УДК 004.054 **DOI** 10.15622/ia.24.6.7

С.Н. Карпович, А.В. Смирнов, Н.Н. Тесля

МЕТОДОЛОГИЯ ПОСТРОЕНИЯ ЭТАЛОННОГО ТЕСТА ДЛЯ ОЦЕНКИ РАБОТЫ LLM С ЧИСЛИТЕЛЬНЫМИ

Карпович С.Н., Смирнов А.В., Тесля Н.Н. **Методология построения эталонного теста для оценки работы LLM с числительными.**

Аннотация. В статье представлена методология разработки эталонного теста для оценки навыков числового мышления в больших языковых моделях (Large Language Models, LLM). Под числовым мышлением в контексте LLM понимается способность модели корректно интерпретировать, обрабатывать и использовать числовую информацию в тексте – включая понимание значений чисел, их соотношений, выполнение арифметических операций, а также корректную генерацию числительных в ответах. Предложенная методология основана на декомпозиции прикладных задач и позволяет оценивать отдельные аспекты числового мышления на примере задач с числительными. Особое внимание уделяется способу представления чисел в текстовых инструкциях к LLM, поскольку это напрямую влияет на качество итогового ответа. Необходимость точной оценки числового мышления LLM обусловлена тем, что эта способность критически важна для широкого спектра прикладных задач работы с текстами, в том числе для автоматизированного составления кратких изложений, генерации аналитических отчётов, извлечения и интерпретации количественных данных, а также для диалоговых систем, работающих с финансовой, научной или технической информацией. На основе анализа современных подходов к оценке LLM сформулированы основные принципы построения эталонных тестов с упором на универсальность и применимость в реальных сценариях. В соответствии с предложенной методологией разработан эталонный тест MUE (Math Understanding Evaluation), включающий пять наборов тестовых заданий, каждый из которых предназначен для оценки отдельного аспекта числового мышления LLM. Проведена сравнительная оценка качества популярных LLM, определены лидеры, а также выявлены сильные и слабые стороны их числового мышления. Полученные результаты могут использоваться разработчиками LLM для улучшения архитектур и стратегий обучения, а также конечными пользователями и интеграторами для выбора оптимальной модели в прикладных проектах.

Ключевые слова: методология, большие языковые модели (LLM), эталонный тест LLM, обработка естественного языка (NLP), числительные.

1. Введение. В настоящее время большие языковые модели (LLM), такие как GPT [1], Gemini [2], Mistral [3], LLaMA [4], GigaChat [5], YandexGPT [6], Qwen, DeepSeek [7, 8] успешно применяются при создании цифровых продуктов в образовании, здравоохранении, юриспруденции, информационных технологиях, маркетинге, научных исследованиях, государственном секторе. Под большими языковыми моделями (Large Language Models, LLM) понимаются такие языковые модели, масштаб которых и архитектура позволяют им демонстрировать эмерджентные свойства – функциональные возможности, не наблюдаемые в меньших по размеру

моделях. Как правило, LLM содержат от сотен миллионов до десятков миллиардов параметров и обучены на обширных корпусах текстов, что обеспечивает способность решать широкий спектр задач без специализированного дообучения. Основные прикладные задачи, решаемые с помощью LLM, включают генерацию текста, перевод, редактирование и улучшение текста, анализ и классификация, ответы на вопросы и диалоги, обработка естественного языка, извлечение именованных сущностей.

Для таких продуктов и задач важно обеспечить высокое качество и стабильность ответов. LLM галлюцинируют (генерируют правдоподобную, но фактически неверную информацию), генерируют ошибки, дают некорректные или неожиданные ответы. Это требует внедрения дополнительных механизмов валидации и контроля качества создаваемого контента.

Одним из способов повышения надежности является оптимизация формулировок текстовых инструкций – промптов. Промпт-инженеры экспериментально подбирают конфигурации, обеспечивающие стабильные и точные ответы. Однако этот процесс осложняется большим количеством языков, наличием синонимов и различных формулировок для одной и той же задачи.

Особую сложность для LLM представляют задачи, связанные с обработкой и интерпретацией числовых данных [8 – 15]. Например, может быть поставлена задача сгенерировать текст, ограниченный заданным количеством символов, или выполнить арифметические операции, встроенные в текстовый запрос. Несмотря на то, что современные LLM демонстрируют впечатляющие результаты в обработке текста, их способность к точным математическим расчетам и интерпретации чисел остается недостаточно изученной и, зачастую, ограниченной.

Основной целью исследования является разработка методологии построения эталонного теста (бенчмарка) для систематического тестирования LLM в части работы с числительными. Методология должна определить принципы оценки способности моделей интерпретировать, обрабатывать и генерировать числовую информацию в различных форматах и задачах. В рамках исследования, представленного в данной статье, были проанализированы существующие подходы и эталонные тесты, выявлена их применимость к анализу числительных; разработана методология построения эталонного теста с набором заданий, критериями и процедурой тестирования; проведено сравнительное тестирование популярных LLM (GPT, Gemini, Llama, Claude и др.); проанализированы результаты,

выделены сильные и слабые стороны моделей, классифицированы наиболее частые ошибки и даны практические рекомендации по формулировке запросов с числительными.

Материал статьи структурирован следующим образом. Во втором разделе обосновывается важность разработки методологии для оценки работы больших языковых моделей (LLM) с акцентом на обработку числительных. В третьем разделе представлен обзор существующих подходов к тестированию LLM, который включает два направления: эталонные тесты для оценки математических способностей (3.1) и эталонные тесты для анализа общего понимания, знаний и рассуждений (3.2). Четвертый раздел содержит формальную постановку задачи (4.1), раскрывает предложенную методологию построения эталонного теста, включая основные принципы (4.2) и детальное описание этапов разработки (4.3), ориентированных на оценку точности, устойчивости и интерпретируемости моделей при работе с числительными. Пятый раздел демонстрирует практическое применение методологии: описывается структура разработанного набора данных (5.1), и представлен эталонный тест MUE (5.2), предназначенный для анализа работы LLM с числовыми данными. Шестой раздел содержит результаты экспериментов, подтверждающих эффективность предложенного подхода. В заключении сформулированы выводы о научной и практической ценности методологии, а также обозначены перспективы дальнейших исследований.

2. Актуальность разработки методологии построения эталонного теста для оценки работы LLM с числительными. При подготовке входных данных к обработке каждая LLM использует два последовательных шага: токенизацию и векторизацию. Токенизация – это процесс разбиения текста на отдельные части (токены), которые служат минимальными единицами анализа для последующей обработки в задачах NLP. Эти токены могут быть словами, подсловами, символами, числами, знаками препинания или даже фразами, в зависимости от выбранного метода. Токенизация является ключевым этапом в обработке текста, который определяет, как модель будет интерпретировать данные. Современные методы токенизации включают алгоритм Byte Pair Encoding (BPE) [16], который используется в GPT и RoBERTa, алгоритм WordPiece [17], применяемый в моделях BERT, и алгоритм SentencePiece [18], используемый в T5 и ALBERT. Векторизация – это процесс преобразования токенизированных текстовых данных в числовые векторы (эмбеддинги), которые могут быть обработаны алгоритмами

машинного обучения и глубокого обучения, с сохранением семантических и синтаксических свойств текста.

Разные подходы к токенизации и векторизации в LLM приводят к различным векторным представлениям одного и того же текста. Это актуально для числительных. В то время как для человека различные формы записи числа (текстовые, на разных языках – «сто двадцать три», «one hundred twenty-three», «cent vingt-trois»; цифровые – «CXXIII», «123») имеют одинаковое значение, для модели они могут представлять существенно разные последовательности токенов. Это приводит к соответствующим различиям в векторных представлениях, степень семантической близости которых определяется, например, косинусным расстоянием. Такие различия могут влиять на способность LLM корректно интерпретировать числовые данные и выполнять соответствующие задачи.

В качестве примера рассмотрим косинусные расстояния между векторными представлениями числа «123», полученными с помощью API большой языковой модели GigaChat (таблица 1), рассчитанное по формуле 1, где V_1 и V_2 – векторные представления токенов.

$$\overline{V}_1 = \left(v_{1x}, v_{1y}\right); \overline{V}_2 = \left(v_{2x}, v_{2y}\right),$$

$$Sim_{\cos}(\overline{V}_1, \overline{V}_2) = \frac{\left(\overline{V}_1, \overline{V}_2\right)}{\left|\overline{V}_1\right| * \left|\overline{V}_2\right|} = \frac{v_{1x} * v_{2x} + v_{1y} * v_{2y}}{\sqrt{v_{1x}^2 + v_{1y}^2} * \sqrt{v_{2x}^2 + v_{2y}^2}} \cdot \quad (1)$$

Таблица 1. Пример косинусных расстояний между векторными представлениями числа 123 в различных форматах записи

	Русский	Английский	Французский	Китайский	Цифры
Русский	1,00	0,95	0,93	0,91	0,87
Английский		1,00	0,95	0,90	0,89
Французский			1,00	0,91	0,87
Китайский				1,00	0,87
Цифры					1,00

Данные таблицы 1 свидетельствуют о влиянии формы представления числительного на его внутренне представление и интерпретацию в LLM. При схожем семантическом содержании текстовых форм на русском, английском и французском языках (косинусное расстояние близко к 1), представление на китайском уже заметно отличается (0,91). Наибольшее расхождение наблюдается между текстовыми и числовыми формами (например, 0,87 для

русского), что может вызывать различия в реакциях LLM на инструкции.

