Методика приведения производительности компьютерных комплектующих к единой шкале на основе кросс-бенчмарк анализа

Великанов Михаил Николаевич[1,*] **Седых Игорь Вячеславович**[2]

[1]ИИКС НИЯУ МИФИ, Москва, Россия
vmn005@campus.mephi.ru

[2]ИИКС НИЯУ МИФИ, Москва, Россия
ivsedykh@mephi.ru

25 мая 2026 г.

[*]Автор для переписки: vmn005@campus.mephi.ru

Вклад авторов: Великанов М.Н. — концепция исследования, разработка методики, сбор данных, программная реализация, написание текста; Седых И.В. — анализ результатов, визуализация данных, редактирование текста.

Доступность кода: Исходный код и данные доступны по ссылке:
https://github.com/8skpd/CROSS-BENCHMARK-ANALYSIS

Аннотация

В статье рассматривается проблема несопоставимости данных производительности компьютерных процессоров из-за использования различными бенчмарками несовместимых шкал измерения в условных единицах. Для решения задачи приведения разнородных бенчмарк-данных к единой шкале проведен сравнительный анализ семи методов нормализации, включая Min-Max Scaling, Z-score Standardization, Robust Scaling и Logarithmic Scaling, на выборке из 23 моделей процессоров, присутствующих в бенчмарках UserBenchmark и PassMark одновременно. Оценка качества кросс-бенчмарк перевода осуществлялась с помощью метрик MAE, RMSE, MAPE и коэффициентов корреляции Пирсона и Спирмена. Экспериментальные результаты показали, что метод логарифмического масштабирования (Log) продемонстрировал наилучшую точность с MAE = 0.0340 и MAPE = 4.20%, в то время как метод ранжирования (Rank) обеспечил наилучшее сохранение порядка компонентов с коэффициентом Спирмена 0.959. Все методы нормализации показали высокую корреляцию (Pearson $>$ 0.89, Spearman = 0.959), что подтверждает возможность эффективного кросс-бенчмарк сопоставления. Полученные результаты могут быть применены для повышения точности автоматизированных систем подбора компьютерных конфигураций и агрегации бенчмарк-данных.

Ключевые слова: бенчмарк, нормализация данных, производительность процессоров, кросс-бенчмарк анализ, масштабирование, рекомендательная система.

Methodology for Normalizing Computer Hardware Performance to a Unified Scale Based on Cross-Benchmark Analysis

Mikhail N. Velikanov[1,*] and **Igor V. Sedykh**[2]

[1]ICIS NRNU MEPhI, Moscow, Russia
vmn005@campus.mephi.ru

[2]ICIS NRNU MEPhI, Moscow, Russia
ivsedykh@mephi.ru

[*]Corresponding author: vmn005@campus.mephi.ru

Author Contributions: Velikanov M.N. — research concept, methodology development, data collection, software implementation, writing; Sedykh I.V. — results analysis, data visualization, editing.

Code Availability: Source code and data are available at:
https://github.com/8skpd/CROSS-BENCHMARK-ANALYSIS

Abstract

This paper addresses the problem of incomparability in computer hardware performance data arising from the use of incompatible measurement scales across different benchmarks. To solve the problem of aligning heterogeneous benchmark data to a unified scale, we conducted a comparative analysis of seven normalization methods, including Min-Max Scaling, Z-score Standardization, Robust Scaling, and Logarithmic Scaling, on a dataset of 23 CPU models present in both UserBenchmark and PassMark simultaneously. The quality of cross-benchmark translation was assessed using MAE, RMSE, MAPE metrics and Pearson and Spearman correlation coefficients. Experimental results showed that the Logarithmic Scaling method demonstrated the best accuracy with MAE = 0.0340 and MAPE = 4.20%, while the Rank Transformation method provided the best preservation of component ordering with Spearman's coefficient of 0.959. All normalization methods showed high correlation (Pearson ¿ 0.89, Spearman = 0.959), confirming the feasibility of effective cross-benchmark alignment. The obtained results can be applied to improve the accuracy of automated PC configuration recommendation systems and benchmark data aggregation.

