

ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ/INFORMATICS AND INFORMATION PROCESSES

DOI: https://doi.org/10.60797/IRJ.2026.169.108 EDN: YXBRJH

УПРАВЛЕНИЕ ГЕНЕРАТИВНЫМИ ХАРАКТЕРИСТИКАМИ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ С ПОМОЩЬЮ ПАРАМЕТРОВ TEMPERATURE, TOP_P, FREQUENCY PENALTY И PRESENCE PENALTY

Научная статья

Царёв Р.Ю.[1, *], Бочаров М.И.[2]
[1] ORCID : 0000-0002-6740-1840;
[2] ORCID : 0000-0002-3356-3251;
[1, 2] МИРЭА – Российский технологический университет, Москва, Российская Федерация

* Корреспондирующий автор (tsarev.sfu[at]mail.ru)

Предложена: 14.06.2026; Принята: 06.07.2026; Опубликована: 17.07.2026

Аннотация

Современные большие языковые модели применяются в различных областях науки и техники. Как правило, для пользователя они представляют собой черный ящик. Однако существует возможность управлять языковыми моделями через параметры генерации. В статье рассмотрены такие параметры генерации как temperature, top_p, frequency penalty и presence penalty. Дано их описание, характеристики, диапазоны возможных значений. Приведен код программы на Python, который не только демонстрирует применение API при установке значений параметров генерации и запросе к языковой модели, но и позволяет воспроизвести эксперимент. В статье представлен пример влияния различных наборов значений параметров генерации на результат работы большой языковой модели. На основе проведенного эксперимента формализованы три режима генерации (консервативный, умеренный, творческий) с соответствующими диапазонами параметров. Представленные результаты позволяют произвести выбор требуемых значений параметров для конкретной целевой задачи.

Ключевые слова: LLM, промпт, параметры генерации, API, образовательные задачи.

MANAGEMENT OF THE GENERATIVE TRAITS OF LARGE LANGUAGE MODELS USING THE PARAMETERS ‘TEMPERATURE’, ‘TOP_P’, ‘FREQUENCY PENALTY’ AND ‘PRESENCE PENALTY’

Research article

Caryov R.Y.[1, *], Bocharov M.[2]
[1] ORCID : 0000-0002-6740-1840;
[2] ORCID : 0000-0002-3356-3251;
[1, 2] MIREA – Russian Technological University, Moscow, Russian Federation

* Corresponding author (tsarev.sfu[at]mail.ru)

Suggested: 14.06.2026; Accepted: 06.07.2026; Published: 17.07.2026

Abstract

Modern large language models are used in various fields of science and technology. Typically, they appear as a black box to the user. However, it is possible to control language models through generation parameters. This paper discusses generation parameters such as temperature, top_p, frequency penalty, and presence penalty. It provides descriptions, characteristics, and ranges of possible values. Python code is provided that not only demonstrates the use of the API when setting generation parameter values and making requests to the language model, but also allows the experiment to be replicated. The paper presents an example of how different sets of generation parameter values affect the performance of a large language model. Based on the experiment, three generation modes (conservative, moderate, creative) are formalized with corresponding parameter ranges. The presented results allow for the selection of the required parameter values for a specific objective.

Keywords: LLM, prompt, generation parameters, API, educational tasks.

Введение

Большие языковые модели (от англ. Large Language Models, LLM) активно входят в обиход и используются для решения крайне широкого спектра задач, начиная от бытовых и заканчивая наукоемкими и узкоспециализированными [1], [2]. Такие современные LLM, как GigaChat, Алиса, DeepSeek, ChatGPT демонстрируют высокую эффективность и позволяют автоматизировать различные процедуры [3], [4]. Областями применения больших языковых моделей выступают программирование, анализ данных, образование и педагогика, финансы и маркетинг, перевод и локализация, научные исследования [5], [6].

На основе текстового запроса (промпта) к соответствующей языковой модели она генерирует текстовый ответ, изображение, видео [7], [8]. Сама LLM для пользователя выглядит как чёрный ящик. Большинство пользователей взаимодействуют с большими языковыми моделями посредством веб-интерфейса через чат. Пользователь вводит промпт и получает от языковой модели результат. Если выданный LLM результат не устраивает пользователя, он может сделать повторный запрос и получит новую выдачу. При таком подходе отклик модели отличается шаблонностью и непредсказуемостью, а ответ может потребовать донастройки. Все параметры при этом скрыты от пользователя и установлены на усредненных значениях разработчиками модели.

