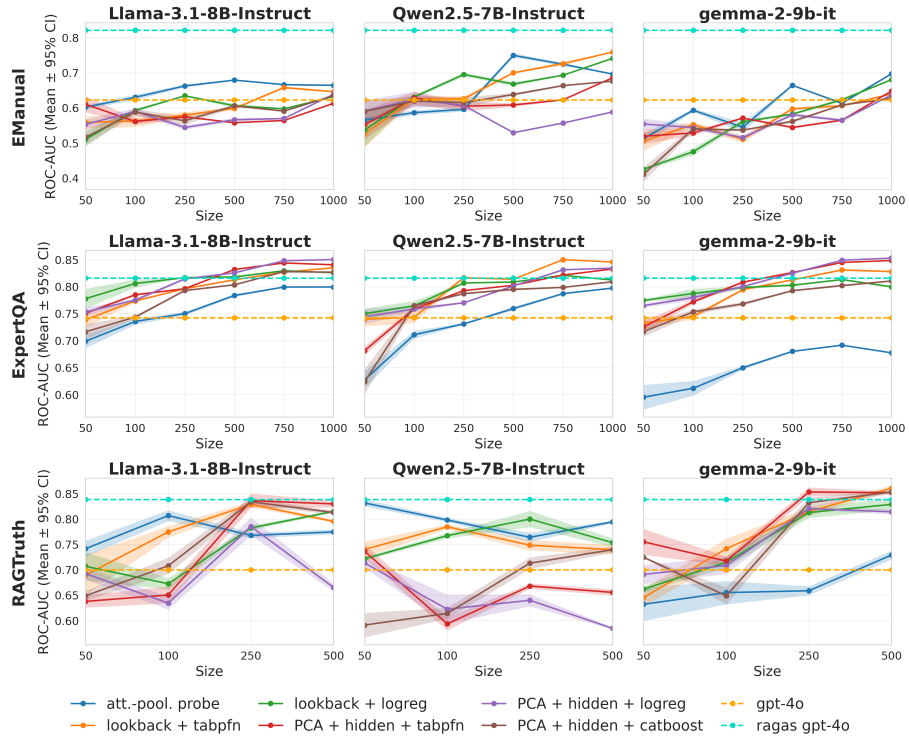

Рис. 9: Зависимость качества на тестовой выборке от размера обучающего набора для предложенных методов пробинга и бейзлайнов.

Анализ результатов показывает, что наиболее эффективным подходом является извлечение признаков типа lookback с опциональным применением PCA для снижения размерности и последующая классификация методом TabPFN. Все предложенные методы демонстрируют высокие результаты даже на ограниченном количестве обучающих примеров (около 250 образцов), что особенно важно для индустриальных приложений, где сбор и аннотация данных могут быть дорогостоящими.

Интересно отметить, что классификатор TabPFN, основанный на in-context learning подходе для табличных данных, показывает наилучшие результаты среди всех рассмотренных классификаторов, что подтверждает эффективность подобных подходов даже для задач пробинга галлюцинаций.

Сравнительный анализ обучаемых и необучаемых методов. Следующим шагом стало сравнение эффективности различных методов детекции галлюцинаций на разных по доменам вопросно-ответных датасетах с контекстом. В таблицах 1 и 2 представлены результаты как обучаемых методов (Attention-pooling probe, Linear probe, Lookback+TabPFN), так и необучаемых методов (SelfCheckGPT, Semantic entropy, Tokenwise entropy, INSIDE, MTopDiv и MTopDiv+Uncertainty) рассмотренных ранее. Из-за ограничения вычислительных ресурсов для топологического анализа были использованы только выделенные в работе [46] головы внимания.

Результаты показывают, что обучаемые методы в целом превосходят необучаемые методы по качеству детекции галлюцинаций. Однако среди необучаемых подходов наиболее эффективным оказался предложенный подход MTopDiv+Uncertainty, который достигает результатов, близких к supervised методам, в особенности на датасете CoQA.

Ансамблирование MTopDiv с методами оценки неопределенности по токенам позволяет значительно улучшить качество детекции галлюцинаций. При этом важно отметить, что для разных моделей и датасетов оптимальные метрики неуверенности и способы их агрегации могут различаться.

Анализ методов потокенной оценки неопределенности. Для более детального анализа эффективности различных методов оценки неопределенности по токенам и способов их агрегации был проведен перебор всех возможных комбинаций для ансамблирования с MTopDiv. В таблицах 3, 4 и 5 представлены результаты этого анализа для датасетов CoQA, SQuAD и RAGTruth соответственно.

Таблица 1: Результаты методов детекции галлюцинаций для модели Llama-2-7b-chat-hf.

