Научная статья
УДК 004.89; DOI: 10.61260/2218-130X-2026-2-171-178

ВЕРИФИКАЦИЯ КОЛИЧЕСТВЕННЫХ РЕЗУЛЬТАТОВ МЕРОПРИЯТИЙ НАЦИОНАЛЬНЫХ ПРОЕКТОВ НА ОСНОВЕ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ

✉Матвеев Алексей Сергеевич.
МИРЭА – Российский технологический университет, Москва, Россия
✉matveev@mirea.ru

Аннотация. Рассмотрена задача автоматизированной проверки исполнения мероприятий национальных проектов по их количественным результатам. Источником информации служит пакет отчётных документов, поступающий от исполнителя в контролирующий орган. Состав пакета заранее не известен, документы разнородны по типу и информационной природе. Простое суммирование числовых значений из таких документов даёт завышенный результат, поскольку одна транзакция, как правило, отражена в нескольких документах сразу. В статье предложена архитектура количественной проверки, использующая три последовательных пути: суммирование по серии однотипных первичных документов, сравнение с максимумом значений из агрегатных документов, дедуплицированная сумма с привлечением языковой модели. Пути расположены по убыванию степени доверия к результату. Архитектура реализована в составе программного пайплайна и апробирована на реальных пакетах. Научная новизна состоит в иерархической организации путей, при которой вероятностный метод применяется только тогда, когда другие способы не сработали.

Ключевые слова: автоматизированный аудит, национальные проекты, большие языковые модели, верификация количественных показателей, дедупликация транзакций, on-premise large language model

Для цитирования: Матвеев А.С. Верификация количественных результатов мероприятий национальных проектов на основе больших языковых моделей // Научно-аналитический журнал «Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России». 2026. № 2. С. 171–178. DOI: 10.61260/2218-130X-2026-2-171-178

Scientific article

VERIFICATION OF QUANTITATIVE RESULTS OF NATIONAL PROJECT ACTIVITIES BASED ON LARGE LANGUAGE MODELS

✉Matveev Alexey S.
MIREA – Russian technological university, Moscow, Russia
✉matveev@mirea.ru

Abstract. The paper considers the task of automated verification of Russian national project activities by their quantitative outcomes. The source of information is a package of reporting documents submitted by the executor to the oversight body. The composition of the package is not known in advance, and the documents are heterogeneous in type and informational nature. A simple summation of numerical values from such documents yields an inflated result, since one transaction is typically reflected in several documents at once. The paper proposes a verification architecture that uses three sequential paths: summation over a series of uniform primary documents, comparison with the maximum among aggregate documents, and a deduplicated sum obtained with a language model. The paths are ordered by decreasing degree of trust in the result. The architecture has been implemented within a software pipeline and tested on real document packages. The scientific novelty lies in the hierarchical organization of paths, where the probabilistic method is engaged only when the other methods have failed.

Keywords: automated audit, national projects, large language models, quantitative verification, transaction deduplication, on-premise large language model

For citation: Matveev A.S. Verification of quantitative results of national project activities based on large language models // Scientific and analytical journal «Vestnik Saint-Petersburg university of State fire service of EMERCOM of Russia». 2026. № 2. P. 171–178. DOI: 10.61260/2218-130X-2026-2-171-178

Введение

Ежегодная проверка исполнения мероприятий национальных проектов требует от контролирующего органа сопоставления плановых количественных показателей с фактически достигнутыми. План задан в паспорте мероприятия и содержит число, срок и тип агрегации. Факт же приходит к проверяющему в виде пакета отчётных документов от исполнителя. Состав пакета не регламентирован, и в нём могут встречаться накладные, акты, договоры, сводные отчёты, итоговые письма, фотоотчёты и многое другое. Объём такого пакета доходит до одного гигабайта в формате PDF.

Сегодня значительная часть этой работы выполняется вручную. Специалист открывает каждый документ, находит в нём числовые значения, оценивает их связь с проверяемым мероприятием и складывает соответствующие значения. Очевидно, что при тысячах мероприятий такой подход плохо масштабируется. Возникает потребность в автоматизации хотя бы первичного этапа – извлечения и сопоставления количественных показателей. Современные методы понимания макета документа [1, 2] и большие языковые модели [3, 4] позволяют такую задачу решать. Их применение в задачах аудита и государственной отчётности обсуждается в работах [5, 6].