При этом возможность влиять на результат работы языковой модели в отдельных областях является необходимостью. В частности это относится к академической среде и профессиональному контексту, для которых форма ответа, степень его детализации и вариативность, а также воспроизводимость столь же важна, как его смысловое содержание и истинность [9], [10], [11]. Как отмечается в [12], эффективная реализация образовательных и научных процессов невозможна без современной научно-образовательной инфраструктуры. Важным элементом такой инфраструктуры являются инструменты цифровой обработки и преобразования текстов на основе больших языковых моделей. В частности, при генерации типовых документов требуется высокая точность, унификация, детерминированность, а при поиске новых идей, напротив, максимальная разнообразность, вариативность, в допустимой степени непредсказуемость.

Таким образом, возникает необходимость управления результатами работы больших языковых моделей, которое выходит за рамки ограничений, наложенных на промпт. Решение этой задачи возможно с помощью использования API, предоставляемого современными LLM. Изменяя значения параметров, можно влиять на процесс генерации, адаптируя поведения модели под конкретную задачу.

Целью данной работы является описание основных параметров генерации больших языковых моделей: temperature, top_p, frequency penalty и presence penalty и оценка их влияния на результат работы LLM.

Методы и принципы исследования

Работа с большими языковыми моделями (LLM) осуществляется посредством запросов — промптов. Генерация текста в ответ на запрос происходит последовательно таким образом, что следующий токен, которым может быть как целое слово, так и его часть, появляется с определенной вероятностью на основе предыдущего контекста. То есть в основе лежит вероятностное распределение следующего токена. На это распределение можно влиять, изменяя значения параметров генерации.

Работу больших языковых моделей можно описать следующим образом. В ее основе лежит авторегрессионная генерация: на каждом шаге модель вычисляет вероятности для всех возможных токенов, которые могут последовать далее в выдаче, учитывая как промпт, так и уже сгенерированную часть ответа. Затем она выбирает один из них, после чего добавляет к уже сгенерированному тексту. Модель повторяет эту последовательность действий, пока ответ не будет сгенерирован полностью. Именно на выбор токенов можно влиять посредством параметров генерации модели.

Ключевым параметром является температура (temperature). Он отвечает за случайность или креативность ответа LLM. Его значение лежит в интервале от 0 до 2. При малом значении этого параметра на выход поступает наиболее очевидный ответ, при большом, напротив, более спонтанный или непредсказуемый.

Параметр ядерной выборки (от англ. nucleus sampling) – top_p – позволяет отбирать такое количество вероятных слов, чтобы их суммарная вероятность достигла установленного значения top_p. Пробегая по отсортированному в порядке убывания вероятности появления в ответе множеству слов, модель формирует выборку, из которой впоследствии случайным образом делает окончательный выбор. Значения данного параметра варьируются от 0 до 1, что фактически соответствует сумме вероятностей. Например, при top_p = 0,1 модель будет рассматривать те слова, сумма вероятностей появления которых в ответе достигает 10%.

Другие два важных параметра, которые позволяют управлять большими языковыми моделями — это параметры штрафов: штраф за частоту и штраф за присутствие.

Штраф за частоту (frequency penalty) отвечает за разнообразие лексики, за отсутствие повторов. Имея диапазон возможных значений от –2 до 2, чем выше установлено значение этого параметра, тем активнее модель ищет синонимы и альтернативные формулировки, старается избегать повторений.

Штраф за присутствие (presence penalty), этот параметр в отличие от предыдущего штрафует за присутствие одних и тех же тем, а не слов. Frequency penalty штрафует модель за лексический повтор, т.е. за повтор слов. Presence penalty штрафует модель за смысловой повтор, даже если слова разные. Стоит отметить, что presence penalty штрафует модель в том случае, если она ушла от темы, а в последствии опять к ней вернулась. Если же выдача происходит в рамках одной темы, то штраф не накладывается. Значение параметра presence penalty находится также в интервале от –2 до 2. Для данного параметра, равно как и для frequency penalty, установка отрицательных значений будет стимулировать повторение, что может быть важным для некоторых задач.

Приведенные интервалы параметров были окончательно стандартизированы в спецификациях OpenAI и поддерживаются большинством современных API. В отдельных больших языковых моделях могут быть определены другие границы, но для работы с наиболее известными LLM, такими, как ChatGPT, DeepSeek, GigaChat, актуальны именно эти значения.

Приведенные параметры генерации являются универсальными и наиболее часто используемыми. Однако, есть и другие параметры, которые позволяют осуществлять тонкую настройку моделей, например, Top_k, Seed, Max tokens, Stop sequences, Repetition penalty, Min_p, Best_of / n, Logit bias, Response schema.