При разработке методологии построения эталонного теста необходимо определить принципы отбора задач для оценки числового мышления на пересечении математики и языкового понимания. Методология должна быть ориентирована на специалистов, использующих LLM в прикладных задачах, охватывающих аспекты числового мышления (восприятие чисел – идентификация и извлечение числовых значений из текста, выполнение базовых арифметических действий, сравнение числовых значений, логические рассуждения – поиск закономерностей в последовательностях чисел) и языкового понимания (контекстуальное и лексическое понимание, причинно-следственные связи, анализ текста, ответы на вопросы на основе общих знаний).

3. Обзор существующих эталонных тестов для анализа больших языковых моделей. Эталонные тесты LLM представляют собой наборы задач и методик оценки для сравнения больших языковых моделей по заранее установленным метрикам. Они могут включать тесты на понимание и генерацию текста, перевод, краткое изложение, ответы на вопросы и другие задачи, требующие обработки естественного языка. Наиболее известные международные коллекции эталонных тестов представлены на платформе Hugging Face [19], где собраны такие популярные тестовые наборы как Instruction-Following Evaluation (IFEval) [20], Big Bench Hard (BBH) [21], Mathematics Aptitude Test of Heuristics (MATH) [8], Graduate-Level Google-Proof Q&A (GPQA) [22], Multistep Soft Reasoning (MuSR) [23] и Massive Multitask Language Understanding–Professional (MMLU-Pro) [24]. Создаются специализированные сравнительные таблицы, объединяющие результаты оценки моделей по нескольким эталонным тестам, такие как, например, MERA, для оценки русскоязычных моделей [25]. В целом, основной задачей эталонных тестов является стандартизация процесса оценки LLM, предоставляя единый подход к оценке посредством единых задач и метрик для сравнения и отслеживания прогресса моделей.

Далее рассмотрим основные эталонные тесты, используемые для оценки языковых моделей.

3.1. Эталонные тесты для оценки математических способностей. Эталонный тест HARDMATH [8] сфокусирован на задачах прикладной математики. Он включает задания на асимптотический анализ, работу с полиномами (нормализация,

нахождение корней), решение нелинейных дифференциальных уравнений, вычисление интегралов и задачи со сложным контекстом.

Набор задач DeepMind [9] разработан для оценки математического рассуждения LLM и состоит из последовательных текстовых вопросов, охватывающих различные области математики: от арифметики до анализа.

Набор данных MATH [10] содержит 12 500 сложных математических задач уровня соревнований с пошаговыми решениями, классифицированных по пяти уровням сложности. Он предназначен для углублённой оценки способности моделей в математическом рассуждении.

Мультимодальный эталонный тест MATHVISTA [11], предназначен для оценки способностей математического мышления ИИ в визуальных контекстах. Он содержит 6 141 пример, объединяющий задачи из различных областей математики и визуальные данные (из 28 существующих и трех новых наборов данных: IQTest, FunctionQA, PaperQA).

Эталонный тест FrontierMath [12] состоит из оригинальных и ранее не публиковавшихся математических задач исследовательского уровня (теория чисел, алгебраическая геометрия и др.), требующих творческого подхода.

NumericBench [13] оценивает базовые навыки числового мышления, необходимые для реальных приложений, таких как анализ финансовых данных или прогнозирование погоды, включающие в себя распознавание чисел, арифметические операции, контекстное извлечение, сравнение, суммирование, логическое рассуждение.

Эталонный тест NUPA [14] охватывает четыре основных представления чисел (целые, вещественные числа, дроби и научная нотация) и 17 различных типов задач, связанных с их обработкой.

Эталонный тест Numberland [15] оценивает способности к числовому мышлению, абстрактному пониманию чисел и их взаимосвязей («числовой интуиции»), которое позволяет решать математические задачи с ограниченными вычислительными ресурсами.

3.2. Эталонные тесты для оценки общего понимания, знаний и рассуждений. Набор данных MuSR (Multi-Step Reasoning) [23] разработан для оценки способности LLM к многошаговым «мягким» рассуждениям на основе текстовых нарративов (детективные истории, задачи на размещение объектов и другие).

Эталонный тест MMLU (Massive Multitask Language Understanding) [24], представленный на конференции ICLR 2021, оценивает многозадачное понимание и знания LLM на 57

разнообразных темах разного уровня сложности (от элементарной математики и истории до права и информатики). Оценка проводится в режимах «Zero-Shot» (промпты без примеров результата работы LLM) и «Few-Shot» (промпты с одним и более примерами результата работы LLM).

Для комплексной оценки LLM на русском языке предложен открытый эталонный тест MERA (Multimodal Evaluation of Russian-language Architectures) [25]. Он включает в себя 21 задачу, являющиеся сложными для фундаментальных моделей: вопросы охватывают знания о мире, логику, причинно-следственные связи, этику ИИ и многое другое.

SuperGLUE [26] – это расширенный и усложненный эталонный тест, пришедший на смену GLUE, для оценки общего понимания языка системами NLP. Включает в себя 8 задач (BoolQ, CommitmentBank, COPA, MultiRC, ReCoRD, RTE, WiC, WSC), требующих глубокого языкового понимания и способностей к рассуждению для решения разнообразных задач.

RussianSuperGLUE [27] является адаптированным для русского языка аналогом эталонного тест SuperGLUE. Включает 9 задач (LiDiRus, RUSSE, PARus, TERRa, RCB, RWSD, MuSeRC, RuCoS, DaNetQA), каждая из которых оценивает различные аспекты понимания русского языка, такие как разрешение языковых неоднозначностей, понимание причинно-следственных связей, распознавание текстового следствия и машинное чтение.

BIG-Bench Hard (BBH) [21] представляет собой поднабор из 23 особо сложных задач, выбранных из более широкого эталонного теста BIG-Bench. Эти задачи были отобраны, так как предыдущие LLM не могли превзойти на них средний человеческий результат, и они требуют продвинутых способностей к многошаговому рассуждению (включая такие виды, как алгоритмическое, арифметическое, на основе знаний о мире и понимания языка).

IFEval [20] – это эталонный тест, разработанный для объективной и автоматизированной оценки способности моделей LLM следовать инструкциям, заданным на естественном языке.

Эталонный тест GPQA (Graduate-Level Google-Proof Q&A) [22] состоит из 448 сложных вопросов с множественным выбором по биологии, физике и химии, созданных и проверенных экспертами. Вопросы спроектированы так, чтобы ответ на них было трудно найти с помощью поиска («Google-Proof»).

В таблице 2 приведён систематизированный перечень рассмотренных эталонных тестов с указанием их функциональной

направленности (математические задачи или оценка языкового понимания), а также характеристик: уровень сложности тестовых заданий, частота обновления набора данных, возможность расширения пользовательскими элементами. Знаком вопроса обозначено отсутствие информации о возможностях эталонного теста.

Таблица 2. Сравнение эталонных тестов по типам задач

Название эталонного теста	Оценка уровня сложности	Обновляемость набора заданий	Возможность расширения
Математические задачи			
HARDMATH [8]	Высокий	–	+
Набор задач DeepMind [9]	От элементарного до высокого	+	+
MATH [10]	Высокий	–	?
MATHVISTA [11]	Высокий	–	?
FrontierMath [12]	Исключительно высокий	–	–
NumericBench [13]	Высокий	–	?
NUPA [14]	От элементарного до высокого	–	?
Numberland [15]	От элементарного до высокого	–	–
Языковое понимание			
MuSR [23]	Высокий	+	+
MMLU [24]	Высокий	–	?
MERA [25]	Высокий	–	–
SuperGLUE [26]	Высокий	–	–
RussianSuperGLUE [27]	Высокий	–	–
BIG-Bench Hard [21]	Высокий	–	–
IFEval [20]	Средний	–	+
GPQA [22]	Высокий	–	–

Анализ таблицы 2 показывает, что существующие эталонные тесты охватывают широкий диапазон тестовых сценариев – от прикладных вычислительных и мультимодальных задач до комплексных проверок предметных знаний, логических рассуждений и многошагового вывода. При этом сохраняются области, в которых оценка возможностей LLM носит неполный или фрагментарный характер. Заметен недостаток эталонных тестов, предназначенных для комплексного тестирования числового мышления и языковой компетентности, имеющих ключевое значение в ряде прикладных задач. Установлено, что большинство рассмотренных наборов заданий имеют высокий уровень сложности, что целесообразно в

академической среде, однако не обеспечивает репрезентативной оценки производительности LLM в условиях простых прикладных задач. Кроме того, подавляющее число эталонных тестов не обновляются, что повышает вероятность предобученности моделей на исходных данных и не позволяет объективно оценить их способность к обобщению. Небольшое число эталонных тестов допускает добавление новых тестовых заданий, что существенно ограничивает их применимость в динамично изменяющихся условиях. Разработка эталонных тестов, обеспечивающих комплексную оценку нескольких аспектов работы LLM, включающих задания различной сложности, обладающих регулярной обновляемостью и возможностью интеграции пользовательских тестов, представляется перспективным направлением, способным устранить указанные недостатки и расширить функциональную применимость систем оценки.