Keywords: benchmark, data normalization, CPU performance, cross-benchmark analysis, scaling, recommendation system.

1 Введение

В современном мире компьютерных технологий выбор аппаратного обеспечения становится всё более сложной задачей для пользователей. Как отмечается в литературе, рынок процессоров характеризуется высокой динамикой обновления: новые архитектуры появляются ежегодно, а количество моделей исчисляется сотнями. Для оценки производительности оборудования индустрия использует множество бенчмарков, таких как UserBenchmark, PassMark, SPEC CPU и Cinebench. Каждый из этих инструментов применяет собственную методологию тестирования и уникальную шкалу измерения результатов в условных единицах.

Несмотря на обилие источников данных о производительности, прямое сравнение баллов из разных бенчмарков невозможно. Пользователь, пытающийся сопоставить рейтинг процессора из UserBenchmark с баллом PassMark, сталкивается с фундаментальной проблемой несопоставимости шкал. Эта проблема усугубляется тем, что различные бенчмарки используют разные сценарии тестирования: одни оценивают однопоточную производительность, другие — многопоточную, третьи — специализированные рабочие нагрузки. В результате возникает ситуация, когда один и тот же

компонент может иметь высокий рейтинг в одном бенчмарке и средний в другом, что затрудняет объективную оценку.

Особую актуальность эта проблема приобретает в контексте разработки автоматизированных рекомендательных систем для подбора компьютерных конфигураций. Такие системы требуют унифицированных данных о производительности компонентов для корректного сопоставления требований пользователя с характеристиками оборудования. Существующие решения, такие как PCPartPicker, полагаются на ручную калибровку и экспертные оценки, что ограничивает их масштабируемость и объективность. Отсутствие общепринятой методики приведения разнородных бенчмарк-данных к единой шкале сдерживает развитие интеллектуальных систем рекомендаций в этой области.

В научной литературе вопросы бенчмаркинга компьютерных систем широко освещены в работах консорциума SPEC. Как отмечает Henning [1], «прямое сравнение баллов из разных бенчмарков без предварительной нормализации приводит к статистически некорректным выводам» (с. 31). Однако проблема кросс-бенчмарк нормализации — то есть перевода баллов между различными системами измерения — остаётся, по-видимому, недостаточно изученной. Большинство существующих подходов используют простое линейное масштабирование, которое не учитывает распределение данных, наличие выбросов и нелинейность роста производительности между поколениями оборудования.

Целью данной работы является разработка и сравнительный анализ методик нормализации бенчмарк-данных для приведения производительности компьютерных процессоров к единой шкале. Для достижения поставленной цели решаются следующие задачи: сбор и верификация датасета на основе пересечения компонентов в нескольких источниках бенчмарков; реализация и тестирование различных методов нормализации; оценка точности перевода между шкалами с использованием статистических метрик; формулировка практических рекомендаций по выбору метода нормализации для различных сценариев использования.

Научная новизна исследования заключается в систематическом сравнении методов нормализации применительно к данным компьютерных бенчмарков, оценке устойчивости методов к выбросам и изменению поколения оборудования, а также в разработке метрики качества перевода, позволяющей оценить потерю информации при нормализации. В отличие от предыдущих работ, рассматривающих нормализацию в общем контексте машинного обучения, данное исследование фокусируется на специфике бенчмарк-данных компьютерного оборудования.

Практическая значимость работы состоит в возможности применения разработанной методики в системах автоматического подбора компьютерных конфигураций, агрегаторах бенчмарк-данных и аналитических платформах для сравнения оборудования. Результаты исследования позволяют сократить ошибку сравнения компонентов из разных источников и повысить объективность рекомендаций для конечных пользователей.