Для того чтобы задать параметры генерации для большой языковой модели, используется API. Схематично код на языке программирования Python представлен на рис. 1.


```python
from openai import OpenAI

# Подключение к API GitHub Models
client = OpenAI(
    api_key="<API_TOKEN>",
    base_url="https://models.github.ai/inference"
)

response = client.chat.completions.create(
    model="openai/gpt-4.1",
    messages=[{"role": "user", "content": "Объясни, что такое ИИ."}],
    temperature=0.7,
    top_p=0.9,
    frequency_penalty=0.6,
    presence_penalty=0.3
)

print(response.choices[0].message.content)
```

Рисунок 1 - Код программы с установленными значениями параметров генерации

DOI: https://doi.org/10.60797/IRJ.2026.169.108.1

На рис. 1 приведен один из возможных вариантов настройки ответов LLM. Данный код может быть выполнен в любой среде, поддерживающей Python, например, Google Colab или Jupyter Motebook. Для этих же целей могут быть использованы и другие модели и сервисы, которые поддерживают OpenAI-совместимый API. К ним относятся GitHub Models, Groq, Mistral AI, OpenRouter и модели, локально развертываемые через vLLM. При этом достаточно заменить значения переменных base_url, api_key и model.

Основные результаты

В данном разделе приведен пример использования большой языковой модели с двумя разными наборами значений параметров генерации. Значения были выбраны такими, чтобы проиллюстрировать влияние параметров temperature, top_p, frequency penalty и presence penalty на выдачу языковой модели.

На вход LLM был подан промпт: «Придумай 3 креативных метафоры, объясняющих, что такое искусственный интеллект. Метафоры должны быть неожиданными, яркими и запоминающимися».

В первом случае были заданы следующие значения параметров: temperature = 0,1, top_p = 0,2, frequency_penalty = 0, presence_penalty = 0

Во втором случае, при том же самом промпте, были заданы следующие параметры: temperature = 0,9, top_p = 0,9, frequency_penalty = 0,6, presence_penalty = 0,4.

Результаты работы LLM при различных наборах параметров приведены ниже.

Ответ языковой модели при наборе значений параметров генерации 1:

1. Искусственный интеллект — это цифровой алхимик, который превращает сырые данные в золото знаний, создавая из хаоса структурированные идеи и решения, как древние мастера искали философский камень.

2. Искусственный интеллект — это невидимый дирижёр, который управляет симфонией информации, заставляя каждый инструмент (данные) звучать в унисон, создавая гармонию из разрозненных мелодий.

3. Искусственный интеллект — это волшебный зеркальный лабиринт, отражающий наши мысли и желания, но при этом открывающий новые проходы и перспективы, которые мы сами не могли бы увидеть, если бы не его магия.

Ответ языковой модели при наборе значений параметров генерации 2:

1. Искусственный интеллект — это кулинарный шеф-повар на кухне нашего разума, который смешивает ингредиенты из опыта и данных, чтобы создать уникальные блюда из решений и предсказаний. Каждое блюдо — это новый взгляд на привычные проблемы.

2. Искусственный интеллект — это цифровой оркестр, где каждую ноту играют алгоритмы, а гармония создаётся из миллиардов данных. Как виртуозы, они интерпретируют мир, превращая шум информации в симфонию знаний.

3. Искусственный интеллект — это волшебник с цифровой палочкой, который превращает хаос в порядок, оживляя мёртвые строки кода и делая их понимаемыми для нас. Он колдует над данными, создавая чудеса из рутинной работы.

Проводя исследования влияния параметра температуры при фиксированных значениях других параметров top_p = 0,9, frequency_penalty = 0, presence_penalty = 0, было выявлено, что низкое значение temperature = 0,1 приводит к выдаче консервативного, классического, стандартного ответа модели. При высоком значении temperature, например, 1,2 ответы становятся более неожиданными, красочными, креативными и нестандартными.

В рамках исследований, экспериментально было установлено, что распределения вероятностей токенов от температуры зависит от модели. Были протестированы модели GPT-4.1, GPT-4o, GPT-4o-mini, DeepSeek-R1-0528. Среди них только GPT-4.1 показала монотонный рост энтропии с увеличением температуры (рис. 2). Модели GPT-4o и GPT-4o-mini оказались детерминированными, энтропия была постоянной независимо от изменения температуры в диапазоне от 0,0 до 2,0. Модель DeepSeek-R1 реагировала на температуру лишь при значении temperature > 1,4.