Метод	RAGTruth QA		SQuAD		CoQA	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
Supervised						
Linear probe	0.731 ± 0.028	0.638 ± 0.038	0.967 ± 0.003	0.975 ± 0.004	0.800 ± 0.006	0.754 ± 0.011
Attention-pooling probe	**0.787 ± 0.018**	0.674 ± 0.017	0.995 ± 0.004	0.996 ± 0.003	0.955 ± 0.005	0.963 ± 0.004
Lookback+TabPFN probe	0.751 ± 0.008	**0.698 ± 0.007**	**0.998 ± 0.001**	**0.998 ± 0.001**	**0.958 ± 0.008**	**0.960 ± 0.007**
Unsupervised						
SelfCheckGPT	0.642	**0.482**	0.659	0.732	0.777	0.812
Semantic entropy	0.555	0.444	0.673	0.780	0.781	0.809
Max entropy	0.596	0.454	0.615	0.703	0.721	0.725
INSIDE	0.452	0.298	0.470	0.557	0.719	0.721
MTopDiv	0.625	0.424	0.704	0.727	0.811	0.835
MTopDiv+Uncertainty	**0.650**	0.469	**0.764**	**0.819**	**0.838**	**0.853**

Таблица 2: Результаты методов детекции галлюцинаций для модели Mistral-7B-Instruct-v0.1.

Метод	RAGTruth QA		SQuAD		CoQA	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	PR-AUC	PR-AUC
Supervised						
Linear probe	0.841 ± 0.019	0.638 ± 0.049	0.996 ± 0.002	0.993 ± 0.003	0.922 ± 0.006	0.936 ± 0.004
Attention-pooling probe	**0.915 ± 0.004**	**0.737 ± 0.021**	0.989 ± 0.002	0.983 ± 0.003	0.981 ± 0.002	0.982 ± 0.002
Lookback+TabPFN probe	0.821 ± 0.013	0.673 ± 0.016	**0.998 ± 0.002**	**0.998 ± 0.004**	**0.983 ± 0.002**	**0.985 ± 0.002**
Unsupervised						
SelfCheckGPT	0.684	0.348	0.796	0.668	**0.937**	**0.950**
Semantic entropy	0.495	0.222	0.728	0.583	0.861	0.859
Max entropy	0.727	0.405	**0.941**	**0.917**	0.837	0.881
INSIDE	0.621	0.278	0.576	0.408	0.867	0.874
MTopDiv	0.716	0.441	0.749	0.526	0.856	0.872
MTopDiv+Uncertainty	**0.778**	**0.510**	0.928	0.906	0.901	0.929

Анализ результатов показывает, что наиболее эффективными методами оценки неопределенности являются Entropy и MaxProb, что в целом объясняется тем, что Margin является лишь приближением энтропии. А оптимальные способы агрегации варьируются в зависимости от набора данных. Так, для датасета CoQA наиболее эффективной агрегацией является среднее значение (mean), для SQuAD – максимальное значение (max), а для RAGTruth также наиболее эффективным оказалось максимальное значение. Важно отметить, что эффективность методов оценки неопределенности зависит и от модели: для Mistral-7B-Instruct-v0.1 методы оценки неопределенности работают в целом лучше, чем для Llama-2-7b-chat-hf.

Для итогового ансамблирования с MTopDiv была выбрана комбинация усредненных по токенам метрик MaxProb и Margin, а также максимальное значение Entropy (как наиболее устойчивые варианты агрегаций) – при этом эксперементы показали, что каждая метрика важна для финального качества. Важно отметить, что для при-

Таблица 3: Топ-5 комбинаций методов оценки неопределенности и способов агрегации для датасета CoQA.

Метод	Агрегация	ROC-AUC	PR-AUC
Llama-2-7b-chat-hf			
Entropy	mean	**0.851**	**0.864**
MaxProb	mean	0.829	0.839
Entropy	0.8-quantile	0.826	0.827
Margin	mean	0.815	0.814
MaxProb	0.8-quantile	0.815	0.811
Mistral-7B-Instruct-v0.1			
Entropy	mean	**0.913**	**0.933**
Entropy	0.4-quantile	0.900	0.925
MaxProb	0.4-quantile	0.891	0.917
Margin	0.4-quantile	0.890	0.918
MaxProb	mean	0.885	0.917

Таблица 4: Топ-5 комбинаций методов оценки неопределенности и способов агрегации для датасета SQuAD.

Метод	Агрегация	ROC-AUC	PR-AUC
Llama-2-7b-chat-hf			
Entropy	max	**0.615**	**0.703**
MaxProb	max	0.550	0.650
Margin	max	0.524	0.610
Margin	0-quantile	0.479	0.423
MaxProb	0-quantile	0.479	0.423
Mistral-7B-Instruct-v0.1			
Entropy	max	**0.941**	**0.917**
MaxProb	max	0.858	0.832
Margin	0.8-quantile	0.830	0.803
Margin	max	0.827	0.746
MaxProb	0.8-quantile	0.813	0.790

менения ансамбля требуется калибровка коэффициентов линейной комбинации для каждой модели-генератора, при этом калибровка может проводиться на не целевой выборке – в данном случае калибровка проводилась на 40 валидационных примерах целевых наборов данных, однако схожие результаты были получены при валидации на наборе ExpertQA.

5.1.3 Выводы

Проведенные эксперименты позволяют сделать следующие выводы:

1. Обучаемые методы детекции галлюцинаций в целом превосходят необучаемые методы по качеству, особенно при наличии достаточного количества обучающих

данных.

2. Предложенный подход MTopDiv+Uncertainty является наиболее эффективным среди необучаемых методов и достигает результатов, близких к supervised методам.

3. Эффективность методов детекции галлюцинаций зависит от модели и датасета, что указывает на необходимость адаптации методов к конкретной задаче.