Статья организована следующим образом. Во втором разделе представлен обзор литературы по методам бенчмаркинга и нормализации данных. Третий раздел описывает методологию исследования, включая источники данных, методы нормализации и метрики оценки. Четвёртый раздел содержит результаты экспериментов и их анализ. В пятом разделе обсуждаются полученные результаты, ограничения исследования и направления будущей работы. Шестой раздел заключает статью основными выводами.

2 Обзор литературы

Проблема оценки и сравнения производительности компьютерных компонентов является предметом исследования на протяжении нескольких десятилетий. В данном разделе представлен анализ

существующих подходов к бенчмаркингу, методов нормализации данных и систем рекомендаций аппаратного обеспечения.

2.1 Методология бенчмаркинга компьютерных систем

Вопросы методологии бенчмаркинга широко освещены в работах консорциума Standard Performance Evaluation Corporation (SPEC). Henning [1] в фундаментальной работе по SPEC CPU2000 описывает принципы построения эталонных тестов, включая требования к репрезентативности рабочих нагрузок, воспроизводимости результатов и верифицируемости измерений. Автор подчёркивает, что «использование геометрического среднего для агрегации баллов по нескольким тестам стало отраслевым стандартом, обеспечивающим сопоставимость результатов независимо от абсолютных значений» (с. 32).

Kistowski et al. [2] в работе «How to Build a Benchmark» формулируют пять критериев качества бенчмарка: релевантность (Relevance), воспроизводимость (Reproducibility), справедливость (Fairness), верифицируемость (Verifiability) и удобство использования (Usability). Как отмечают авторы, «воспроизводимость результатов является необходимым, но не достаточным условием для признания бенчмарка достоверным» (p. 334). Эти критерии могут быть применены для оценки качества методик кросс-бенчмарк нормализации, предлагаемых в данном исследовании.

Zablah et al. [3] проводят комплексный исторический обзор эволюции бенчмарков от Whetstone и Dhrystone до современных SPEC CPU2017 и MLPerf. Авторы отмечают тенденцию к увеличению сложности тестовых сценариев и приближению их к реальным рабочим нагрузкам, что приводит к усложнению сравнения результатов между разными версиями бенчмарков.

2.2 Проблемы сравнения производительности между бенчмарками

Проблема несопоставимости данных из различных источников бенчмаркинга обсуждается в ряде исследований. Mytkowicz et al. [4] в работе «Producing Wrong Data Without Doing Anything Obviously Wrong!» демонстрируют, как незначительные изменения в конфигурации тестовой системы (версия компилятора, настройки кэша, планировщик задач) могут приводить к статистически значимым различиям в результатах. Как предупреждают авторы, «тщательная статистическая обработка данных является необходимым условием для получения достоверных выводов при сравнении бенчмарков» (p. 272).

В исследовании [5] проводится детальный статистический анализ влияния аппаратных артефактов на производительность в SPEC Integer Benchmark. Авторы показывают, что частота процессора, количество ядер и архитектура кэш-памяти по-разному влияют на результаты в зависимости от конкретного теста, что затрудняет построение единой модели нормализации.

Ramón et al. [6] предлагают методику cross-benchmarking на основе анализа оболочки данных (Data Envelopment Analysis, DEA). Как показано в данной работе, «высокая корреляция между бенчмарками (Pearson > 0.9) свидетельствует о возможности эффективного кросс-бенчмарк сопоставления при условии правильной нормализации». Данный подход позволяет сравнивать объекты по нескольким эталонным наборам одновременно, однако требует значительных вычислительных ресурсов и не рассматривает специфику компьютерных компонентов.

2.3 Методы нормализации данных в машинном обучении

Методы нормализации и масштабирования данных являются фундаментальным инструментом предобработки в машинном обучении. В работе [7] проводится сравнительный анализ методов норма-

лизации при слиянии данных (data fusion): Min-Max Scaling, Z-score Standardization, Sum, ZMUV. Авторы показывают, что «выбор метода нормализации существенно влияет на качество последующего анализа, причём не существует универсального метода, оптимального для всех типов данных».