4. Методология построения эталонного теста для оценки работы LLM с числительными. Несмотря на наличие множества подходов для оценки LLM (например, CheckList [28]), сохраняется потребность в инструментах, направленных на систематическое исследование способности моделей инвариантно воспринимать числовые данные в различных формах записи. Это имеет значение при проверке их эффективности в прикладных задачах, требующих точной обработки числовой информации. Разработка методологии для построения кастомизированных эталонных тестов представляет собой важную стратегическую задачу, позволяя каждому практику и исследователю самостоятельно проводить оценку LLM и выполнять проверку качества информационных систем.

4.1. Постановка задачи. Пусть N множество чисел. Для каждого числа $n \in N$ определим множество его текстовых и символьных представлений $R(n)$. Это множество включает:

– Цифровую запись: $r_{\text{dig}}(n)$, например, 123;

– Запись римскими цифрами: $r_{\text{rom}}(n)$, например, CXXII;

– Текстовую запись на различных языках.

Пусть $L = \{l_1, l_2, l_3, \ldots, l_k\}$ – множество естественных языков (например, l_1 – русский, l_2 – английский). Тогда $r_{\text{text},l}(n)$ – это текстовое представление числа n на языке $l \in L$. Для текстовой записи возможны два варианта, когда задание и числительные написаны на одном языке и на разных, например задание дано на русском языке, а другой язык использован только для числительного.

Таким образом множество всех представлений числа n можно определить как:

$$R(n) = \left\{ r_{\text{dig}}(n), r_{\text{rom}}(n) \right\} \bigcup \left\{ r_{\text{text},l}(n) \mid l \in L \right\}.$$

Пусть M – это большая языковая модель, которую рассматриваем как функцию, преобразующую множество входных текстовых промптов P в всех возможных ответов модели O:

$$M : P \to O.$$

Определим тестовое задание T как кортеж (упорядоченный набор) эталонных ответов $a_i^{\text{ref}} \in A$, которые могут быть числом, текстом или другой структурой, соответствующих промпту $p_i \in P$. При этом будем считать, что часть возможных ответов модели входит в множество эталонных ответов:

$$T_i = \left(p_i, a_i^{\text{ref}} \right), \exists a_i^{\text{ref}} : a_i^{\text{ref}} \in O, A \bigcap O \neq \varnothing.$$

Задача заключается в создании эталонного теста B, представляющего собой множество тестовых заданий T_i – кортежей, содержащих промпт $p_i \in P$ и эталонный ответ $a_i^{\text{ref}} \in A$, таких, чтобы они содержали все возможные представления чисел из $R(n)$:

$$B = \left\{ T_1, T_2, \ldots, T_m, \right\}.$$

Одно из основных требований к эталонному тесту заключается в том, что он должен содержать группы задач, построенные на одной и той же логической основе, но с использованием разных представлений чисел. В соответствии с этим, определим G_j как j-ю группу задач. Все задачи в этой группе имеют общую семантическую основу и единый эталонный ответ $a_i^{\text{ref}} \in A$, но используют разные вариации промпта. Промпты $p_{ij} \in P$ создаются путем подстановки различных представлений $r \in R(n)$ одних и тех же чисел n в базовый шаблон промпта ψ_j:

$$G_j = \left\{ \left(p_{i1}, a_j^{\text{ref}} \right), \left(p_{i2}, a_j^{\text{ref}} \right), \ldots, \left(p_{ik}, a_j^{\text{ref}} \right) \right\},$$

где $p_{ij} = \psi_j(r_1, r_2, \ldots)$ с различными $r_k \in R(n_k)$.

Для оценки результата введем функцию сравнения EM, которая определяет, является ли вывод модели o_i эквивалентным эталонному ответу a_i^{ref}:

$$EM\left(o_i, a_i^{\text{ref}}\right) \to \{0,1\},$$

где *1* означает правильный ответ, а *0* – неправильный. Функция EM устойчива к формату, например, $EM(cmo, 100) = 1$.

В данной постановке задачи обеспечивается оценка робастности, устойчивости к формату представления чисел – способность модели давать одинаково правильный результат на семантически эквивалентных задачах, независимо от формы записи чисел. Для этого в каждой группе задач G_j вычисляется доля правильных ответов. Идеально робастная модель будет иметь $Acc(M, G_j) = 1$ для всех *j*, где она знает правильный ответ.

4.2. Основные принципы методологии. По результатам анализа существующих наборов данных и эталонных тестов были сформулированы следующие принципы методологии построения эталонного теста для оценки работы LLM.

1) Ориентация на задачу. Полученный эталонный тест должен гарантировать, что критерии оценки и метрики непосредственно связаны с реальными целями применения модели.

2) Изоляция тестирования отдельных навыков модели. Многие задачи требуют комплексного подхода при их решении и могут быть разделены на отдельные подзадачи. Принцип изоляции отдельных навыков позволяет точно определить сильные и слабые стороны модели, избегая смешения эффектов при проверке сложных многокомпонентных задач. Для упрощения сложных задач до более простых, атомарных используется подход «Downward Evolution» [29].

3) Репрезентативность и валидность метрик в общей и частных задачах. Метрики, используемые в эталонном тесте, должны отражать как общую эффективность модели по задаче, так и адекватно оценивать каждый навык в отдельности. Валидность в данном принципе означает, что полученные оценки действительно соответствуют заявленным аспектам поведения модели.

4) Репрезентативность набора данных для тестирования. Набор тестовых примеров должен покрывать значимые сценарии и вариативность входных данных, чтобы результаты были применимы к реальным условиям.

5) Воспроизводимость результатов тестирования. Процедуры запуска эталонных тестов, настройки окружения и расчёта метрик документируются так, чтобы при повторном проведении (иными исследователями) получались идентичные выводы.

4.3. Процесс построения эталонного теста. Исходя из сформулированных принципов, процесс построения эталонного теста в рамках методологии разделен на этапы, представленные на рисунке 1. Рассмотрим подробнее методы и подходы, используемых на каждом этапе методологии построения эталонного теста для оценки работы LLM на примере задачи краткого изложения текста.

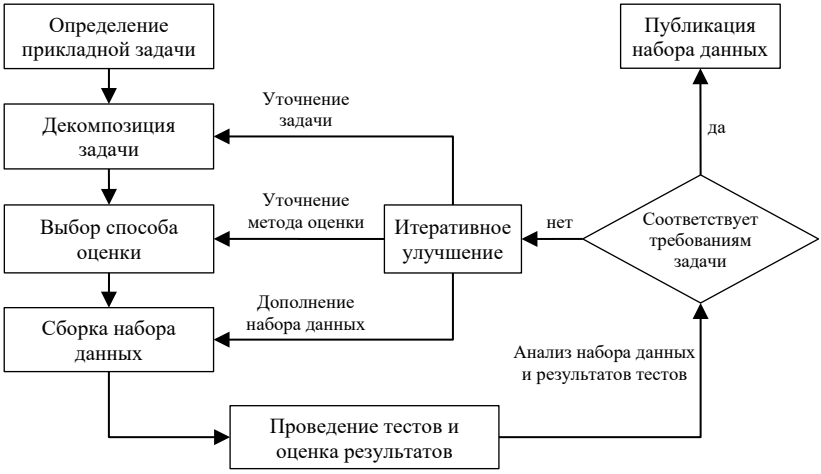

Рис. 1. Ключевые этапы процесса создания эталонного теста

Определение прикладной задачи. На первом этапе необходимо четко определить задачу для тестирования, где корректная работа LLM является важным компонентом. Чем детальнее будет определена задача и требования к ее выполнению на данном этапе, тем легче будет определить последующие шаги по разработке тестов. В задаче краткого изложения языковая модель должна не только провести анализ смысла текста, извлечь сущностную часть и кратко ее изложить, но и учесть инструкции, указывающие на ограничение

количества символов в результате, а также сохранить смысл всех важных фрагментов, выраженных числительными.