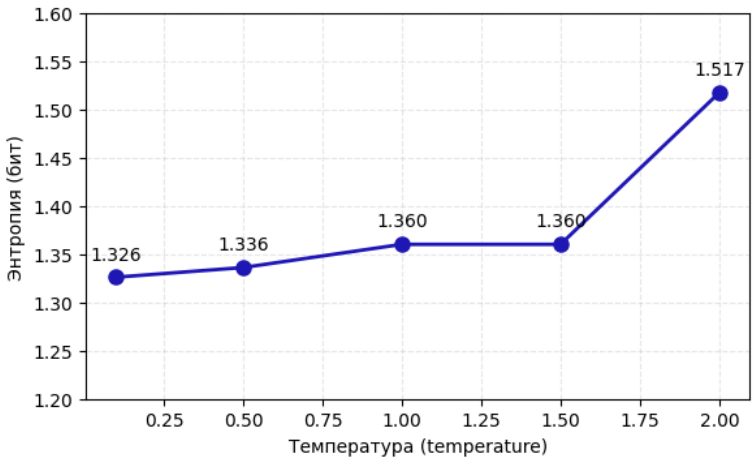

Рисунок 2 - Зависимость энтропии распределения вероятностей первых токенов от температуры для модели GPT-4.1

DOI: https://doi.org/10.60797/IRJ.2026.169.108.2

На рис. 3 представлена визуальная иллюстрация принципа применения параметра top_p. Здесь значения по оси абсцисс соответствуют номерам токенов, расположенных в порядке убывания вероятности их выбора, а по оси ординат — накопленная вероятность. Точками показано увеличение суммы вероятностей при последовательном добавлении токенов. Пунктирными линиями обозначены пороговые значения параметра генерации top_p: 0,2, 0,5, 0,9. В данном примере при пороговом значении равном 0,2 в ядро выборки попадает лишь один токен. При top_p = 0,5 в ядро выборки попадают два токена. При пороговом значении top_p, равном 0,9 в ядро попадает девять токенов.

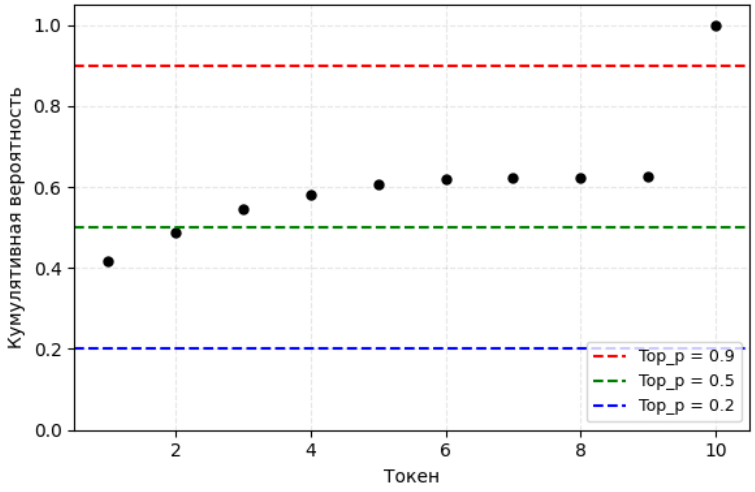

Рисунок 3 - Формирование ядра выборки (nucleus sampling)

DOI: https://doi.org/10.60797/IRJ.2026.169.108.3

На основе проведенных исследований можно формализовать три набора значений параметров генерации, представленных в таблице 1.

Таблица 1 - Наборы значений параметров генерации

DOI: https://doi.org/10.60797/IRJ.2026.169.108.4

Название	temperature	top_p	frequency_penalty	presence_penalty
Консервативный	0,0–0,2	0,1–0,3	0	0
Умеренный	0,4–0,6	0,5–0,7	0,3–0,5	0
Творческий	0,8–1,2	0,8–0,95	0,6–1,0	0,4–0,8

Обсуждение

Полученные результаты демонстрируют влияние параметров генерации как на стилистические, так и семантические характеристики результата работы большой языковой модели. При первом наборе значений параметров модель более консервативна, формальна, ее ответы более структурированы. При втором же наборе модель становится более творческой и гибкой, ее ответы более динамичны и разнообразны.

При первом наборе, где параметры отличаются малыми значениями (temperature = 0,1, top_p = 0,2, штрафы = 0) прослеживается единый синтаксический шаблон. Генерацию можно охарактеризовать как детерминированную и предсказуемую. Наблюдается высокая когерентность: все описания выдержаны в едином ключе, создается впечатление единого текста. Это достигается за счет низкой температуры (стохастичность минимальна) и малого значения top_p (модель выбирает только среди максимально вероятных слов).