Однако наивная автоматизация – извлечь все числа из документов и сложить – приводит к завышению результата. Дело в том, что одна и та же транзакция обычно отражена сразу в нескольких документах. Поставка из ста единиц оборудования упомянута в накладной, в акте приёма-передачи и затем в сводном отчёте за квартал. Простое сложение даст 300 единиц вместо 100. Дополнительная проблема состоит в том, что документы внутри пакета неравноценны по своей роли. Накладная фиксирует одну транзакцию, тогда как сводный отчёт уже содержит накопленный итог. Складывать их между собой нельзя. Задача дедупликации однотипных записей в разных источниках исследуется в рамках направлений record linkage и entity resolution [7, 8]; для документооборота нацпроектов её осложняет отсутствие единых идентификаторов транзакций.

Цель настоящей работы – предложить архитектуру количественной верификации, которая разводит документы пакета по их информационной природе и применяет к каждой группе свою стратегию подтверждения. Задачи работы: построить логику выбора стратегии, формализовать три типа агрегации (суммарный, нарастающий, убывающий) и апробировать архитектуру на реальных пакетах отчётности.

Методы исследования

Формальная постановка: мероприятие задаётся кортежем (n, P, d, A): n – название, P – плановое значение, d – крайний срок, A – тип агрегации. Возможные типы агрегации – суммарный, нарастающий и убывающий. Тип закреплён в паспорте мероприятия и в ходе проверки не меняется. На вход системе подаётся пара: мероприятие из паспорта и пакет документов от исполнителя. На выход требуется получить вердикт из трёх возможных – подтверждено, не подтверждено, требует ручной проверки, – а также текстовое обоснование к нему.

Перед собственно верификацией каждый документ пакета классифицируется по своей информационной природе. Выделено семь канонических категорий: акты приёма-передачи, финансовые транзакционные документы, накладные и инвентаризационные документы,

агрегатные документы (сводные отчёты, итоговые письма), бухгалтерские документы, прочие, неклассифицированные. Эта классификация выполняется отдельным шагом пайплайна и далее используется для маршрутизации документов по разным веткам верификации. Архитектурные вопросы клиент-серверной организации и единого хранилища промежуточных результатов рассмотрены в работах [9, 10].

Далее автор переходит к описанию основной логики – той, что применяется при суммарном типе агрегации. Логика организована как последовательность трёх путей. Первый путь самый простой и не требует языковой модели вовсе. Второй также детерминирован, но опирается на правильную классификацию документов на предыдущем шаге. Третий путь привлекает языковую модель и применяется только тогда, когда первые два не сработали. Такой порядок отражает разную степень доверия к результату: чем ниже путь в иерархии, тем больше в нём вероятностной составляющей.

Путь 1 – серия первичных документов одного типа. Если в пакете есть серия однотипных первичных документов, например, набор накладных одной серии нумерации, то их значения складываются и сравниваются с планом P. При сумме не менее P принимается вердикт «подтверждено». Серия однотипных документов одной нумерации не может описывать одну и ту же транзакцию по самой природе организационного документооборота, поэтому никакой дополнительной дедупликации тут не требуется.

Путь 2 – агрегатный максимум. Если первый путь не сработал, берутся документы из категории «агрегатные» – сводные отчёты и итоговые письма. Из их значений отбирается максимальное. При этом максимуме не менее P принимается вердикт «подтверждено». Агрегатные значения нельзя суммировать ни между собой, ни с первичными: каждое из них уже содержит накопленный итог, и сложение приведёт к двойному учёту.

Путь 3 – дедуплицированная сумма от языковой модели. Если первые два пути не дали подтверждения, привлекается языковая модель. Её работа разбита на два последовательных обращения. В первом обращении модели передаются документы пакета и числовые значения из них; задача – сгруппировать документы в уникальные транзакции, объединив дубликаты (то есть записи в разных документах, относящиеся к одному и тому же событию). Во втором обращении формируется текстовое обоснование принятого вердикта. Если дедуплицированная сумма достигает или превышает P, вердикт – «требует ручной проверки», а не «подтверждено». Это сознательное решение: группировка, выполненная моделью, не имеет формальных гарантий корректности, и финальное слово остаётся за человеком. Способы снижения вариативности языковых моделей рассмотрены в работах [11, 12], а вопросы эффективного инференса – в [13]. В авторской работе [14] показано, что многократный вызов одной модели с голосованием большинства повышает воспроизводимость результата по сравнению с одиночным вызовом.