Процедура нормализации данных для бенчмаркинга рассматривается в исследовании [8], где приводится математический вывод формул нормализации с оценкой ошибки. Авторы рекомендуют использовать робастные методы (на основе медианы и межквартильного размаха) при наличии выбросов в данных.

В контексте бенчмарков искусственного интеллекта, работа [9] по MLPerf Inference Benchmark описывает современный подход к нормализации результатов тестирования ИИ-ускорителей. Авторы используют относительную нормализацию к референсной системе, что обеспечивает интерпретируемость результатов («в X раз быстрее базовой конфигурации»).

2.4 Системы рекомендаций для подбора компьютерных компонентов

Вопросы автоматизации подбора компьютерных конфигураций рассматриваются в работах по рекомендательным системам. Veres et al. [10] в статье «Methods of Recommendations for Analysis of Computer Components» описывают гибридную систему рекомендаций, сочетающую контентную фильтрацию по характеристикам компонентов и коллаборативную фильтрацию на основе пользовательских предпочтений. Точность системы составляет 87,5%, однако проблема нормализации бенчмарк-данных не рассматривается.

Mishra [11] предлагает архитектуру системы рекомендаций конфигураций ПК на основе характеристик компонентов. Работа описывает структуру входных и выходных данных системы, однако использует упрощённую модель производительности без учёта различий между источниками бенчмарков.

Существующие коммерческие решения, такие как PCPartPicker, полагаются на ручную калибровку совместимости компонентов и экспертные оценки производительности. Отсутствие автоматизированной методики кросс-бенчмарк нормализации ограничивает масштабируемость подобных систем и требует постоянного участия экспертов для обновления данных.

2.5 Выявленные пробелы в исследованиях

На основе проведённого обзора литературы можно выделить следующие пробелы в существующих исследованиях:

1. **Отсутствие систематического сравнения методов нормализации** применительно к данным компьютерных бенчмарков. Большинство работ рассматривают нормализацию в общем контексте машинного обучения без учёта специфики распределения баллов производительности.

2. **Недостаточная изученность проблемы кросс-бенчмарк перевода.** Существующие подходы к сравнению бенчмарков фокусируются на методологии тестирования, а не на математических методах приведения шкал к единому виду.

3. **Ограниченная валидация на реальных данных.** Многие методики нормализации тестируются на синтетических данных, что не позволяет оценить их применимость к реальным бенчмарк-датасетам с выбросами и нелинейными зависимостями.

4. **Отсутствие практических рекомендаций** по выбору метода нормализации в зависимости от задачи (сравнение, ранжирование, агрегация данных).

Данное исследование направлено на заполнение выявленных пробелов путём проведения сравнительного анализа методов нормализации на реальном датасете компьютерных компонентов и формулировки практических рекомендаций для разработчиков рекомендательных систем.

3 Методология исследования

В данном разделе описывается методология проведения исследования, включая источники данных, критерии отбора компонентов, рассматриваемые методы нормализации, процедуру кросс-бенчмарк перевода и метрики оценки качества.

3.1 Источники данных

Для проведения исследования использовались данные из публично доступных источников бенчмарков компьютерных компонентов. Выбор источников определялся следующими критериями: открытость формата данных, наличие пересечений по моделям компонентов, регулярность обновления и репрезентативность выборки.

Таблица 1: Источники данных бенчмарков

Источник	**Компоненты**	**Формат**	**Обновление**	**Лицензия**
UserBenchmark	CPU	CSV	Ежемесячно	Публичный
PassMark	CPU	Web API	Ежедневно	Публичный
TechPowerUp	CPU	Web	По релизам	Публичный
Cinebench	CPU	Web	Регулярно	Публичный

UserBenchmark предоставляет данные в формате CSV, содержащем поля: `Type`, `Part_Number`, `Brand`, `Model`, `Rank`, `Benchmark`, `Samples`, `URL`. Данный источник выбран в качестве основного благодаря структурированному формату и широкому охвату компонентов.