При втором наборе, параметры имеют большие значения: temperature = 0,9, top_p = 0,9, frequency_penalty = 0,6, presence_penalty = 0,4. Стиль ответов модели радикально отличается, он изобилует метафорами, ответы становятся тематически разнородными. Presence penalty, который штрафует модель за возврат к уже затронутым темам, а также frequency penalty, который стимулирует лексическое разнообразие, заставляют модель переключаться между тематическими доменами. Ответы становятся эклектичными, представляя собой смесь разнородных элементов, единая линия повествования отсутствует.

Целесообразно устанавливать низкое значение температуры, если необходимо получить инструкцию, описание стандартной процедуры, высокое же значение больше подойдет для мозгового штурма или дискуссий.

Различная реакция исследуемых моделей на изменение температуры объясняется особенностями их архитектуры. GPT-4.1 демонстрирует сглаживание распределения вероятностей, что делает целесообразным ее применение в задачах, требующих управляемой стохастичности. Модели GPT-4o и GPT-4o-mini, напротив, отличаются детерминизмом ответов. Их лучше использовать в задачах, где требуется единообразие ответов при одинаковых исходных данных. DeepSeek-R1 ориентирована, прежде всего, на логические рассуждения. Вариативность ответов проявляется в ней только при больших значениях параметра температуры.

С помощью параметра top_p можно влиять на ширину выбора LLM. При низком значении этого параметра, например, top_p = 0,2 модель дает предсказуемый результат, имея в своем распоряжении довольно ограниченный набор токенов, из которых можно произвести выбор. Результаты в этом случае слабо различимы. При высоком значении, например, top_p = 0,9 модель охватывает гораздо больший набор токенов, что позволяет получить большее разнообразие в ответах, включая те, что основаны на редких токенах. Важно, что математическая или структурная основа при этом сохраняется полностью. Таким образом, низкие значения параметра top_p рекомендуются для отработки алгоритмов или стандартных процедур, а высокие — для генерации разнородных заданий, например, с целью исключения списывания студентами друг у друга при выполнении заданий или прохождения тестов. Представленный пример показывает, что небольшие значения параметра top_p существенно ограничивают модель в выборе возможного токена, в то время как большие значения существенно расширяют пространство выбора.

С помощью параметров штрафов frequency penalty и presence penalty можно влиять на шаблонность ответов. Так, при отсутствии штрафов модель может генерировать однотипные ответы. Если же установить значение параметра frequency penalty равным 0,8, то модель будет вынуждена разнообразить лексику. Параметр же presence penalty, установленный, например, равным 0,6 заставит модель переключаться между различными тематическими областями при генерации ответов. То есть параметр frequency penalty обогащает ответ лексически, а presence penalty – способствует тематическому разнообразию.

Набор «Консервативный» хорошо подходит для генерации инструкций, типовых описаний, критериев. Он обеспечивает точность и воспроизводимость. При малых значениях temperature и top_p, а также нулевых штрафах, результат становится детерминированным.

Набор «Умеренный» целесообразно применять при генерации пояснений или однотипных вариантов. Он обладает большим разнообразием по сравнению с предыдущим набором, однако не выходит за определенные рамки. В целом данный набор отличается балансом между излишне формальными ответами, с одной стороны, и чрезвычайно яркими — с другой.

Наконец, набор «Творческий» обладает наибольшим потенциалом для получения нестандартных, нетривиальных и необычных результатов генерации. За счет высокой температуры, широкого окна выбора, определяемого параметром top_p, а также штрафов за лексическое и тематическое однообразие, LLM будет генерировать максимально разнообразные ответы. Этот режим может быть актуальным при формировании различных задач и примеров или при мозговом штурме.

Малое значение температуры ориентирует модель на моду распределения, LLM выбирает самые вероятные слова, то есть наиболее общепринятые или стандартные для данного контекста. Большие значения температуры смещают выбор модели в сторону более редких токенов, что и характеризует модель как более творческую, а ее результаты делает более неожиданными.

Дополнительный вклад в результат большой языковой модели вносят штрафы: frequency penalty заставляет модель избегать лексических повторов, а presence penalty стимулирует модель на смену тем.

Заключение

Возможность изменения значений параметров генерации позволяет пользователю настраивать работу большой языковой модели для конкретной задачи, что существенно расширяет диапазон возможного применения LLM и позволяет повысить эффективность решения задач с их помощью.