4. Для практического применения в индустриальных задачах можно рекомендовать использование подхода Lookback+TabPFN при наличии небольшого количества размеченных данных (около 250 образцов) или MTopDiv+Uncertainty при отсутствии размеченных данных.

5. Для ансамблирования MTopDiv с методами оценки неопределенности необходима калибровка для каждой модели, при этом домен данных для калибровки может отличаться от целевого домена.

Таблица 5: Топ-5 комбинаций методов оценки неопределенности и способов агрегации для датасета RAGTruth.

Метод	**Агрегация**	**ROC-AUC**	**PR-AUC**
Llama-2-7b-chat-hf			
Margin	max	**0.615**	**0.540**
Entropy	max	0.596	0.454
Entropy	mean	0.561	0.373
MaxProb	max	0.548	0.403
MaxProb	mean	0.544	0.359
Mistral-7B-Instruct-v0.1			
Entropy	max	**0.727**	0.405
MaxProb	max	0.723	**0.426**
Margin	max	0.642	0.383
Entropy	mean	0.637	0.292
MaxProb	mean	0.616	0.277

5.2 Минимизация контекстуальных галлюцинаций

5.2.1 Постановка экспериментов

После разработки эффективных методов детекции галлюцинаций естественным шагом является исследование подходов к их уменьшению. В рамках данной работы были рассмотрены два основных подхода:

1. Семплирование с выбором оптимальной генерации – метод, основанный на генерации нескольких ответов с последующим выбором наиболее достоверного ответа на основе классификатора галлюцинаций.

2. Дообучение модели с использованием DPO – подход, при котором модель дообучается на парах ответов (предпочтительный и непредпочтительный), где предпочтение определяется на основе классификатора галлюцинаций.

Для оценки качества сгенерированных ответов использовались две метрики, которые оценивают два критических аспекта сгенерированных ответов – близость к эталону и фактологическую точность, что напрямую соотносится с задачей сформулированной в подразделе 4.1:

– *BERTScore* [6] – метрика, основанная на контекстуальных эмбеддингах, которая оценивает семантическую близость сгенерированного ответа к референсному. В данном случае использовался state-of-the-art энкодер ModernBERT-base. Формально BERTScore вычисляется как:

$$\text{BERTScore} = \frac{1}{|r|} \sum_{t_r \in r} \max_{t_{\hat{r}} \in \hat{r}} \cos(\text{BERT}(t_r), \text{BERT}(t_{\hat{r}})), \tag{5.3}$$

где r – референсный ответ, $\hat{r}$ – сгенерированный ответ, t_r и $t_{\hat{r}}$ – токены этих ответов, а $\text{BERT}(t)$ – эмбеддинг токена t.

– *Faithfulness* от RAGAS [39] – метрика, оценивающая фактологическую точность сгенерированного ответа относительно предоставленного контекста. Вычисляется с использованием большой проприетарной модели GPT-4o как:

$$\text{Faithfulness}_{\text{RAGAS}} = \frac{|S|}{|V|}, \tag{5.4}$$

где S – множество утверждений в ответе LLM, подтвержденных контекстом, а V – множество всех утверждений в ответе.

5.2.2 Описание результатов

Эксперименты с семплированием. Для экспериментов с семплированием были выбраны следующие методы декодирования:

– *Lookback guided decoding* [35] – метод, который направляет процесс генерации, увеличивая вероятность токенов, имеющих высокие значения lookback

ratio. В данной реализации использовался усредненный для всех голов среднего слоя коэффициент lookback, размер окна 8 токенов, семплирование методом beam search с параметрами: num_beams=15, num_beam_groups=3, num_candidates=5, diversity_penalty=1.0.

– *Context-aware decoding (CAD)* [42] – метод, который модифицирует распределение вероятностей токенов на основе их контекстуальной релевантности. Использовался параметр контрастности $\alpha = 0.5$.

– *DoLa decoding* [29] – метод, использующий динамическое регулирование влияния различных слоев модели на процесс генерации.

– *Greedy decoding* – стандартный метод генерации, основанный на жадном алгоритме декодирования.

– *Ensemble guided decoding* (предложенный подход) – метод, использующий все предыдущие подходы к декодированию для генерации нескольких ответов с последующим выбором ответа с наименьшим (*Ensemble*$^{+}$) и наибольшим (*Ensemble*$^{-}$) значением предложенного ранее ансамбля MTopDiv+Uncertainty. Для каждого входного примера генерировались кандидаты всеми четырьмя методами, после чего вычислялись значения метрики галлюцинации и выбирались экстремальные варианты.

Все методы декодирования реализованы с использованием библиотеки transformers [1]. Остальные параметры генерации использовались по умолчанию согласно спецификации библиотеки. Все эксперименты проводились на A100 Nvidia GPU без использования квантизации в режиме half precision. Для оптимизации памяти использовалось KV-кеширование. Максимальная длина генерации составляла 1024 новых токенов. Результаты экспериментов для метрик BERTScore и RAGAS Faithfulness представлены в таблицах 6 и 7 соответственно.

Предложенный подход демонстрирует высокие результаты по метрике RAGAS Faithfulness на датасетах CoQA и RAGTruth для обеих моделей, что свидетельствует о его эффективности в уменьшении галлюцинаций. По метрике BERTScore предложенный подход также показывает конкурентные результаты, достигая наилучших значений на датасете CoQA. Разрыв между Ensemble^{+} и Ensemble^{-} прослеживается

[1] https://huggingface.co/docs/transformers

Таблица 6: Результаты оценки методов декодирования по метрике BERTScore (ModernBERT-base).