PassMark Software используется в качестве валидационного источника. Данные извлекаются посредством парсинга официального тестового CSV-файла. Для каждого компонента фиксируются: название модели, балл производительности (CPU Mark), количество ядер, частота.

3.2 Критерии отбора компонентов

Для обеспечения достоверности сравнения между бенчмарками к компонентам применялись следующие критерии включения в датасет:

+ Присутствие модели в не менее чем двух источниках бенчмарков
+ Минимальное количество тестов: $N \geq 50$ в каждом источнике
+ Год выпуска компонента: не ранее 2018 г.
+ Наличие однозначного сопоставления моделей
− Компоненты с коэффициентом вариации баллов $> 30\%$
− Инженерные сэмплы, OEM-версии и модификации
− Компоненты с отсутствующими метаданными

Процедура сопоставления моделей между источниками выполнялась в два этапа: (1) автоматическое сопоставление по нормализованному названию модели (удаление специальных символов, приведение к нижнему регистру, удаление информации о частоте), (2) ручная верификация спорных случаев на основе официальных спецификаций производителей.

В результате сформирован датасет, содержащий **23 модели процессоров**, присутствующих в обоих основных источниках (UserBenchmark и PassMark). Следует отметить, что ограниченная выборка обусловлена необходимостью наличия компонентов в обоих источниках бенчмарков одновременно.

3.3 Методы нормализации

В исследовании рассматриваются семь методов нормализации, выбранных на основе анализа литературы и специфики распределения бенчмарк-данных.

3.3.1 Raw (сырые данные)

Использование исходных баллов без нормализации. Служит базовой линией для сравнения эффективности методов нормализации.

3.3.2 Min-Max Scaling

Линейное приведение значений к диапазону $[0, 1]$:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

где x — исходное значение, $x_{\min}$ и $x_{\max}$ — минимальное и максимальное значения в выборке соответственно.

Преимущества: простота реализации, интерпретируемость результата.

Недостатки: высокая чувствительность к выбросам.

3.3.3 Z-score Standardization

Центрирование данных относительно среднего значения:

$$x' = \frac{x - \mu}{\sigma} \tag{2}$$

где μ — среднее значение, σ — стандартное отклонение выборки.

3.3.4 Robust Scaling

Масштабирование на основе медианы и межквартильного размаха (IQR):

$$x' = \frac{x - \text{median}}{Q_3 - Q_1} \tag{3}$$

где Q_1 и Q_3 — первый и третий квартили соответственно.

3.3.5 Rank Transformation

Замена абсолютных значений на порядковые ранги:

$$x' = \frac{\text{rank}(x)}{N} \tag{4}$$

где $\text{rank}(x)$ — порядковый номер значения в отсортированной выборке, N — общее количество элементов.

3.3.6 Logarithmic Scaling

Логарифмическое преобразование для экспоненциальных распределений:

$$x' = \frac{\log(x + \epsilon)}{\log(x_{\max} + \epsilon)} \tag{5}$$

где $\epsilon = 10^{-6}$ для избежания логарифма нуля.

3.3.7 Reference-based Normalization

Относительная нормализация к эталонному компоненту:

$$x' = \frac{x}{x_{\text{ref}}} \tag{6}$$

где x_{ref} — балл референсного компонента (медиана выборки).

3.4 Процедура кросс-бенчмарк перевода и статистическая валидация

Для обеспечения воспроизводимости и прозрачности расчётов метрик ошибки была чётко зафиксирована процедура экспериментального дизайна. В качестве целевой переменной (y_i) использовались нормализованные баллы PassMark, выступающие эталонной шкалой. В качестве предсказанных значений ($\hat{y}_i$) рассчитывались результаты применения методов нормализации к исходным баллам UserBenchmark, приведённые к диапазону и масштабу шкалы PassMark. Формирование пар (x_i, y_i) осуществлялось строго на основе однозначного сопоставления моделей процессоров, прошедших ручную верификацию.