Декомпозиция задачи до минимальных простых подзадач. Этот шаг позволяет изолированно тестировать специфические навыки, и необходим для стандартизации тестов, объективизации оценки и унификации результатов.

Применительно к задаче краткого изложения и другим задачам, требующим работы с числительными, можно выделить следующие типы простых подзадач:

Соблюдение количественных ограничений:

– Генерация текста заданной длины (в символах, словах, предложениях, токенах).

– Выбор определенного количества элементов из списка.

– Ответ на вопрос с использованием точного числа.

Понимание и интерпретация числительных:

– Распознавание чисел, записанных цифрами (арабскими, римскими) и словами (на разных языках).

– Сравнение чисел.

– Выполнение простых арифметических инструкций, встроенных в текст.

– Извлечение числовых значений из текста.

Базовое понимание структуры текста (опосредованно влияющее на работу с числами в контексте):

– Определение слова, предложения, знака препинания.

– Понимание частей речи (существительное, прилагательное, глагол), что может быть важно для интерпретации инструкций с числами (например, «найди первые три существительных»).

– Понимание позиции слова в предложении.

Хотя нет возможности предвидеть все варианты корректных свободных ответов на сложную задачу целиком, можно провести тестирование на понимание числительных, используя задания с числами, представленными на разных языках и в разной форме (цифры, текст). Также есть возможность подготовить набор данных с заданным входным текстом и просить модель обрабатывать его или извлекать нужные части этого текста для точной сверки с эталоном.

Выбор способа оценки. На следующем этапе важно определить, каким способом будет оцениваться результат. В различных эталонных тестах используются разнообразные метрики и подходы:

– Точное совпадение (Exact Match, EM) [30, 13 – 15] – актуально для задач с единственно верным числовым ответом или при проверке точного воспроизведения числовых ограничений.

– F1-мера (F1 Score) – используется, когда важны и точность, и полнота (например, при извлечении набора чисел).

– IFEval [20] – оценка следования верифицируемым инструкциям (Instruction-Following Eval).

– BERTScore [32], ROUGE [32] – для оценки семантической близости и совпадения n-грамм, полезны при оценке качества сгенерированного текста, содержащего числа.

– Оценка человеком (Human Judgment) [33, 34] – незаменима для комплексной оценки, особенно для новых или сложных задач.

– Анкетирование и сравнение с эталоном [35] – структурированный сбор мнений экспертов или пользователей.

– Метрика pass@k [36] – часто используется в задачах генерации кода, но может быть адаптирована.

– Подход «LLM-as-a-judge» [37] – использование одной LLM для оценки результатов другой.

– Коэффициент корреляции Мэтьюса (Matthews Correlation Coefficient, MCC) [38].

Для задач, связанных с числительными, важны метрики, способные оценить точность (EM), соблюдение количественных ограничений (например, проверка длины) и корректность извлечения или генерации числовых значений. В таблице 3 представлены способы оценки, применяемые в различных эталонных тестах.

Сборка набора данных. Это самая трудоемкая часть работы. Важно учитывать следующие требования:

– источники данных должны быть релевантны задачам;

– данные должны быть подобраны так, чтобы максимально эффективно тестировать целевые навыки (например, разнообразие форматов числительных, разные диапазоны значений);

– возможна генерация синтетических данных или использование шаблонов для создания тестовых примеров с числами;

– нужна валидация данных на корректность, однозначность;

– набор данных должен быть достаточным по объему для получения статистически значимых результатов.

Проведение тестов и оценка результатов. Этот этап включает запуск эталонного теста на различных моделях и сопоставление результатов. Важно проводить оценку репрезентативности сделанных замеров и анализировать не только количественные показатели, но и качественные аспекты ошибок. Если полученный тест соответствует требованиям к задаче он отправляется на публикацию, иначе принимается решение о необходимости улучшения набора данных.

Итеративное улучшение. Построение эталонного теста представляет собой итеративный процесс, включающий регулярное пополнение набора данных новыми примерами, особенно для пограничных случаев и выявленных слабых мест моделей, а также корректировку формулировки задачи и добавление новых подзадач, что обеспечивает актуальность и релевантность эталонного теста.

Таблица 3. Способы оценки результата, применяемые в эталонных тестах

Название эталонного теста	**Метод оценки**
HARDMATH	EM, Human Evaluation
Набор задач DeepMind	EM, Accuracy
MATH	EM
MATHVISTA	Accuracy
FrontierMath	Human Evaluation
NumericBench	EM
NUPA	EM
Numberland	EM
MuSR	Human Evaluation
MMLU	Human Evaluation
MERA	EM, pass@k, Accuracy, Human Evaluation
SuperGLUE	Accuracy, EM
RussianSuperGLUE	Accuracy, EM, F1, Matthews Correlation Coefficient (MCC)
BIG-Bench Hard	Human Evaluation
IFEval	IFEval
GPQA	Human Evaluation

5. Пример использования предложенной методологии построение эталонных тестов. В качестве примера применения предложенной методологии рассмотрено построение эталонного теста для оценки «числового мышления» больших языковых моделей (LLM) на задаче краткого изложения текста. Эта задача, помимо требований сжатого и точного воспроизведения содержания, часто включает работу с количественными данными, что позволяет выявить способность LLM корректно интерпретировать и воспроизводить числовую информацию. Для исключения влияния лингвистической вариативности предложена декомпозиция на подзадачи: (1) проверка соблюдения количественных ограничений на длину ответа, включающая тестирование понимания различных форм числовых инструкций (цифрами и словами) и оценку процента успешного соблюдения; (2) проверка точности воспроизведения ключевых числовых данных с использованием

метрики точного совпадения (EM) для чисел. Такой подход обеспечивает изоляцию проверяемых навыков и позволяет выявлять сильные и слабые стороны LLM в аспекте числового мышления.

5.1. Описание структуры набора данных. Для автоматизации тестирования и обеспечения воспроизводимости результаты должны быть представлены в структурированном виде. В методологии предложен стандартизированный формат на основе JavaScript Object Nonation (JSON), обеспечивающий четкую структуру и однозначное разделение компонентов тестовых заданий.

Спецификация структуры JSON-объекта, представляющего одно тестовое задание, приведена в Листинге 1.

```
{
   "instruction": "Выполни текстовое задание {inputs}",
   "inputs": "Напиши сто пятнадцать",
   "outputs": ["115", "115.", "сто пятнадцать", "CXV"],
   "meta": {
      "id": 14,
      "task_type": "Кодирование числовых данных ",
      " type_input ": " arabic_num "
   }
}
```

Листинг 1. Пример структуры JSON-файла элементарного задания на проверку числового мышления

Каждый объект в Листинге 1 включает следующие обязательные поля:

– instruction: инкапсулирует базовую инструкцию для оцениваемой LLM, функционально эквивалентную системному промпту (system prompt). Для оценки базовых способностей LLM к числовому мышлению, формулировка инструкции целенаправленно исключает применение продвинутых методик промпт-инжиниринга (например, «Few-Shot» или «Chain-of-Thought»), если только тест не направлен на изучение их влияния на результат.

– inputs: содержит текст непосредственной задачи (например, исходный текст для краткого изложения или более конкретный вопрос на понимание числа).

– outputs: представляет собой список строк, каждая из которых является одним из допустимых эталонных (правильных) вариантов ответа.

– meta: вспомогательный раздел для хранения метаданных (например, идентификатор задачи, задача, тип записи задачи числовой или текст).

5.2. Описание созданного эталонного теста. По предложенной методологии построен эталонный тест MUE (Math Understanding Evaluation, https://github.com/cimsweb/MUE), который состоит из пяти групп тестовых заданий, каждая из которых оценивает отдельную способность LLM, составляющую «числовое мышление»:

– MUE-1 (Кодирование числовых данных). Оценивается способность LLM давать одинаково правильный результат на семантически эквивалентных задачах, несмотря на разную форму записи чисел.

– MUE-2 (Анализ текстовых элементов). Оценивается способность LLM идентифицировать и оперировать элементами текста по их количественным признакам (например, «напиши второе предложение», «сколько слов в этом тексте?», «какой текст короче?»), а также выполнять перестановку и замену элементов.

– MUE-3 (Количественный пересчет). Предназначен для оценки способности LLM выполнять подсчет объектов, указанных в инструкции или тексте (например, «Напиши три точки»).

– MUE-4 (Сравнение числовых значений). Проверяет способность LLM сравнивать численные значения (например, «Какое число больше: А) 1 999 000 или Б) 1 млн?»).

– MUE-5 (Интерпретация и применение формул). Оценивает понимание LLM общеизвестных количественных фактов, свойств объектов и единиц измерения (например, «Сколько секунд в сутках?»), а также способность конвертации единиц измерения и выполнения базовых статистических операций (медиана, среднее).