Если ни один из трёх путей не дал положительного результата, вердикт – «не подтверждено». Описанная логика относится к суммарному типу агрегации. Для нарастающего типа задача проще: код собирает из всех документов пары (дата, накопленное значение), упорядочивает их по времени и проверяет, достигнут ли план к нужному сроку. Для убывающего типа логика зеркальная – ищется минимум.

Кроме трёх основных путей, в системе работает предварительный санитарный фильтр. Он распределяет документы по пяти бакетам: кандидаты на подсчёт суммы, агрегатные документы, бухгалтерские (в подсчёте количественного результата не участвуют), документы с подозрительными значениями (выбросы) и нерелевантные. Фильтр работает на детерминированном коде и языковую модель не задействует.

Реализация выполнена на Java. Инференс языковой модели обеспечен фреймворком vLLM с continuous batching [13]. В качестве модели используется Gemma 4 26B A4B (instruction-tuned) – MoE-модель с общим числом параметров порядка 25 млрд, из которых при каждом проходе активны примерно 3,8 млрд. Разреженная активация позволяет получить пропускную способность, близкую к плотной модели меньшего размера, при качестве, ожидаемом от модели с существенно большим общим числом параметров. Модель

развёрнута на одном графическом ускорителе NVIDIA H100 в квантизации FP8 (как для весов, так и для KV-кэша). Суммарная пропускная способность инференса с учётом входных и выходных токенов составляет порядка 1 400 токенов в секунду. Промежуточные результаты всех шагов пайплайна пишутся в PostgreSQL. Весь контур работает локально, без передачи данных внешним сервисам, что соответствует требованиям к работе с ведомственной информацией.

Стоит отдельно выделить ещё одно решение, заложенное в архитектуру. Вердикт – то есть выбор одного из трёх значений – определяется кодом, а не языковой моделью. Модель используется только для написания текстового обоснования. Такое разделение ролей делает результат воспроизводимым: один и тот же пакет документов с одним и тем же планом всегда даст один и тот же вердикт. Кроме того, логика принятия решения легко проверяется по коду, а не по диалогу с моделью.

Результаты исследования и их обсуждение

Архитектура реализована в составе пайплайна и апробирована на реальных пакетах отчётных документов. По итогам предшествующих шагов в базу данных загружено 2 370 валидных результатов извлечения числовых значений. Это итог обработки 2 910 исходных строк, из которых 452 (15,5 %) были отсеяны по лимиту длины OCR-представления в 120 000 символов, а 88 (3,0 %) – по причине синтаксических ошибок в извлечённых данных. Поскольку расчёт количественных метрик качества (accuracy, precision, recall) на данном этапе работы не проводился, в виде апробации приводится качественный разбор поведения архитектуры на серии обезличенных кейсов. Кейсы подобраны так, чтобы покрыть срабатывание каждого из трёх путей, нарастающий режим и граничный случай с отрицательным вердиктом.

В первом рассмотренном пакете сработал Путь 1. Мероприятие – оснащение комплектами образовательного оборудования. План – 120 комплектов к 31 декабря 2025 г., тип агрегации – суммарный. В пакете лежат восемь накладных одной серии нумерации (по 15 комплектов каждая), сводный отчёт за квартал и фотоотчёт. На шаге классификации накладные отнесены к первичным инвентаризационным, сводный отчёт – к агрегатным, фотоотчёт – к прочим. Шаг верификации сразу применил Путь 1: серия восьми накладных даёт $8 \times 15 = 120$ комплектов, что равно плану. Вердикт – «подтверждено». Остальные пути не запускались, и значение из сводного отчёта в подсчёте не участвовало – что и требовалось для предотвращения двойного учёта.

Во втором пакете результат был подтверждён по агрегатному документу. Мероприятие – проведение мероприятий регионального уровня. План – 50 мероприятий, тип агрегации суммарный. В пакете итоговое письмо губернатора региона с фразой о проведённых 52 мероприятиях, три отдельных протокола конкретных мероприятий и фотоотчёт. Путь 1 здесь не сработал: три протокола слишком разнотипны и не образуют единой серии, а их арифметическая сумма равна трём. Сработал Путь 2: максимум среди агрегатных документов равен 52, что не меньше 50. Вердикт – «подтверждено». Если бы значения из протоколов были прибавлены к значению из итогового письма, получилось бы 55 – неверный ответ. Корректный ответ – 52, и архитектура его даёт.