Набор данных	Модель	Lookback	CAD	DoLa	Greedy	Ensemble$^+$	Ensemble$^-$
CoQA	Llama-2-7b-chat-hf	0.820	0.813	0.820	0.820	**0.837**	0.796
CoQA	Mistral-7B-Instruct-v0.1	**0.822**	**0.822**	0.816	0.805	**0.822**	0.803
RAGTruth	Llama-2-7b-chat-hf	0.884	0.868	**0.890**	**0.890**	0.872	0.886
RAGTruth	Mistral-7B-Instruct-v0.1	**0.887**	0.835	0.853	0.872	0.831	0.853
SQuAD	Llama-2-7b-chat-hf	**0.781**	0.774	0.775	0.778	0.780	0.774
SQuAD	Mistral-7B-Instruct-v0.1	**0.822**	0.814	0.813	0.810	0.821	0.808

Таблица 7: Результаты оценки методов декодирования по метрике RAGAS Faithfulness (GPT-4o).

Набор данных	Модель	Lookback	CAD	DoLa	Greedy	Ensemble$^+$	Ensemble$^-$
CoQA	Llama-2-7b-chat-hf	0.789	0.574	0.774	0.707	**0.819**	0.570
CoQA	Mistral-7B-Instruct-v0.1	**0.755**	0.421	0.572	0.532	0.713	0.395
RAGTruth	Llama-2-7b-chat-hf	0.675	0.693	0.745	0.742	**0.749**	0.691
RAGTruth	Mistral-7B-Instruct-v0.1	**0.700**	0.467	0.532	0.642	0.548	0.507
SQuAD	Llama-2-7b-chat-hf	0.498	0.453	0.469	0.493	**0.516**	0.433
SQuAD	Mistral-7B-Instruct-v0.1	0.644	0.542	0.599	0.588	**0.647**	0.547

явным образом, что указывает также на эффективность предложенного ансамбля в разделении качественных и некачественных ответов с точки зрения контекстуальных галлюцинаций.

Эксперименты с дообучением. Для дальнейшего улучшения качества ответов были проведены эксперименты с дообучением моделей с использованием алгоритма DPO. В качестве пар предпочтений использовались, определенные ранее:

– *Положительные примеры (y^+)*: ответы, выбранные методом Ensemble$^+$ (с минимальным значением метрики галлюцинации).

– *Отрицательные примеры (y^-)*: ответы, выбранные методом Ensemble$^-$ (с максимальным значением метрики галлюцинации).

Для сравнения также было проведено дообучение с использованием стандартного подхода SFT, где в качестве целевых ответов использовались только положительные примеры.

Из-за ограничения вычислительных ресурсов референсная модель и обучаемая модель разделяли между собой 85% нижних слоев. Обучение производилось в режиме bf16 с оптимизатором AdamW с помощью библиотеки transformers на 4xA100

Nvidia GPU со следующими гиперпараметрами: learning_rate = 2e-6, warmup_ratio = 0.03, num_train_epochs = 1.

Таблица 8: Результаты оценки методов декодирования по метрике BERTScore (ModernBERT-base).

Набор данных	**Модель**	**Lookback**	**Greedy**	**Ensemble$^+$**	**SFT**	**DPO**
CoQA	Llama-2-7b-chat-hf	0.820	0.820	<u>0.837</u>	0.800	**0.839**
CoQA	Mistral-7B-Instruct-v0.1	**0.822**	<u>0.805</u>	**0.822**	0.783	0.792
RAGTruth	Llama-2-7b-chat-hf	<u>0.884</u>	**0.890**	0.872	0.877	0.879
RAGTruth	Mistral-7B-Instruct-v0.1	<u>0.887</u>	0.872	0.831	**0.890**	0.814
SQuAD	Llama-2-7b-chat-hf	<u>0.781</u>	0.778	0.780	**0.796**	**0.796**
SQuAD	Mistral-7B-Instruct-v0.1	**0.822**	0.810	<u>0.821</u>	0.810	0.811

Таблица 9: Результаты оценки методов декодирования по метрике RAGAS Faithfulness (GPT-4o).

Набор данных	**Модель**	**Lookback**	**Greedy**	**Ensemble$^+$**	**SFT**	**DPO**
CoQA	Llama-2-7b-chat-hf	0.789	0.707	<u>0.819</u>	0.739	**0.835**
CoQA	Mistral-7B-Instruct-v0.1	<u>0.755</u>	0.532	0.713	0.712	**0.768**
RAGTruth	Llama-2-7b-chat-hf	0.675	0.742	<u>0.749</u>	0.736	**0.753**
RAGTruth	Mistral-7B-Instruct-v0.1	<u>0.700</u>	0.642	0.548	**0.740**	0.519
SQuAD	Llama-2-7b-chat-hf	0.498	0.493	<u>0.516</u>	0.449	**0.540**
SQuAD	Mistral-7B-Instruct-v0.1	<u>0.644</u>	0.588	**0.647**	0.555	0.561

Результаты дообучения представлены в таблицах 8 и 9. Дообучение с использованием DPO демонстрирует наилучшие результаты по метрике Faithfulness на большинстве датасетов для обеих моделей, превосходя как базовые модели, так и модели, дообученные с использованием SFT. По метрике BERTScore DPO также показывает значительное улучшение.