Каждая группа задач в наборе данных включает в себя 100 заданий, предназначенных для оценки языковых моделей. Количество заданий определяется проверяемой задачи, минимально достаточное количество определяется экспериментальным путем. Эти задания структурированы таким образом, чтобы охватить различные форматы представления числительных и языковые особенности постановки задачи. В частности, числительные записаны в следующих форматах: арабскими цифрами, римскими цифрами, русским текстом, английским текстом в русскоязычной постановке задания, английским текстом в англоязычной постановке задания. Эта многоформатная структура позволяет провести комплексный анализ робастности языковых моделей к вариациям в представлении численных данных и языковых контекстах.

Оценка задач в эталонном тесте MUE осуществляется с использованием метрики Exact Match (EM), в рамках которой для каждого примера присваивается значение 1, если целевая последовательность точно совпадает с предсказанной последовательностью (одним из вариантов ответов, перечисленных в поле outputs) (2). В противном случае присваивается значение 0. Общий результат вычисляется как среднее значение точности по всем последовательностям (3).

$$EM\left(A,R\right)=\begin{cases}1, A=R\\0, A\neq R\end{cases}, \tag{2}$$

$$EM_{\text{total}}=\frac{1}{N}\sum\nolimits_{i=1}^{N}EM\left(A_i,R_i\right). \tag{3}$$

При тестировании всех LLM параметр температуры в API устанавливается на низкое значение (0.1) для повышения детерминированности и воспроизводимости результатов. Перед сравнением ответов выполняется их очистка от артефактов форматирования (пробелов, кавычек и др.), затрудняющих оценку. В ходе исследований выявлены характерные особенности генерации, требующие унификации очистки и приведения результатов к единому формату, сопоставимому с эталонными ответами, с целью исключения ложных срабатываний при оценке. Целесообразно удалять пробелы, переносы строк, специальные символы в начале и конце текста. DeepSeek [22] добавляет дополнительные элементы разметки «*think*», в которых описывается процесс рассуждения. В экспериментах ожидается, что модель предоставит точный ответ, строго соответствующий формулировке задания. В статье IFEval [9, 20] предложен подход, позволяющий снизить количество ложно отрицательных ответов LLM, однако для текущего эталонного теста использование такого метода видится избыточным.

Для формирования воспроизводимых и интерпретируемых эталонных тестов, минимизирующих влияние субъективных факторов, необходимо изолированно тестировать отдельные навыки LLM с применением метрики Exact Match (EM) с целью оценки результатов. Предлагаемый эталонный тест MUE обеспечивает возможность комплексной оценки различных аспектов числового мышления LLM. Задача краткого изложения текста служит наглядным примером того, как возможности больших языковых моделей проявляются в прикладных задачах.

6. Результаты экспериментов. Для проведения экспериментов были выбраны следующие модели: GigaChat, GPT, Grok, Gemini, LLaMA, Yandex GPT, Qwen, DeepSeek, Mistral. Версии тестируемых моделей приведены в таблице 4 с результатами экспериментов.

Таблица 4. Результаты экспериментов

Модели	Тип ввода числительного	Тесты					
		MUE-1	MUE-2	MUE-3	MUE-4	MUE-5	AVG
GigaChat-2	arabic_num	0,95	0,05	0,15	0,55	0,75	0,49
	text_en	0,95	0,25	0,30	0,65	0,70	0,57
	roman_num	0,70	0,15	0,05	0,40	0,55	0,37
	text_ru	0,95	0,15	0,15	0,60	0,70	0,51
	text_ru_en	0,90	0,15	0,00	0,65	0,60	0,46
	total	0,89	0,15	0,13	0,57	0,66	0,48
GigaChat-2-Max	arabic_num	**1,00**	0,30	0,45	0,75	0,90	0,68
	text_en	**1,00**	0,40	0,55	0,80	0,85	0,72
	roman_num	0,80	0,40	0,35	0,65	0,55	0,55
	text_ru	**1,00**	0,35	0,55	0,85	0,85	0,72
	text_ru_en	**1,00**	0,35	0,50	0,80	0,85	0,70
	total	0,96	0,36	0,48	0,77	0,80	0,67
GigaChat-2-Pro	arabic_num	0,95	0,20	0,45	0,80	0,90	0,66
	text_en	0,95	0,35	0,35	0,80	0,85	0,66
	roman_num	0,80	0,20	0,20	0,65	0,50	0,47
	text_ru	0,95	0,30	0,55	0,90	0,85	0,71
	text_ru_en	0,80	0,25	0,35	0,80	0,80	0,60
	total	0,89	0,26	0,38	0,79	0,78	0,62
YandexGPT	arabic_num	0,90	0,25	0,25	0,65	0,90	0,59
	text_en	0,85	0,40	0,40	0,85	0,85	0,67
	roman_num	0,75	0,30	0,20	0,30	0,55	0,42
	text_ru	0,85	0,45	0,45	0,75	0,90	0,68
	text_ru_en	0,90	0,35	0,30	0,70	0,90	0,63
	total	0,85	0,35	0,32	0,65	0,82	0,60
yandexgpt-lite	arabic_num	0,85	0,10	0,15	0,30	0,90	0,46
	text_en	0,30	0,30	0,20	0,25	0,80	0,37
	roman_num	0,75	0,25	0,05	0,10	0,60	0,35
	text_ru	0,80	0,30	0,20	0,25	0,80	0,47
	text_ru_en	0,80	0,25	0,15	0,10	0,75	0,41
	total	0,70	0,24	0,15	0,20	0,77	0,41

Продолжение Таблицы 4

Модели	Тип ввода числительного	Тесты					
		MUE-1	MUE-2	MUE-3	MUE-4	MUE-5	AVG
gpt-4o	arabic_num	**1,00**	0,15	0,40	0,80	**1,00**	0,67
	text_en	**1,00**	0,40	**0,65**	0,65	0,80	0,70
	roman_num	0,85	0,20	0,30	0,60	0,70	0,53
	text_ru	**1,00**	0,40	0,60	0,85	0,85	0,74
	text_ru_en	**1,00**	0,30	0,45	0,80	0,80	0,67
	total	0,97	0,29	0,48	0,74	0,83	0,66
DeepSeek-R1-Distill-Llama-70B	arabic_num	0,95	0,30	0,30	0,75	0,55	0,57
	text_en	0,40	0,40	0,20	0,40	0,05	0,29
	roman_num	0,60	0,50	0,20	0,85	0,35	0,50
	text_ru	0,90	0,50	0,50	0,65	0,40	0,59
	text_ru_en	0,75	0,45	0,15	0,75	0,50	0,52
	total	0,72	0,43	0,27	0,68	0,37	0,49
Llama-3.3-70B-Instruct	arabic_num	0,50	0,20	0,20	0,80	0,85	0,51
	text_en	0,65	0,45	0,35	0,75	0,75	0,59
	roman_num	0,90	0,30	0,20	0,55	0,50	0,49
	text_ru	**1,00**	0,40	0,35	0,75	0,80	0,66
	text_ru_en	**1,00**	0,40	0,25	0,70	0,75	0,62
	total	0,81	0,35	0,27	0,71	0,73	0,57
RuadaptQwen2.5-32B-Pro-Beta	arabic_num	0,85	0,20	0,15	0,80	0,90	0,58
	text_en	0,80	0,25	0,25	0,60	0,80	0,54
	roman_num	0,90	0,30	0,15	0,70	0,65	0,54
	text_ru	**1,00**	0,25	0,40	0,85	0,85	0,67
	text_ru_en	**1,00**	0,30	0,40	0,75	0,85	0,66
	total	0,91	0,26	0,27	0,74	0,81	0,60
mistral-large	arabic_num	0,75	0,20	0,40	0,75	0,85	0,59
	text_en	0,75	0,20	0,35	0,70	0,60	0,52
	roman_num	0,80	0,25	0,25	0,35	0,50	0,43
	text_ru	**1,00**	0,40	0,45	0,70	0,75	0,66
	text_ru_en	0,95	0,40	0,30	0,65	0,75	0,61
	total	0,85	0,29	0,35	0,63	0,69	0,56
grok-3	arabic_num	**1,00**	0,25	0,30	0,80	0,95	0,66
	text_en	**1,00**	0,50	0,45	0,90	0,95	0,76
	roman_num	0,95	0,45	0,30	0,50	0,70	0,58
	text_ru	**1,00**	0,55	0,45	0,85	0,95	0,76
	text_ru_en	**1,00**	0,55	0,40	0,75	0,95	0,73
	total	0,99	0,46	0,38	0,76	0,90	0,70
gemini-2.5-flash	arabic_num	**1,00**	0,25	0,40	0,85	**1,00**	0,70
	text_en	0,95	0,50	0,55	**0,95**	0,95	0,78
	roman_num	**1,00**	0,55	0,30	0,75	0,90	0.70
	text_ru	**1,00**	**0,65**	0,55	**0,95**	0,95	**0,82**
	text_ru_en	**1,00**	**0,65**	0,45	**0,95**	0,95	0,80
	total	0,99	0,52	0,45	0,89	0,95	0,76

В контексте анализа сложных задач для больших языковых моделей (LLM) внимание уделяется обработке текстовых элементов и количественному пересчёту. Выявлено, что кодирование численных данных представляет собой наименее трудоёмкую задачу для большинства LLM (таблица 5).