Третий кейс – единственный, в котором срабатывает Путь 3. Мероприятие – ввод в эксплуатацию инженерных коммуникаций. План – 25 километров к 30 июня 2025 г., тип агрегации суммарный. В пакете акт ввода первого этапа на 12 км, договор подряда с упоминанием тех же 12 км, сводный отчёт с фразой «12 км по первому этапу и 14 км по второму», а также акт ввода второго этапа на 14 км. Путь 1 неприменим – акты относятся к разным этапам и в одну серию не складываются. Путь 2 тоже неприменим – сводный отчёт даёт детализацию по этапам, а не один итог, который можно сопоставить с планом. Подключается Путь 3. Языковая модель в первом вызове группирует документы в две уникальные транзакции – 12 км первого этапа (подтверждается тремя документами) и 14 км

второго этапа (подтверждается двумя). Дедуплицированная сумма – 12 + 14 = 26 км, что не меньше плана. Вердикт – «требует ручной проверки», потому что группировку выполнила модель, и окончательное слово остаётся за специалистом-аудитором.

Четвёртый пакет иллюстрирует нарастающий режим. Мероприятие – охват программой повышения квалификации специалистов накопительным итогом. План – 5 000 специалистов к 31 декабря 2025 г. В пакете четыре ежеквартальных отчёта со значениями 1 400, 2 900, 4 200, 5 100. Код извлёк пары (дата, значение), упорядочил их и нашёл пик 5 100, достигнутый к нужному сроку. Поскольку 5 100 не меньше 5 000, вердикт – «подтверждено».

Пятый пакет показывает ситуацию с отрицательным вердиктом. Мероприятие – поставка единиц технологического оборудования. План – 80 единиц, тип агрегации суммарный. В пакете три накладные на 15, 15 и 10 единиц, агрегатных документов нет. Путь 1: сумма по накладным 40, что меньше 80. Путь 2 неприменим – нет агрегатных документов. Путь 3: дедуплицированная сумма по тем же документам также 40, что меньше 80. Вердикт по всем путям одинаков – «не подтверждено». Кейс важен тем, что показывает, как архитектура ведёт себя при объективной нехватке документов: система не экстраполирует отсутствующие данные и возвращает согласованный отрицательный ответ.

Логика проверки в суммарном режиме сведена в таблице.

Таблица

Иерархия путей верификации в суммарном режиме

Путь	Условие применения	Решающее правило	Вердикт
1	Есть серия однотипных первичных документов	Сумма серии ≥ P	Подтверждено
2	Путь 1 не сработал; есть агрегатные документы	Максимум значений агрегатных документов ≥ P	Подтверждено
3	Пути 1 и 2 не сработали	Дедуплицированная сумма (LLM) ≥ P	Требует ручной проверки
–	Ни один из путей не сработал	–	Не подтверждено

Далее автор разбирает общие свойства предложенной архитектуры. Иерархия путей устроена так, что приоритет имеет наиболее простой способ – серия однотипных первичных документов, для которой никакая дедупликация вообще не нужна. Языковая модель в этой архитектуре стоит в самом конце очереди, и её результат специально ограничен вердиктом «требует ручной проверки». Это сделано осознанно: дедупликация моделью не имеет формальных гарантий, и автоматически подтверждать что-либо на её основе было бы неправильно.

Предложенный подход сравнивается с прямолинейным – суммированием всех чисел из всех документов пакета и сравнением итога с планом. Прямолинейный подход даёт неверный результат сразу в трёх ситуациях. Первая: в пакете есть агрегатный документ с итоговым значением, и он суммируется с первичными – получается завышение. Вторая: одна транзакция отражена в нескольких документах, и каждый из них считается отдельно – снова завышение. Третья: пакет смешанный, то есть в нём и первичные, и агрегатные источники одновременно – завышение по обеим причинам сразу. Предложенная архитектура каждую из этих ситуаций разводит.