5.2.3 Выводы

Проведенные эксперименты позволяют сделать следующие выводы:

– Предложенный подход, основанный на генерации нескольких ответов с последующим выбором ответа с наименьшим значением ансамбля MTopDiv+Uncertainty, демонстрирует высокую эффективность в уменьшении галлюцинаций. Метод показывает улучшение до 15% по метрике RAGAS Faithfulness относительно базового Greedy декодирования. Корректность работы подтверждается значительным разрывом между Ensemble$^+$ и Ensemble$^-$.

– Дообучение с использованием DPO на автоматически сформированных парах предпочтений (где Ensemble$^+$ используется для положительных примеров, а Ensemble$^-$ – для отрицательных) позволяет достичь наилучших результатов. Улучшение по метрике RAGAS Faithfulness достигает 20% относительно базовой модели.

– Эффективность различных подходов варьируется в зависимости от характеристик датасета и используемой модели. Например, на датасете RAGTruth наблюдаются более скромные улучшения, что может быть связано с его специфическими особенностями.

– Предложенная методология формирования пар предпочтений на основе автоматических метрик детекции галлюцинаций может быть легко адаптирована для использования с другими классификаторами или их комбинациями, что открывает возможности для дальнейшего улучшения результатов. Также перспективным направлением развития является адаптация методики для онлайн обучения.

Результаты экспериментов подтверждают эффективность предложенного двухэтапного подхода: сначала использование ансамблевого метода для выбора лучших генераций, затем дообучение модели на автоматически сформированных парах предпочтений. Такой подход позволяет значительно снизить уровень галлюцинаций в ответах языковых моделей без необходимости ручной разметки данных.

6 Заключение

В данной работе представлена комплексная методология для детекции и минимизации контекстуальных галлюцинаций при ответах больших языковых моделей на вопросы, основанная на white-box подходах с использованием внутренних состояний модели. Исследование охватило как теоретические аспекты проблемы галлюцинаций, так и практические решения для их обнаружения и минимизации.

Научная значимость. Основным вкладом работы является впервые предложенное ансамблирование топологического метода анализа карт внимания с вероятностными метриками неопределенности для необучаемой детекции галлюцинаций, а также разработка методологии автоматического формирования пар предпочтений для DPO-дообучения без использования ручной разметки или проприетарных моделей. Полученные результаты расширяют понимание внутренних механизмов возникновения галлюцинаций в трансформерных моделях и предлагают новые подходы к их контролю.

Практическая значимость. Предложенные методы адаптированы для работы в условиях ограниченных вычислительных ресурсов и данных, что критически важно для индустриальных приложений. Необучаемые подходы обеспечивают масштабируемость применения к любым трансформерным моделям без дополнительного обучения, а обучаемые методы демонстрируют эффективность при использовании всего 250 обучающих примеров. Результаты работы непосредственно применимы для повышения надежности вопросно-ответных RAG-систем и других промышленных решений на основе больших языковых моделей.

Несмотря на достигнутые результаты, работа имеет ряд ограничений, которые определяют направления для будущих исследований:

- *Ограниченность архитектур*: предложенные методы протестированы на моделях семейства трансформеров размером до 10 млрд. параметров. Требуется исследование их применимости к другим архитектурам и современным крупномасштабным моделям.

- *Калибровка ансамблевых методов*: для необучаемых подходов необходима калибровка коэффициентов линейной комбинации для каждой целевой модели, что может усложнять их практическое применение в некоторых сценариях.

- *Ограниченность типов задач*: эксперименты проводились преимущественно на

задачах вопросно-ответного взаимодействия с контекстом. Необходимо расширение исследований на другие типы задач генерации текста, включая суммаризацию, диалоговые системы и творческое письмо.

– *Адаптация к онлайн-обучению*: текущая реализация дообучения предполагает предварительное формирование полного набора пар предпочтений. Перспективным направлением является разработка методов онлайн-дообучения языковых моделей, где пары предпочтений формируются динамически в процессе обучения на основе текущего состояния модели и её ошибок. Такой подход позволит моделям непрерывно адаптироваться к новым типам галлюцинаций и изменяющимся требованиям к фактологической точности без необходимости полного переобучения.

Список литературы

1. Attention is All you Need / A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin // Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA / под ред. I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett. — 2017. — С. 5998—6008. — URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

2. CatBoost: unbiased boosting with categorical features / L. O. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, A. Gulin // Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada / под ред. S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett. — 2018. — С. 6639—6649. — URL: https://proceedings.neurips.cc/paper/2018/hash/14491b756b3a51daac41c24863285549-Abstract.html.

3. *Rajpurkar P.*, *Jia R.*, *Liang P.* Know What You Don't Know: Unanswerable Questions for SQuAD // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers / под ред. I. Gurevych, Y. Miyao. — Association for Computational Linguistics, 2018. — С. 784—789. — DOI: 10.18653/V1/P18-2124. — URL: https://aclanthology.org/P18-2124/.