В данной статье были рассмотрены основные и наиболее часто используемые параметры генерации, такие как temperature, top_p, frequency penalty и presence penalty. Было показано, как их изменение влияет на конечный результат работы большой языковой модели.

Проведенный анализ результатов, а также описание интервалов значений каждого из параметров позволяет сделать осознанный выбор необходимых значений для целевой задачи.

В статье показан наглядный пример использования API при установке требуемых значений параметров генерации и запросе к языковой модели. Приведенный код на языке программирования Python является воспроизводимым и позволяет его использовать для других задач, изменив значения параметров генерации и содержание запроса — промпта. При необходимости можно локально установить модель, инструкция для этого также представлена в статье.

На примере показано влияние изменения значений параметров генерации на стиль, структуру и лексическое разнообразием ответов модели. Так, при малых значениях параметров temperature и top_p модель генерирует достаточно консервативные, формальные и однообразные ответы. Такой набор значений параметров может быть использован для задач, требующих точности, определенности, предсказуемости. При более высоких значениях этих параметров и штрафах на повторение модель становится более творческой и гибкой, а ответы приобретают динамичный и произвольный характер, отличаясь гораздо большим разнообразием. Такие значения параметров целесообразно устанавливать при мозговом штурме или при генерации новых идей.

Представленное в работе описание параметров генерации и их характерных может быть полезен при выборе значения параметров для решения конкретной задачи. Небольшие значения параметров temperature (0–0,2) и top_p (0,1–0,3) при нулевых штрафах обеспечивают точность, предсказуемость и детерминизм результатов генерации. Такой набор значений параметров целесообразно использовать при генерации типовых процедур, инструкций, примеров правильных ответов. Большие значения параметров temperature (0,8–1,2) и top_p (0,8–0,95) при штрафах frequency penalty (0,6–1) и presence penalty (0,4–0,8) приводят к разнообразным неординарным ответам, что может быть важным при мозговом штурме или разработке ситуативных задач. Промежуточные значения указанных параметров генерации представляют собой сбалансированный режим, пригодный для большинства типовых учебных и методических заданий.

Дальнейшие исследования могут быть направлены на изучение других комбинаций значений параметров генерации и их влияния на результат работы LLM.

Конфликт интересов

Не указан.

Рецензия

Все статьи проходят рецензирование. Но рецензент или автор статьи предпочли не публиковать рецензию к этой статье в открытом доступе. Рецензия может быть предоставлена компетентным органам по запросу.

Conflict of Interest

None declared.

Review

All articles are peer-reviewed. But the reviewer or the author of the article chose not to publish a review of this article in the public domain. The review can be provided to the competent authorities upon request.

Список литературы / References

1. Бесфамильная Е.М. Контекстно-зависимая маршрутизация запросов к LLM в гибридных кластерах с контролем затрат / Е.М. Бесфамильная, А.В. Смирнов, О.М. Минаев // Мягкие измерения и вычисления. — 2025. — 97(12-2). — С. 96–103. — URL: https://s-lib.com/issues/smc_2025_12_2_a11/ (дата обращения: 15.06.26). — DOI: 10.36871/2618-9976.2025.12-2.011

2. Shaykhulova A. Enhancing engineering decision-support with efficiently fine-tuned reasoning LLMs / A. Shaykhulova // International Research Journal. — 2025. — 12(162). — URL: https://research-journal.org/archive/12-162-2025-december/10.60797/IRJ.2025.162.125 (accessed: 15.06.26). — DOI: 10.60797/IRJ.2025.162.125

3. Вакушин А.А. Проектирование многокомпонентных имитационных моделей с помощью большой языковой модели GPT-4 / А.А. Вакушин, Б.И. Клебанов // Инженерный вестник Дона. — 2024. — 7(115). — С. 174–186. — URL: https://www.ivdon.ru/ru/magazine/archive/n7y2024/9340 (дата обращения: 15.06.26).

4. Резцов С.М. Сравнительный анализ языковых моделей в обработке неструктурированных данных на примере Deepseek и GigaChat / С.М. Резцов // Парадигма. — 2025. — 5. — С. 200–204.