Главное практическое ограничение системы на сегодня – лимит длины OCR-представления документа в 120 000 символов. Документы, превышающие этот лимит, отсекаются и в дальнейшей обработке не участвуют. В среднем по прогонам отсекается около 15,5 % документов, но на отдельных подвыборках доля доходит до 37 %. Особенно страдают пакеты с длинными документами – региональными программами, годовыми отчётами, многостраничными приказами. Здесь возможны несколько направлений работы: разбиение длинных документов на части с последующим объединением результатов, переход на модель с расширенным контекстным окном, использование vision-language модели в качестве резервного канала для разбора длинных документов.

Второе направление развития – формальная количественная оценка качества. В рамках настоящей работы такая оценка не приводится: она потребует независимой разметки корпуса несколькими аудиторами с последующим расчётом согласия между ними. Это отдельная исследовательская задача, выходящая за рамки текущего этапа.

Заключение

В работе предложена архитектура автоматизированной верификации количественных результатов мероприятий национальных проектов. В её основе – иерархическая последовательность из трёх путей подтверждения: суммирование по серии однотипных первичных документов, сравнение с максимумом среди агрегатных документов и дедуплицированная сумма, восстановленная языковой моделью. Архитектура обрабатывает разнородные пакеты отчётности, в которых одна транзакция может быть отражена в нескольких документах одновременно, а часть документов является агрегатными по своей природе.

Научная новизна работы состоит в самой иерархии путей. Приоритет отдан тем способам подтверждения, которые не привлекают вероятностную модель и опираются на свойства документооборота, известные заранее. Вероятностный путь применяется только в случае, когда первые два не сработали, и его результатом служит более консервативный вердикт – «требует ручной проверки». Такая организация принципиально отличается от прямолинейного сложения всех чисел из всех документов и устраняет систематическое завышение, которое возникает из-за дублирования транзакций и смешения первичных и агрегатных источников.

Архитектура реализована в составе пайплайна на локально развёрнутой языковой модели и апробирована на реальных пакетах отчётных документов. Качественный разбор обезличенных кейсов показал согласованное поведение всех путей и корректность принимаемых вердиктов. Полученные результаты могут быть использованы в задачах ведомственного аудита исполнения программ и проектов, имеющих количественные целевые показатели. Дальнейшее развитие связано с обработкой длинных OCR-представлений, созданием размеченного корпуса для количественной оценки качества и переносом иерархического принципа на другие задачи автоматизированной проверки отчётности.

Список источников

1. LayoutLM: Pre-training of Text and Layout for Document Image Understanding / Y. Xu [et al.] // Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020. P. 1192–1200.

2 OCR-Free Document Understanding Transformer / G. Kim [et al.] // Proceedings of the European Conference on Computer Vision. 2022. P. 498–517.

3. Language Models are Few-Shot Learners / T.B. Brown [et al.] // Advances in Neural Information Processing Systems. 2020. Vol. 33. P. 1877–1901.

4. Llama 2: Open Foundation and Fine-Tuned Chat Models / H. Touvron [et al.] // arXiv preprint arXiv:2307.09288. 2023.

5. Дудихин В.В., Кондрашов П.Е. Методология использования больших языковых моделей для решения задач государственного и муниципального управления по интеллектуальному реферированию и автоматическому формированию текстового контента // Государственное управление. Электронный вестник. 2024. № 105. С. 5–18.

6. Натарова Е.В. Цифровая трансформация в системе государственного аудита // Научный вестник: финансы, банки, инвестиции. 2024. № 4 (69). С. 76–88.

7. Christen P. Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution and Duplicate Detection. Heidelberg: Springer, 2012. 270 p. DOI: 10.1007/978-3-642-31164-2

8. Köpcke H., Rahm E. Frameworks for entity matching: A comparison // Data & Knowledge Engineering. 2010. Vol. 69. № 2. P. 197–210. DOI: 10.1016/j.datak.2009.10.003

9. Лиманова Н.И., Селезнев И.А. Анализ эффективности клиент-серверной архитектуры // Бюллетень науки и практики. 2022. Т. 8. № 7. С. 392–396. DOI: 10.33619/2414-2948/80/37

10. vLLM. Official documentation. URL: https://docs.vllm.ai/en/latest/ (дата обращения: 21.05.2026).

11. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models / J. Wei [et al.] // Advances in Neural Information Processing Systems. 2022. Vol. 35. P. 24824–24837.

12. Self-Consistency Improves Chain of Thought Reasoning in Language Models / X. Wang [et al.] // Proceedings of the International Conference on Learning Representations. 2023.