4. Ranking Generated Summaries by Correctness: An Interesting but Challenging Application for Natural Language Inference / T. Falke, L. F. R. Ribeiro, P. A. Utama, I. Dagan, I. Gurevych // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics / под ред. A. Korhonen, D. Traum, L. Màrquez. — Florence, Italy : Association for Computational Linguistics, 07.2019. — С. 2214—2220. — DOI: 10.18653/v1/P19-1213. — URL: https://aclanthology.org/P19-1213/.

5. *Reddy S.*, *Chen D.*, *Manning C. D.* CoQA: A Conversational Question Answering Challenge // Transactions of the Association for Computational Linguistics / под ред. L. Lee, M. Johnson, B. Roark, A. Nenkova. — Cambridge, MA, 2019. — Т. 7. —

C. 249—266. — DOI: 10.1162/tacl_a_00266. — URL: https://aclanthology.org/Q19-1016/.

6. BERTScore: Evaluating Text Generation with BERT / T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi // 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. — OpenReview.net, 2020. — URL: https://openreview.net/forum?id=SkeHuCVFDr.

7. *McInnes L.*, *Healy J.*, *Melville J.* UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. — 2020. — arXiv: 1802.03426 [stat.ML]. — URL: https://arxiv.org/abs/1802.03426.

8. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks / P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, D. Kiela // Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual / под ред. H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin. — 2020. — URL: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

9. Generating Literal and Implied Subquestions to Fact-check Complex Claims / J. Chen, A. Sriram, E. Choi, G. Durrett // Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing / под ред. Y. Goldberg, Z. Kozareva, Y. Zhang. — Abu Dhabi, United Arab Emirates : Association for Computational Linguistics, 12.2022. — C. 3495—3516. — DOI: 10.18653/v1/2022.emnlp-main.229. — URL: https://aclanthology.org/2022.emnlp-main.229/.

10. *Lin S.*, *Hilton J.*, *Evans O.* TruthfulQA: Measuring How Models Mimic Human Falsehoods // Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) / под ред. S. Muresan, P. Nakov, A. Villavicencio. — Dublin, Ireland : Association for Computational Linguistics, 05.2022. — C. 3214—3252. — DOI: 10.18653/v1/2022.acl-long.229. — URL: https://aclanthology.org/2022.acl-long.229/.

11. Training language models to follow instructions with human feedback / L. Ouyang [и др.] // Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022 / под ред. S. Koyejo, S. Mohamed, A.

Agarwal, D. Belgrave, K. Cho, A. Oh. — 2022. — URL: http://papers.nips.cc/paper%5C_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.

12. Training language models to follow instructions with human feedback / L. Ouyang [и др.]. — 2022. — arXiv: 2203.02155 [cs.CL]. — URL: https://arxiv.org/abs/2203.02155.

13. *Azaria A.*, *Mitchell T.* The Internal State of an LLM Knows When It's Lying // Findings of the Association for Computational Linguistics: EMNLP 2023 / под ред. H. Bouamor, J. Pino, K. Bali. — Singapore : Association for Computational Linguistics, 12.2023. — С. 967—976. — DOI: 10.18653/v1/2023.findings-emnlp.68. — URL: https://aclanthology.org/2023.findings-emnlp.68/.

14. *Chen Z.*, *Feng Y.*, *Zhao D.* From the One, Judge of the Whole: Typed Entailment Graph Construction with Predicate Generation // Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) / под ред. A. Rogers, J. Boyd-Graber, N. Okazaki. — Toronto, Canada : Association for Computational Linguistics, 07.2023. — С. 3534—3551. — DOI: 10.18653/v1/2023.acl-long.196. — URL: https://aclanthology.org/2023.acl-long.196/.

15. Direct Preference Optimization: Your Language Model is Secretly a Reward Model / R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, C. Finn // Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023 / под ред. A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine. — 2023. — URL: http://papers.nips.cc/paper%5C_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

16. Eliciting Latent Predictions from Transformers with the Tuned Lens / N. Belrose, Z. Furman, L. Smith, D. Halawi, I. Ostrovsky, L. McKinney, S. Biderman, J. Steinhardt // CoRR. — 2023. — Т. abs/2303.08112. — DOI: 10.48550/ARXIV.2303.08112. — arXiv: 2303.08112. — URL: https://doi.org/10.48550/arXiv.2303.08112.

17. FActScore: Fine-grained Atomic Evaluation of Factual Precision in Long Form Text Generation / S. Min, K. Krishna, X. Lyu, M. Lewis, W.-t. Yih, P. Koh,

M. Iyyer, L. Zettlemoyer, H. Hajishirzi // Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. — Singapore : Association for Computational Linguistics, 12.2023. — С. 12076—12100. — DOI: 10.18653/v1/2023.emnlp-main.741. — URL: https://aclanthology.org/2023.emnlp-main.741/.

18. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena / L. Zheng, W. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, I. Stoica // CoRR. — 2023. — Т. abs/2306.05685. — DOI: 10.48550/ARXIV.2306.05685. — arXiv: 2306.05685. — URL: https://doi.org/10.48550/arXiv.2306.05685.

19. *Kuhn L.*, *Gal Y.*, *Farquhar S.* Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation // The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. — OpenReview.net, 2023. — URL: https://openreview.net/forum?id=VD-AYtP0dve.