5. Адилжанова С.А. Применение LLM в кибербезопасности: обзор приложений и уязвимостей LLM / С.А. Адилжанова, А.Н. Құрасбек, М.О. Кенжебаева // Вестник Академии гражданской авиации. — 2025. — 38(3). — С. 118–136. — URL: https://vestnik.agakaz.kz/primenenie-llm-v-kiberbezopasnosti-obzor-prilozhenij-i-uyazvimostej-llm/ (дата обращения: 15.06.26). — DOI: 10.53364/24138614_2025_38_3_10

6. Васильев И.М. Влияние LLM-ассистентов на снижение операционной нагрузки в контакт-центрах российских банков / И.М. Васильев // Финансовый бизнес. — 2025. — 12(270). — С. 131–136. — URL: https://fin-biz.ru/gallery/журнал%20Финансовый%20бизнес%202025-12.pdf (дата обращения: 15.06.26).

7. Будаева Д.А. Прагмалингвистические аспекты формулирования промптов для ChatGPT в научной коммуникации / Д.А. Будаева, И.Н. Зырянова // Казанская наука. — 2024. — 7. — С. 220–224. — URL: https://www.kazanscience.ru/files/KH_7_2024.php (дата обращения: 15.06.26).

8. Фатхулин Т.Д. Анализ влияния составляемых текстовых запросов (промптов) на качество изображений, генерируемых нейросетевыми технологиями / Т.Д. Фатхулин, Д.А. Смирнов, И.В. Разумов и др. // Системы синхронизации, формирования и обработки сигналов. — 2024. — 15(2). — С. 52–57. — URL: http://media-publisher.ru/wp-content/uploads/Sinhroinfo-2-2024.pdf (дата обращения: 15.06.26).

9. Кричевский М.Л. Большие языковые модели при решении педагогических задач / М.Л. Кричевский // Образовательные ресурсы и технологии. — 2025. — 2(51). — С. 102–111. — URL: https://vestnik-muiv.ru/article/bolshie-yazykovye-modeli-pri-reshenii-pedagogicheskikh-zadach/ (дата обращения: 15.06.26).

10. Назаров Д.М. Применение больших языковых моделей в образовательном процессе / Д.М. Назаров, С.В. Бегичева // Бизнес. Образование. Право. — 2024. — 3(68). — С. 430–436. — URL: https://vestnik.volbi.ru/upload/numbers/368/article-368-4150.pdf (дата обращения: 15.06.26). — DOI: 10.25683/VOLBI.2024.68.1057

11. Lee J. The life cycle of large language models in education: A framework for understanding sources of bias / J. Lee, Y. Hicke, R. Yu et al. // British Journal of Educational Technology. — 2024. — 55(5). — P. 1982–2002. — URL: https://bera-journals.onlinelibrary.wiley.com/doi/10.1111/bjet.13505 (accessed: 15.06.26). — DOI: 10.1111/bjet.13505

12. Кудж С.А. Формирование комплексного подхода к развитию научно-образовательной инфраструктуры современного инженерного университета / С.А. Кудж, Н.Б. Голованова, Ю.Г. Графов // Russian Technological Journal. — 2026. — 14(2). — С. 134–145. — URL: https://www.rtj-mirea.ru/jour/article/view/1471 (дата обращения: 15.06.26). — DOI: 10.32362/2500-316X-2026-14-2-134-145

Список литературы на английском языке / References in English

1. Besfamil'naya E.M. Kontekstno-zavisimaya marshrutizaciya zaprosov k LLM v gibridny'x klasterax s kontrolem zatrat [Contextual-advanced routing of LLM query in hybrid clusters with cost control] / E.M. Besfamil'naya, A.V. Smirnov, O.M. Minaev // Soft Measurements and Computing. — 2025. — 97(12-2). — P. 96–103. — URL: https://s-lib.com/issues/smc_2025_12_2_a11/ (accessed: 15.06.26). — DOI: 10.36871/2618-9976.2025.12-2.011 [in Russian]

2. Shaykhulova A. Enhancing engineering decision-support with efficiently fine-tuned reasoning LLMs / A. Shaykhulova // International Research Journal. — 2025. — 12(162). — URL: https://research-journal.org/archive/12-162-2025-december/10.60797/IRJ.2025.162.125 (accessed: 15.06.26). — DOI: 10.60797/IRJ.2025.162.125

3. Vakushin A.A. Proektirovanie mnogokomponentny'x imitacionny'x modelej s pomoshh'yu bol'shoj yazy'kovoj modeli GPT-4 [Designing multicomponent simulation models using GPT-based LLM] / A.A. Vakushin, B.I. Klebanov // Engineering Bulletin of Don. — 2024. — 7(115). — P. 174–186. — URL: https://www.ivdon.ru/ru/magazine/archive/n7y2024/9340 (accessed: 15.06.26). [in Russian]