13. Efficient Memory Management for Large Language Model Serving with PagedAttention / W. Kwon [et al.] // Proceedings of the 29th Symposium on Operating Systems Principles. 2023. P. 611–626. DOI: 10.1145/3600006.3613165

14. Задорнов А.А., Никонов С.С., Федотов М.В. Влияние единого информационного пространства на автоматизацию процесса получения и обработки информации // Вестник экономических и социологических исследований. 2024. № 1. С. 21–27.

References

1. LayoutLM: Pre-training of Text and Layout for Document Image Understanding / Y. Xu [et al.] // Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020. P. 1192–1200.

2 OCR-Free Document Understanding Transformer / G. Kim [et al.] // Proceedings of the European Conference on Computer Vision. 2022. P. 498–517.

3. Language Models are Few-Shot Learners / T.B. Brown [et al.] // Advances in Neural Information Processing Systems. 2020. Vol. 33. P. 1877–1901.

4. Llama 2: Open Foundation and Fine-Tuned Chat Models / H. Touvron [et al.] // arXiv preprint arXiv:2307.09288. 2023.

5. Dudihin V.V., Kondrashov P.E. Metodologiya ispol'zovaniya bol'shih yazykovyh modelej dlya resheniya zadach gosudarstvennogo i municipal'nogo upravleniya po intellektual'nomu referirovaniyu i avtomaticheskomu formirovaniyu tekstovogo kontenta // Gosudarstvennoe upravlenie. Elektronnyj vestnik. 2024. № 105. S. 5–18.

6. Natarova E.V. Cifrovaya transformaciya v sisteme gosudarstvennogo audita // Nauchnyj vestnik: finansy, banki, investicii. 2024. № 4 (69). S. 76–88.

7. Christen P. Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution and Duplicate Detection. Heidelberg: Springer, 2012. 270 p. DOI: 10.1007/978-3-642-31164-2

8. Köpcke H., Rahm E. Frameworks for entity matching: A comparison // Data & Knowledge Engineering. 2010. Vol. 69. № 2. P. 197–210. DOI: 10.1016/j.datak.2009.10.003

9. Limanova N.I., Seleznev I.A. Analiz effektivnosti klient-servernoj arhitektury // Byulleten' nauki i praktiki. 2022. T. 8. № 7. S. 392–396. DOI: 10.33619/2414-2948/80/37

10. vLLM. Official documentation. URL: https://docs.vllm.ai/en/latest/ (data obrashcheniya: 21.05.2026).

11. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models / J. Wei [et al.] // Advances in Neural Information Processing Systems. 2022. Vol. 35. P. 24824–24837.

12. Self-Consistency Improves Chain of Thought Reasoning in Language Models / X. Wang [et al.] // Proceedings of the International Conference on Learning Representations. 2023.

13. Efficient Memory Management for Large Language Model Serving with PagedAttention / W. Kwon [et al.] // Proceedings of the 29th Symposium on Operating Systems Principles. 2023. P. 611–626. DOI: 10.1145/3600006.3613165

14. Zadornov A.A., Nikonov S.S., Fedotov M.V. Vliyanie edinogo informacionnogo prostranstva na avtomatizaciyu processa polucheniya i obrabotki informacii // Vestnik ekonomicheskih i sociologicheskih issledovanij. 2024. № 1. S. 21–27.

Информация о статье:
Статья поступила в редакцию: 29.05.2026; одобрена после рецензирования: 26.06.2026;
принята к публикации: 29.06.2026

Information about the article:
The article was submitted to the editorial office: 29.05.2026; approved after review: 26.06.2026;
accepted for publication: 29.06.2026

Информация об авторах:
Матвеев Алексей Сергеевич, младший научный сотрудник Научно-исследовательской лаборатории НОЦ ИИ Института ИИ МИРЭА – Российского технологического университета (119454, Москва, пр. Вернадского, д. 78), e-mail: matveev_as@mirea.ru, https://orcid.org/0009-0002-6013-0950

Information about authors:
Matveev Alexey S., junior researcher at the research laboratory of the scientific and educational center, Institute of artificial intelligence of the MIREA – Russian university of technology (119454, Moscow, Vernadsky ave., 78), e-mail: matveev_as@mirea.ru, https://orcid.org/0009-0002-6013-0950