20. *Manakul P.*, *Liusie A.*, *Gales M. J. F.* SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models // Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. — Singapore : Association for Computational Linguistics, 12.2023. — С. 9004—9017. — DOI: 10.18653/v1/2023.emnlp-main.557. — URL: https://aclanthology.org/2023.emnlp-main.557/.

21. *OpenAI.* GPT-4 Technical Report : тех. отч. / OpenAI. — 2023. — URL: https://arxiv.org/abs/2303.08774.

22. Survey of Hallucination in Natural Language Generation / Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, P. Fung // ACM Computing Surveys. — 2023. — Март. — Т. 55, № 12. — С. 1—38. — ISSN 1557-7341. — DOI: 10.1145/3571730. — URL: http://dx.doi.org/10.1145/3571730.

23. TabPFN: A Transformer That Solves Small Tabular Classification Problems in a Second / N. Hollmann, S. Müller, K. Eggensperger, F. Hutter // The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. — OpenReview.net, 2023. — URL: https://openreview.net/forum?id=cp5PvcI6w8%5C_.

24. The Troubling Emergence of Hallucination in Large Language Models - An Extensive Definition, Quantification, and Prescriptive Remediations / V. Rawte, S. Chakraborty, A. Pathak, A. Sarkar, S. T. I. Tonmoy, A. Chadha, A. Sheth, A. Das // Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing / под ред. H. Bouamor, J. Pino, K. Bali. — Singapore : Association for Computational Linguistics, 12.2023. — С. 2541—2573. — DOI: 10.18653/v1/2023.emnlp-main.155. — URL: https://aclanthology.org/2023.emnlp-main.155/.

25. *Belikova J.*, *Beliakin E.*, *Konovalov V.* JellyBell at TextGraphs-17 Shared Task: Fusing Large Language Models with External Knowledge for Enhanced Question Answering // Proceedings of TextGraphs-17: Graph-based Methods for Natural Language Processing / под ред. D. Ustalov, Y. Gao, A. Panchenko, E. Tutubalina, I. Nikishina, A. Ramesh, A. Sakhovskiy, R. Usbeck, G. Penn, M. Valentino. — Bangkok, Thailand : Association for Computational Linguistics, 08.2024. — С. 154—160. — URL: https://aclanthology.org/2024.textgraphs-1.15/.

26. Chain-of-Verification Reduces Hallucination in Large Language Models / S. Dhuliawala, M. Komeili, J. Xu, R. Raileanu, X. Li, A. Celikyilmaz, J. Weston // Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024 / под ред. L. Ku, A. Martins, V. Srikumar. — Association for Computational Linguistics, 2024. — С. 3563—3578. — DOI: 10.18653/V1/2024.FINDINGS-ACL.212. — URL: https://doi.org/10.18653/v1/2024.findings-acl.212.

27. Detecting hallucinations in large language models using semantic entropy / S. Farquhar, J. Kossen, L. Kuhn, Y. Gal // Nature. — 2024. — Т. 630, № 8017. — С. 625—630. — DOI: 10.1038/s41586-024-07421-0.

28. Do Androids Know They‘re Only Dreaming of Electric Sheep? / S. CH-Wang, B. Van Durme, J. Eisner, C. Kedzie // Findings of the Association for Computational Linguistics: ACL 2024 / под ред. L.-W. Ku, A. Martins, V. Srikumar. — Bangkok, Thailand : Association for Computational Linguistics, 08.2024. — С. 4401—4420. — DOI: 10.18653/v1/2024.findings-acl.260. — URL: https://aclanthology.org/2024.findings-acl.260/.

29. DoLa: Decoding by Contrasting Layers Improves Factuality in Large Language Models / Y.-S. Chuang, Y. Xie, H. Luo, Y. Kim, J. Glass, P. He // Proceedings of the International Conference on Learning Representations (ICLR). — 2024.

30. Fine-grained Hallucination Detection and Editing for Language Models / A. Mishra, A. Asai, V. Balachandran, Y. Wang, G. Neubig, Y. Tsvetkov, H. Hajishirzi // CoRR. — 2024. — T. abs/2401.06855. — DOI: 10.48550/ARXIV.2401.06855. — arXiv: 2401.06855. — URL: https://doi.org/10.48550/arXiv.2401.06855.

31. *Friel R.*, *Belyi M.*, *Sanyal A.* RAGBench: Explainable Benchmark for Retrieval-Augmented Generation Systems // CoRR. — 2024. — T. abs/2407.11005. — DOI: 10.48550/ARXIV.2407.11005. — arXiv: 2407.11005. — URL: https://doi.org/10.48550/arXiv.2407.11005.

32. Gemma 2: Improving open language models at a practical size / G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé [и др.] // arXiv preprint arXiv:2408.00118. — 2024.

33. GPT-4 Technical Report / OpenAI [и др.]. — 2024. — arXiv: 2303.08774 [cs.CL]. — URL: https://arxiv.org/abs/2303.08774.

34. INSIDE: LLMs' Internal States Retain the Power of Hallucination Detection / C. Chen, K. Liu, Z. Chen, Y. Gu, Y. Wu, M. Tao, Z. Fu, J. Ye. — 2024. — arXiv: 2402.03744 [cs.CL].