Таблица 5. Оценка среднего значения EM по типу задачи

Задача	Среднее EM по всем экспериментам (робастность)
MUE-1 (Кодирование числовых данных)	0,88
MUE-2 (Анализ текстовых элементов)	0,33
MUE-3 (Количественный пересчет)	0,33
MUE-4 (Сравнение числовых значений)	0,68
MUE-5 (Интерпретация и применение формул)	0,76

Задача понимания текстовых элементов наряду с количественным пересчётом представляет наибольшую сложность. Задание вида «В ответ напиши только 2 предложения текста» оказалась сложным для многих моделей: LLM возвращали больше, либо меньше текста, чем требовалось, что свидетельствует о поверхностном «понимании» сути задания. Выполнение такого задания не представляет сложности для внимательного человека. В задаче количественного пересчета требуется владеть методами счёта, например, задание «Напиши в одну строку шесть раз слово «яблоко» и в два раза меньше слово «апельсин» содержит очевидную для человека ловушку, но для многих моделей оказывается сложным. Применимо к прикладным задачам следует использовать наиболее простую максимально прямолинейную одноязычную формулировку заданий, избегать сложных конструкций и зависимостей, для числительных использовать либо текстовое написание, либо арабские цифры без сокращений.

Исследование продемонстрировало, что эффективность LLM существенно возрастает при обработке заданий, полностью сформулированных на русском языке. Для повышения стабильности ответов LLM в прикладных задачах целесообразно использовать текстовое написание числительных. При этом инструкции, содержащие римские числа, значительно снижают точность и корректность ответов моделей (таблица 6).

Согласно усреднённой оценке, полученной на данном эталонном тесте, модель семейства Gemini продемонстрировала наивысшую производительность, заняв лидирующую позицию. Второе место в рейтинге эффективности заняла модель семейства Grok, тогда

как модели GigaChat расположились на третьей позиции (таблица 7). Для выявления лидера в прикладных задачах следует проводить тесты разных LLM.

Таблица 6. Оценка среднего значения EM по типу ввода числового значения

Тип ввода числового значения	Среднее EM по всем экспериментам (робастность)
arabic_num	0,60
En	0,60
roman_num	0,49
Ru	0,67
ru_en	0,62

Таблица 7. Оценка среднего значения EM по LLM модели

LLM модель	Среднее EM по всем экспериментам (робастность)
GigaChat-2	0,48
GigaChat-2-Max	0,67
GigaChat-2-Pro	0,62
Yandexgpt	0,60
yandexgpt-lite	0,41
gpt-4o	0,66
DeepSeek-R1-Distill-Llama-70B	0,49
Llama-3.3-70B-Instruct	0,57
RuadaptQwen2.5-32B-Pro-Beta	0,60
mistral-large	0,56
grok-3	0,70
gemini-2.5-flash	0,76

Эталонный тест MUE оценивает пять навыков числового мышления: кодирование числовых данных, анализ текстовых элементов, количественный пересчёт, сравнение числовых значений, а также интерпретацию и применение формул. Экспериментальные результаты выявили ограничения LLM при решении задач, которые для человека являются простыми. Установлено, что на корректность работы модели оказывает влияние способ представления числительных в задании. Основные причины наблюдаемых ошибок связаны, во-первых, с особенностями токенизаторов, формирующих различные векторные представления для разных способов записи чисел, и, во-вторых, с архитектурными ограничениями LLM, затрудняющими решение численных задач. Для повышения точности в прикладных сценариях рекомендуется формулировать инструкции

максимально ясно, избегать сложных синтаксических конструкций, сокращений и противоречивых условий. Стабилизация качества ответов может быть достигнута путём эмпирического подбора наиболее устойчивых формулировок промпта из числа семантически эквивалентных вариантов. Выполнение арифметических операций целесообразно возлагать на специализированные функции, разработанные для решения соответствующих задач.

Заключение. В работе предложена методология построения эталонных тестов для оценки работы больших языковых моделей (LLM) с акцентом на их способность интерпретировать и обрабатывать числовую информацию. В качестве примера подготовлен эталонный тест MUE (Math Understanding Evaluation), включающий наборы данных для изолированной проверки компонентов числового мышления LLM. Экспериментальные исследования, выполненные с использованием MUE, выявили влияние способа записи числительных на результаты, а также подтвердили применимость методологии для объективного сравнения LLM и оценки их пригодности в прикладных сценариях, требующих высокой точности работы с числами.

Результаты исследования подчёркивают необходимость регулярного автоматизированного тестирования LLM, интегрируемых в информационные системы, для обеспечения их надёжности, безопасности и предсказуемости. Приоритетными направлениями развития являются создание токенизаторов, учитывающих специфику числовых данных, формирование целей обучения, ориентированных на численный анализ, и разработка специализированных эталонных тестов для различных прикладных и исследовательских задач. Сравнение результатов в разные периоды позволит отслеживать динамику и фиксировать прогресс в развитии навыков числового мышления у LLM.

Авторы посвящают настоящую статью памяти доктора технических наук, члена-корреспондента Российской академии наук, заслуженного деятеля науки и техники Российской Федерации Юсупова Рафаэля Мидхатовича. Его выдающиеся результаты внесли значительный вклад в развитие отечественной научной школы и формирование современного научного мышления во многих областях информатики, информационных технологий и теории управления. Научное наследие Юсупова Р.М. продолжает служить основой для дальнейших исследований и остаётся важной частью достижений отечественной и мировой науки и техники.

Литература

1. Radford A., Narasimhan K., Salimans T., Sutskever I. Improving language understanding by generative pre-training. 2018.
2. Team G., Anil R., Borgeaud S., Alayrac J.B., Yu J., Soricut R., Blanco L, et al. Gemini: a family of highly capable multimodal models. 2023. arXiv preprint arXiv:2312.11805.
3. Jiang A.Q., Sablayrolles A., Mensch A., Bamford C., Chaplot D.S., Casas D.D., Sayed W.E., et al. Mistral 7B. 2023. arXiv preprint arXiv:2310.06825.
4. Touvron H., Lavril T., Izacard G., Martinet X., Lachaux M.A., Lacroix T., Rozière B., Goyal N., Hambro E., Azhar F., Rodriguez A., Joulin A., Grave E., Lample G. Llama: Open and efficient foundation language models. 2023. arXiv preprint arXiv:2302.13971.
5. Дагаев А.Е., Попов Д.И. Сравнение автоматического обобщения текстов на русском языке. Программные системы и вычислительные методы. 2024. Т. 4. С. 13–22. DOI: 10.7256/2454-0714.2024.4.69474.
6. Tsanda A., Bruches E. Russian-Language Multimodal Dataset for Automatic Summarization of Scientific Papers. 2024. arXiv preprint arXiv:2405.07886.
7. Liu A., Feng B., Wang B., Wang B., Liu B., Zhao C., Dengr C., Ruan C., Dai D., Guo D., Yang D., et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. 2024. arXiv preprint arXiv:2405.04434.
8. Fan J., Martinson S., Wang E.Y., Hausknecht K., Brenner J., Liu D., Peng N., Wang C., Brenner M.P. HARDMath: A Benchmark Dataset for Challenging Problems in Applied Mathematics. 2024. arXiv preprint arXiv:2410.09988.
9. Saxton D., Grefenstette E., Hill F., Kohli P. Analysing mathematical reasoning abilities of neural models. 2019. arXiv preprint arXiv:1904.01557.
10. Hendrycks D., Burns C., Kadavath S., Arora A., Basart S., Tang E., Song D., Steinhardt J. Measuring mathematical problem solving with the math dataset. 2021. arXiv preprint arXiv:2103.03874.
11. Lu P., Bansal H., Xia T., Liu J., Li C., Hajishirzi H., Cheng H., Chang K.W., Galley M., Gao J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. 2023. arXiv preprint arXiv:2310.02255.
12. Glazer E., Erdil E., Besiroglu T., Chicharro D., Chen E., Gunning A., Olsson C.F., Denain J.S., Ho A., Santos E.D., Järviniemi O., et al. Frontiermath: A benchmark for evaluating advanced mathematical reasoning in ai. 2024. arXiv preprint arXiv:2411.04872.
13. Li H., Chen X., Xu Z., Li D., Hu N., Teng F., Li Y., Qiu L., Zhang C.J., Li Q., Chen L. Exposing numeracy gaps: A benchmark to evaluate fundamental numerical abilities in large language models. 2025. arXiv preprint arXiv:2502.11075.
14. Yang H., Hu Y., Kang S., Lin Z., Zhang M. Number cookbook: Number understanding of language models and how to improve it (2024). arXiv preprint arXiv:2411.03766.
15. Rahman R. Large Language Models in Numberland: A Quick Test of Their Numerical Reasoning Abilities. 2025. arXiv preprint arXiv:2504.00226.
16. Sennrich R., Haddow B., Birch A. Neural machine translation of rare words with subword units. 2015. arXiv preprint arXiv:1508.07909.
17. Schuster M., Nakajima K. Japanese and korean voice search. IEEE international conference on acoustics, speech and signal processing (ICASSP). 2012. pp. 5149–5152.
18. Kudo T., Richardson J. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. 2018. arXiv preprint arXiv:1808.06226.