4. Rezczov S.M. Sravnitel'ny'j analiz yazy'kovy'x modelej v obrabotke nestrukturirovanny'x danny'x na primere Deepseek i GigaChat [Comparative analysis of language models in unstructured data processing: a case study of Deepseek and GigaChat] / S.M. Rezczov // Paradigm. — 2025. — 5. — P. 200–204. [in Russian]

5. Adilzhanova S.A. Primenenie LLM v kiberbezopasnosti: obzor prilozhenij i uyazvimostej LLM [The use of LLM in cybersecurity: overview of llm applications and vulnerabilities] / S.A. Adilzhanova, A.N. Қurasbek, M.O. Kenzhebaeva // Bulletin of the CAA. — 2025. — 38(3). — P. 118–136. — URL: https://vestnik.agakaz.kz/primenenie-llm-v-kiberbezopasnosti-obzor-prilozhenij-i-uyazvimostej-llm/ (accessed: 15.06.26). — DOI: 10.53364/24138614_2025_38_3_10 [in Russian]

6. Vasil'ev I.M. Vliyanie LLM-assistentov na snizhenie operacionnoj nagruzki v kontakt-centrax rossijskix bankov [The Impact of LLM Assistants on Reducing Operational Workload in Contact Centers of Russian Banks] / I.M. Vasil'ev // Financial Business. — 2025. — 12(270). — P. 131–136. — URL: https://fin-biz.ru/gallery/журнал%20Финансовый%20бизнес%202025-12.pdf (accessed: 15.06.26). [in Russian]

7. Budaeva D.A. Pragmalingvisticheskie aspekty' formulirovaniya promptov dlya ChatGPT v nauchnoj kommunikacii [Pragmalinguistic aspects of formulating prompts for chatgpt in scientific communication] / D.A. Budaeva, I.N. Zy'ryanova // Kazan Science. — 2024. — 7. — P. 220–224. — URL: https://www.kazanscience.ru/files/KH_7_2024.php (accessed: 15.06.26). [in Russian]

8. Fatxulin T.D. Analiz vliyaniya sostavlyaemy'x tekstovy'x zaprosov (promptov) na kachestvo izobrazhenij, generiruemy'x nejrosetevy'mi texnologiyami [An Analysis of the Impact of Text Prompts on the Quality of Images Generated by Neural Network Technologies] / T.D. Fatxulin, D.A. Smirnov, I.V. Razumov et al. // Signal synchronization, generation, and processing systems. — 2024. — 15(2). — P. 52–57. — URL: http://media-publisher.ru/wp-content/uploads/Sinhroinfo-2-2024.pdf (accessed: 15.06.26). [in Russian]

9. Krichevskij M.L. Bol'shie yazy'kovy'e modeli pri reshenii pedagogicheskix zadach [Large language models in solving pedagogical problems] / M.L. Krichevskij // Educational Resources and Technologies. — 2025. — 2(51). — P. 102–111. — URL: https://vestnik-muiv.ru/article/bolshie-yazykovye-modeli-pri-reshenii-pedagogicheskikh-zadach/ (accessed: 15.06.26). [in Russian]

10. Nazarov D.M. Primenenie bol'shix yazy'kovy'x modelej v obrazovatel'nom processe [Application of large language models in educational process] / D.M. Nazarov, S.V. Begicheva // Business. Education. Law. — 2024. — 3(68). — P. 430–436. — URL: https://vestnik.volbi.ru/upload/numbers/368/article-368-4150.pdf (accessed: 15.06.26). — DOI: 10.25683/VOLBI.2024.68.1057 [in Russian]

11. Lee J. The life cycle of large language models in education: A framework for understanding sources of bias / J. Lee, Y. Hicke, R. Yu et al. // British Journal of Educational Technology. — 2024. — 55(5). — P. 1982–2002. — URL: https://bera-journals.onlinelibrary.wiley.com/doi/10.1111/bjet.13505 (accessed: 15.06.26). — DOI: 10.1111/bjet.13505

12. Kudzh S.A. Formirovanie kompleksnogo podxoda k razvitiyu nauchno-obrazovatel'noj infrastruktury' sovremennogo inzhenernogo universiteta [Formation of a comprehensive approach to developing the scientific and educational infrastructure of a modern engineering university] / S.A. Kudzh, N.B. Golovanova, Yu.G. Grafov // Russian Technological Journal. — 2026. — 14(2). — P. 134–145. — URL: https://www.rtj-mirea.ru/jour/article/view/1471 (accessed: 15.06.26). — DOI: 10.32362/2500-316X-2026-14-2-134-145 [in Russian]