35. Lookback Lens: Detecting and Mitigating Contextual Hallucinations in Large Language Models Using Only Attention Maps / Y.-S. Chuang, L. Qiu, C.-Y. Hsieh, R. Krishna, Y. Kim, J. R. Glass // Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing / под ред. Y. Al-Onaizan, M. Bansal, Y.-N. Chen. — Miami, Florida, USA : Association for Computational Linguistics, 11.2024. — С. 1419—1436. — DOI: 10.18653/v1/2024.emnlp-main.84. — URL: https://aclanthology.org/2024.emnlp-main.84/.

36. Lookback Lens: Detecting and Mitigating Contextual Hallucinations in Large Language Models Using Only Attention Maps / Y. Chuang, L. Qiu, C. Hsieh, R. Krishna, Y. Kim, J. R. Glass // Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024 / под ред. Y. Al-Onaizan, M. Bansal, Y. Chen. — Association for Computational Linguistics, 2024. — С. 1419—1436. — URL: https://aclanthology.org/2024.emnlp-main.84.

37. Qwen2.5 Technical Report / A. Yang [и др.] // arXiv preprint arXiv:2412.15115. — 2024.

38. R-Tuning: Instructing Large Language Models to Say ‘I Don‘t Know’ / H. Zhang, S. Diao, Y. Lin, Y. Fung, Q. Lian, X. Wang, Y. Chen, H. Ji, T. Zhang // Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers) / под ред. K. Duh, H. Gomez, S. Bethard. — Mexico City, Mexico : Association for Computational Linguistics, 06.2024. — С. 7113—7139. — DOI: 10.18653/v1/2024.naacl-long.394. — URL: https://aclanthology.org/2024.naacl-long.394/.

39. RAGAs: Automated Evaluation of Retrieval Augmented Generation / S. Es, J. James, L. Espinosa Anke, S. Schockaert // Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations / под ред. N. Aletras, O. De Clercq. — St. Julians, Malta : Association for Computational Linguistics, 03.2024. — С. 150—158. — URL: https://aclanthology.org/2024.eacl-demo.16/.

40. RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models / C. Niu, Y. Wu, J. Zhu, S. Xu, K. Shum, R. Zhong, J. Song, T. Zhang // Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) / под ред. L.-W. Ku, A. Martins, V. Srikumar. — Bangkok, Thailand : Association for Computational Linguistics, 08.2024. — С. 10862—10878. — DOI: 10.18653/v1/2024.acl-long.585. — URL: https://aclanthology.org/2024.acl-long.585/.

41. The llama 3 herd of models / A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan [и др.] // arXiv preprint arXiv:2407.21783. — 2024.

42. Trusting Your Evidence: Hallucinate Less with Context-aware Decoding / W. Shi, X. Han, M. Lewis, Y. Tsvetkov, L. Zettlemoyer, W.-t. Yih // Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers) / под ред. K. Duh, H. Gomez, S. Bethard. — Mexico City, Mexico : Association for Computational Linguistics, 06.2024. — С. 783—791. — DOI: 10.18653/v1/2024.naacl-short.69. — URL: https://aclanthology.org/2024.naacl-short.69/.

43. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions / L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng,

H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, T. Liu // ACM Transactions on Information Systems. — 2025. — Янв. — Т. 43, № 2. — С. 1—55. — ISSN 1558-2868. — DOI: 10.1145/3703155. — URL: http://dx.doi.org/10.1145/3703155.

44. Data-efficient Meta-Models for Evaluation of Context-Based Questions and Answers in LLMs / J. Belikova, K. Polev, R. Parchiev, D. Simakov // Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval. — Padua, Italy : Association for Computing Machinery, 07.2025. — (SIGIR '25). — Accepted for publication.

45. *DeepSeek-AI.* DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. — 2025. — arXiv: 2501.12948 [cs.CL]. — URL: https://arxiv.org/abs/2501.12948.

46. Hallucination Detection in LLMs with Topological Divergence on Attention Graphs / A. Bazarova, A. Yugay, A. Shulga, A. Ermilova, A. Volodichev, K. Polev, J. Belikova, R. Parchiev, D. Simakov, M. Savchenko, A. Savchenko, S. Barannikov, A. Zaytsev. — 2025. — arXiv: 2504.10063 [cs.CL]. — URL: https://arxiv.org/abs/2504.10063.

47. *Kovács Á., Recski G.* LettuceDetect: A Hallucination Detection Framework for RAG Applications. — 2025. — arXiv: 2502.17125 [cs.CL]. — URL: https://arxiv.org/abs/2502.17125.

48. Luna: A Lightweight Evaluation Model to Catch Language Model Hallucinations with High Accuracy and Low Cost / M. Belyi, R. Friel, S. Shao, A. Sanyal // Proceedings of the 31st International Conference on Computational Linguistics: Industry Track / под ред. O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, S. Schockaert, K. Darwish, A. Agarwal. — Abu Dhabi, UAE : Association for Computational Linguistics, 01.2025. — С. 398—409. — URL: https://aclanthology.org/2025.coling-industry.34/.

49. Medical Hallucinations in Foundation Models and Their Impact on Healthcare / Y. Kim [и др.] // CoRR. — 2025. — Т. abs/2503.05777. — DOI: 10.48550/ARXIV.2503.05777. — arXiv: 2503.05777. — URL: https://doi.org/10.48550/arXiv.2503.05777.