19. Myrzakhan A., Bsharat S.M., Shen Z. Open-LLM-Leaderboard: From Multi-choice to Open-style Questions for LLMs Evaluation, Benchmark, and Arena. 2024. arXiv preprint arXiv:2406.07545.
20. Zhou J., Lu T., Mishra S., Brahma S., Basu S., Luan Y., Zhou D., Hou L. Instruction-following evaluation for large language models. 2023. arXiv preprint arXiv:2311.07911.
21. Suzgun M., Scales N., Schärli N., Gehrmann S., Tay Y., Chung H.W., Chowdhery A., Le Q.V., Chi E.H., Zhou D., Wei J. Challenging big-bench tasks and whether chain-of-thought can solve them. 2022. arXiv preprint arXiv:2210.09261.
22. Rein D., Hou B.L., Stickland A.C., Petty J., Pang R.Y., Dirani J., Michael J., Bowman S.R. Gpqa: A graduate-level google-proof q&a benchmark. 2023. arXiv preprint arXiv:2311.12022.
23. Sprague Z., Ye X., Bostrom K., Chaudhuri S., Durrett G. Musr: Testing the limits of chain-of-thought with multistep soft reasoning. 2023. arXiv preprint arXiv:2310.16049.
24. Hendrycks D., Burns C., Basart S., Zou A., Mazeika M., Song D., Steinhardt J. Measuring massive multitask language understanding. 2020. arXiv preprint arXiv:2009.03300.
25. Fenogenova A., Chervyakov A., Martynov N., Kozlova A., Tikhonova M., Akhmetgareeva A., Emelyanov A., Shevelev D., Lebedev P., Sinev L., Isaeva U., et al. Mera: A comprehensive LLM evaluation in Russian. 2024. arXiv preprint arXiv:2401.04531.
26. Wang A., Pruksachatkun Y., Nangia N., Singh A., Michael J., Hill F., Levy O., Bowman S. Superglue: A stickier benchmark for general-purpose language understanding systems. Advances in neural information processing systems. 2019. vol. 32.
27. Shavrina T., Fenogenova A., Emelyanov A., Shevelev D., Artemova E., Malykh V., Mikhailov V., Tikhonova M., Chertok A., Evlampiev A. RussianSuperGLUE: A Russian language understanding evaluation benchmark. 2020. arXiv preprint arXiv:2010.15925.
28. Ribeiro M.T., Wu T., Guestrin C., Singh S. Beyond accuracy: Behavioral testing of NLP models with CheckList. 2020. arXiv preprint arXiv:2005.04118.
29. Luo H., Sun Q., Xu C., Zhao P., Lou J., Tao C., Geng X., Lin Q., Chen S., Zhang D. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. 2023. arXiv preprint arXiv:2308.09583.
30. Zhang T., Kishore V., Wu F., Weinberger K.Q., Artzi Y. Bertscore: Evaluating text generation with bert. 2019. arXiv preprint arXiv:1904.09675.
31. Lin C.Y. Rouge: A package for automatic evaluation of summaries. Text summarization branches out. 2004. pp. 74–81.
32. Rajpurkar P., Zhang J., Lopyrev K., Liang P. Squad: 100,000+ questions for machine comprehension of text. 2016. arXiv preprint arXiv:1606.05250.
33. Amigó E., Giménez J., Gonzalo J., Màrquez L. MT evaluation: Human-like vs. human acceptable. Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions. 2006. pp. 17–24.
34. Celikyilmaz A., Clark E., Gao J. Evaluation of text generation: A survey. 2020. arXiv preprint arXiv:2006.14799.
35. Dong C., Li Y., Gong H., Chen M., Li J., Shen Y., Yang M. A survey of natural language generation. ACM Computing Surveys. 2022. vol. 55(8). pp. 1–38.
36. Chen M., Tworek J., Jun H., Yuan Q., Pinto H.P., Kaplan J., Edwards H., Burda Y., Joseph N., Brockman G., Ray A., et al. Evaluating large language models trained on code. 2021. arXiv preprint arXiv:2107.03374.

37. Zheng L., Chiang W.L., Sheng Y., Zhuang S., Wu Z., Zhuang Y., Lin Z., Li Z., Li D., Xing E., Zhang H. Judging llm-as-a-judge with mt-bench and chatbot arena. Advances in Neural Information Processing Systems. 2023. vol. 36. 46595–46623.
38. Chicco D., Jurman G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC genomics. 2020. vol. 21. pp. 1–3.

Карпович Сергей Николаевич — канд. техн. наук, директор по информационным технологиям, Общество с ограниченной ответственностью «Рамблер ДС». Область научных интересов: тематическое моделирование, обработка текстов на естественном языке, информационный поиск, машинное обучение. Число научных публикаций — 13. cims@yandex.ru; Варшавское шоссе, 9/1, 117105, Москва, Россия; р.т.: +7(812)328-8071.

Смирнов Александр Викторович — д-р техн. наук, профессор, главный научный сотрудник, руководитель лаборатории, лаборатория интегрированных систем автоматизации, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН). Область научных интересов: системы поддержки принятия решений, интеллектуальные системы, интеллектуальное управление конфигурациями виртуальных и сетевых организаций, логистика знаний. Число научных публикаций — 450. smir@iias.spb.su; 14-я линия В.О., 39, 199178, Санкт-Петербург, Россия; р.т.: +7(812)328-8071.

Тесля Николай Николаевич — канд. техн. наук, старший научный сотрудник, лаборатория интегрированных систем автоматизации, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН). Область научных интересов: управление знаниями, онтологии, обработка текста на естественном языке, языковые модели. Число научных публикаций — 100. teslya@iias.spb.su; 14-я линия В.О., 39, 199178, Санкт-Петербург, Россия; р.т.: +7(812)328-8071.

Поддержка исследований. Работа выполнена в рамках бюджетной темы FFZF-2025-0003.

DOI 10.15622/ia.24.6.7

S. KARPOVICH, A. SMIRNOV, N. TESLYA

METHODOLOGY FOR CREATING A BENCHMARK TO EVALUATE LLM PERFORMANCE ON NUMERALS

Karpovich S., Smirnov A., Teslya N. **Methodology for Creating a Benchmark to Evaluate LLM Performance on Numerals.**

Abstract. The article presents a methodology for designing a benchmark to assess numerical reasoning skills in Large Language Models (LLMs). In the context of LLMs, numerical reasoning is defined as a model's ability to correctly interpret, process, and utilize numerical information in text, including understanding magnitudes and relations between numbers, performing arithmetic operations, and generating numerals accurately in its outputs. The proposed methodology is based on decomposing applied tasks and enables targeted evaluation of specific facets of numerical reasoning using tasks that involve numerals. Particular attention is paid to the representation of numbers in textual prompts to LLMs, as this factor directly affects the quality of the final output. The need for rigorous assessment of LLMs' numerical reasoning stems from its critical role across a wide range of text-centric applications, including automated summarization, generation of analytical reports, extraction and interpretation of quantitative data, and conversational systems operating on financial, scientific, or technical information. Drawing on an analysis of state-of-the-art LLM evaluation approaches, core principles for constructing evaluation benchmarks are formulated with an emphasis on generality and real-world applicability. In accordance with the proposed methodology, the MUE (Math Understanding Evaluation) benchmark is introduced; it comprises five test suites, each designed to assess a distinct aspect of LLM numerical reasoning. A comparative evaluation of popular LLMs is conducted, leading models are identified, and the strengths and weaknesses of their numerical reasoning are characterized. The findings are intended to inform LLM developers in refining architectures and training strategies, and to guide end users and integrators in selecting an optimal model for applied projects.

Keywords: methodology, Large Language Models (LLM), LLM benchmark, Natural Language Processing (NLP), numerals.

References

1. Radford A., Narasimhan K., Salimans T., Sutskever I. Improving language understanding by generative pre-training. 2018.
2. Team G., Anil R., Borgeaud S., Alayrac J.B., Yu J., Soricut R., Blanco L, et al. Gemini: a family of highly capable multimodal models. 2023. arXiv preprint arXiv:2312.11805.
3. Jiang A.Q., Sablayrolles A., Mensch A., Bamford C., Chaplot D.S., Casas D.D., Sayed W.E., et al. Mistral 7B. 2023. arXiv preprint arXiv:2310.06825.
4. Touvron H., Lavril T., Izacard G., Martinet X., Lachaux M.A., Lacroix T., Rozière B., Goyal N., Hambro E., Azhar F., Rodriguez A., Joulin A., Grave E., Lample G. Llama: Open and efficient foundation language models. 2023. arXiv preprint arXiv:2302.13971.
5. Dagaev A.E., Popov D.I. [Comparison of automatic summarization of texts in Russian]. Programmnye sistemy i vychislitel'nye metody – Software systems and computational methods. 2024. vol. 4. pp. 13–22. DOI: 10.7256/2454-0714.2024.4.69474. (In Russ.).

6. Tsanda A., Bruches E. Russian-Language Multimodal Dataset for Automatic Summarization of Scientific Papers. 2024. arXiv preprint arXiv:2405.07886.
7. Liu A., Feng B., Wang B., Wang B., Liu B., Zhao C., Dengr C., Ruan C., Dai D., Guo D., Yang D., et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. 2024. arXiv preprint arXiv:2405.04434.
8. Fan J., Martinson S., Wang E.Y., Hausknecht K., Brenner J., Liu D., Peng N., Wang C., Brenner M.P. HARDMath: A Benchmark Dataset for Challenging Problems in Applied Mathematics. 2024. arXiv preprint arXiv:2410.09988.
9. Saxton D., Grefenstette E., Hill F., Kohli P. Analysing mathematical reasoning abilities of neural models. 2019. arXiv preprint arXiv:1904.01557.
10. Hendrycks D., Burns C., Kadavath S., Arora A., Basart S., Tang E., Song D., Steinhardt J. Measuring mathematical problem solving with the math dataset. 2021. arXiv preprint arXiv:2103.03874.
11. Lu P., Bansal H., Xia T., Liu J., Li C., Hajishirzi H., Cheng H., Chang K.W., Galley M., Gao J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. 2023. arXiv preprint arXiv:2310.02255.
12. Glazer E., Erdil E., Besiroglu T., Chicharro D., Chen E., Gunning A., Olsson C.F., Denain J.S., Ho A., Santos E.D., Järviniemi O., et al. Frontiermath: A benchmark for evaluating advanced mathematical reasoning in ai. 2024. arXiv preprint arXiv:2411.04872.
13. Li H., Chen X., Xu Z., Li D., Hu N., Teng F., Li Y., Qiu L., Zhang C.J., Li Q., Chen L. Exposing numeracy gaps: A benchmark to evaluate fundamental numerical abilities in large language models. 2025. arXiv preprint arXiv:2502.11075.
14. Yang H., Hu Y., Kang S., Lin Z., Zhang M. Number cookbook: Number understanding of language models and how to improve it (2024). arXiv preprint arXiv:2411.03766.
15. Rahman R. Large Language Models in Numberland: A Quick Test of Their Numerical Reasoning Abilities. 2025. arXiv preprint arXiv:2504.00226.
16. Sennrich R., Haddow B., Birch A. Neural machine translation of rare words with subword units. 2015. arXiv preprint arXiv:1508.07909.
17. Schuster M., Nakajima K. Japanese and korean voice search. IEEE international conference on acoustics, speech and signal processing (ICASSP). 2012. pp. 5149–5152.
18. Kudo T., Richardson J. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. 2018. arXiv preprint arXiv:1808.06226.
19. Myrzakhan A., Bsharat S.M., Shen Z. Open-LLM-Leaderboard: From Multi-choice to Open-style Questions for LLMs Evaluation, Benchmark, and Arena. 2024. arXiv preprint arXiv:2406.07545.
20. Zhou J., Lu T., Mishra S., Brahma S., Basu S., Luan Y., Zhou D., Hou L. Instruction-following evaluation for large language models. 2023. arXiv preprint arXiv:2311.07911.
21. Suzgun M., Scales N., Schärli N., Gehrmann S., Tay Y., Chung H.W., Chowdhery A., Le Q.V., Chi E.H., Zhou D., Wei J. Challenging big-bench tasks and whether chain-of-thought can solve them. 2022. arXiv preprint arXiv:2210.09261.
22. Rein D., Hou B.L., Stickland A.C., Petty J., Pang R.Y., Dirani J., Michael J., Bowman S.R. Gpqa: A graduate-level google-proof q&a benchmark. 2023. arXiv preprint arXiv:2311.12022.
23. Sprague Z., Ye X., Bostrom K., Chaudhuri S., Durrett G. Musr: Testing the limits of chain-of-thought with multistep soft reasoning. 2023. arXiv preprint arXiv:2310.16049.

24. Hendrycks D., Burns C., Basart S., Zou A., Mazeika M., Song D., Steinhardt J. Measuring massive multitask language understanding. 2020. arXiv preprint arXiv:2009.03300.
25. Fenogenova A., Chervyakov A., Martynov N., Kozlova A., Tikhonova M., Akhmetgareeva A., Emelyanov A., Shevelev D., Lebedev P., Sinev L., Isaeva U., et al. Mera: A comprehensive LLM evaluation in Russian. 2024. arXiv preprint arXiv:2401.04531.
26. Wang A., Pruksachatkun Y., Nangia N., Singh A., Michael J., Hill F., Levy O., Bowman S. Superglue: A stickier benchmark for general-purpose language understanding systems. Advances in neural information processing systems. 2019. vol. 32.
27. Shavrina T., Fenogenova A., Emelyanov A., Shevelev D., Artemova E., Malykh V., Mikhailov V., Tikhonova M., Chertok A., Evlampiev A. RussianSuperGLUE: A Russian language understanding evaluation benchmark. 2020. arXiv preprint arXiv:2010.15925.
28. Ribeiro M.T., Wu T., Guestrin C., Singh S. Beyond accuracy: Behavioral testing of NLP models with CheckList. 2020. arXiv preprint arXiv:2005.04118.
29. Luo H., Sun Q., Xu C., Zhao P., Lou J., Tao C., Geng X., Lin Q., Chen S., Zhang D. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. 2023. arXiv preprint arXiv:2308.09583.
30. Zhang T., Kishore V., Wu F., Weinberger K.Q., Artzi Y. Bertscore: Evaluating text generation with bert. 2019. arXiv preprint arXiv:1904.09675.
31. Lin C.Y. Rouge: A package for automatic evaluation of summaries. Text summarization branches out. 2004. pp. 74–81.
32. Rajpurkar P., Zhang J., Lopyrev K., Liang P. Squad: 100,000+ questions for machine comprehension of text. 2016. arXiv preprint arXiv:1606.05250.
33. Amigó E., Giménez J., Gonzalo J., Màrquez L. MT evaluation: Human-like vs. human acceptable. Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions. 2006. pp. 17–24.
34. Celikyilmaz A., Clark E., Gao J. Evaluation of text generation: A survey. 2020. arXiv preprint arXiv:2006.14799.
35. Dong C., Li Y., Gong H., Chen M., Li J., Shen Y., Yang M. A survey of natural language generation. ACM Computing Surveys. 2022. vol. 55(8). pp. 1–38.
36. Chen M., Tworek J., Jun H., Yuan Q., Pinto H.P., Kaplan J., Edwards H., Burda Y., Joseph N., Brockman G., Ray A., et al. Evaluating large language models trained on code. 2021. arXiv preprint arXiv:2107.03374.
37. Zheng L., Chiang W.L., Sheng Y., Zhuang S., Wu Z., Zhuang Y., Lin Z., Li Z., Li D., Xing E., Zhang H. Judging llm-as-a-judge with mt-bench and chatbot arena. Advances in Neural Information Processing Systems. 2023. vol. 36. 46595–46623.
38. Chicco D., Jurman G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC genomics. 2020. vol. 21. pp. 1–3.

Karpovich Sergey — Ph.D., Chief Information Officer, LLC “Rambler DC”. Research interests: topic modeling, natural language processing, information retrieval, machine learning. The number of publications — 13. cims@yandex.ru; 9/1, Varshavskoe Hwy, 117105, Moscow, Russia; office phone: +7(812)328-8071.

Smirnov Alexander — Ph.D., Dr.Sci., Professor, Chief researcher, head of the laboratory, Computer-aided integrated systems laboratory, St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS). Research interests: intelligent configuration management of virtual and networked organizations, knowledge logistics, decision support.

The number of publications — 450. smir@iias.spb.su; 39, 14-th Line V.O., 199178, St. Petersburg, Russia; office phone: +7(812)328-8071.

Teslya Nikolay — Ph.D., Senior researcher, Computer-aided integrated systems laboratory, St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS). Research interests: ontologies, knowledge bases, knowledge management, natural language processing. The number of publications — 100. teslya@iias.spb.su; 39, 14-th Line V.O., 199178, St. Petersburg, Russia; office phone: +7(812)328-8071.

Acknowledgements. This research is funded by the state research FFZF-2025-0003.