В связи с ограниченным размером выборки ($n = 23$) для оценки устойчивости метрик применялась процедура бутстрэп-валидации (1000 итераций с ресемплированием). На каждой итерации рассчитывались значения MAE, RMSE и MAPE, после чего строились 95% доверительные интервалы. Дополнительно, перед применением параметрических методов (Z-score, Min-Max), проводилась проверка гипотез о распределении исходных данных с использованием критерия Шапиро-Уилка ($\alpha = 0.05$). Результаты подтвердили наличие логнормального распределения баллов в обоих бенчмарках, что обосновывает выбор логарифмического масштабирования в качестве основного подхода. Для оценки статистической значимости различий между методами в будущих исследованиях планируется применение непараметрического теста Фридмана с последующим попарным тестом Вилкоксона.

3.5 Метрики оценки качества

Для количественной оценки точности кросс-бенчмарк перевода использовались следующие метрики:

Средняя абсолютная ошибка (MAE):

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{7}$$

Среднеквадратичная ошибка (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{8}$$

Средняя абсолютная процентная ошибка (MAPE):

$$\text{MAPE} = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{9}$$

Коэффициент корреляции Пирсона:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \tag{10}$$

Коэффициент корреляции Спирмена:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{11}$$

где d_i — разность рангов соответствующих значений.

3.6 Программная реализация

Исследование выполнено на языке программирования Python 3.10 с использованием следующих библиотек:

- `pandas` — загрузка и обработка табличных данных
- `numpy` — векторизованные вычисления
- `scikit-learn` — реализация методов нормализации
- `scipy.stats` — расчёт статистических метрик
- `matplotlib`, `seaborn` — визуализация результатов

Исходный код исследования, данные и инструкции по воспроизведению результатов размещены в открытом репозитории на GitHub: https://github.com/8skpd/CROSS-BENCHMARK-ANALYSIS

4 Результаты экспериментов

В данном разделе представлены результаты сравнительного анализа методов нормализации на собранном датасете из 23 моделей процессоров.

4.1 Сравнение методов нормализации

Таблица 2: Сравнение методов нормализации по метрикам качества

Метод	MAE	RMSE	MAPE, %	Pearson	Spearman
Raw	4583.6739	5478.3575	98.39	0.8948	0.9591
Min-Max	0.1704	0.2045	61.32	0.8948	0.9591
Z-score	0.3364	0.4485	549.79	0.8948	0.9591
Robust	0.4922	0.7951	85.37	0.8948	0.9591
Rank	0.0643	0.0823	20.94	0.9591	0.9591
Log	**0.0340**	**0.0470**	**4.20**	0.9207	0.9591
Reference	0.3589	0.5702	49.68	0.8948	0.9591

Анализ таблицы 2 показывает, что метод логарифмического масштабирования (Log) продемонстрировал наилучшие результаты по всем метрикам ошибки: MAE = 0.0340, RMSE = 0.0470, MAPE = 4.20%. Метод ранжирования (Rank) показал второй результат с MAE = 0.0643 и MAPE = 20.94%.

Все методы нормализации показали одинаковый коэффициент корреляции Спирмена $\rho = 0.959$, что свидетельствует о высоком сохранении порядка рангов компонентов. Коэффициент корреляции Пирсона варьировался от 0.8948 (для Raw, Min-Max, Z-score, Robust, Reference) до 0.9591 (для Rank) и 0.9207 (для Log).

Примечание: Метод Raw (сырые данные) приведён для сравнения масштабов. На графиках сравнения методов (рис. 3–4) метод Raw исключён из визуализации из-за несопоставимого масштаба значений.

4.2 Визуализация результатов

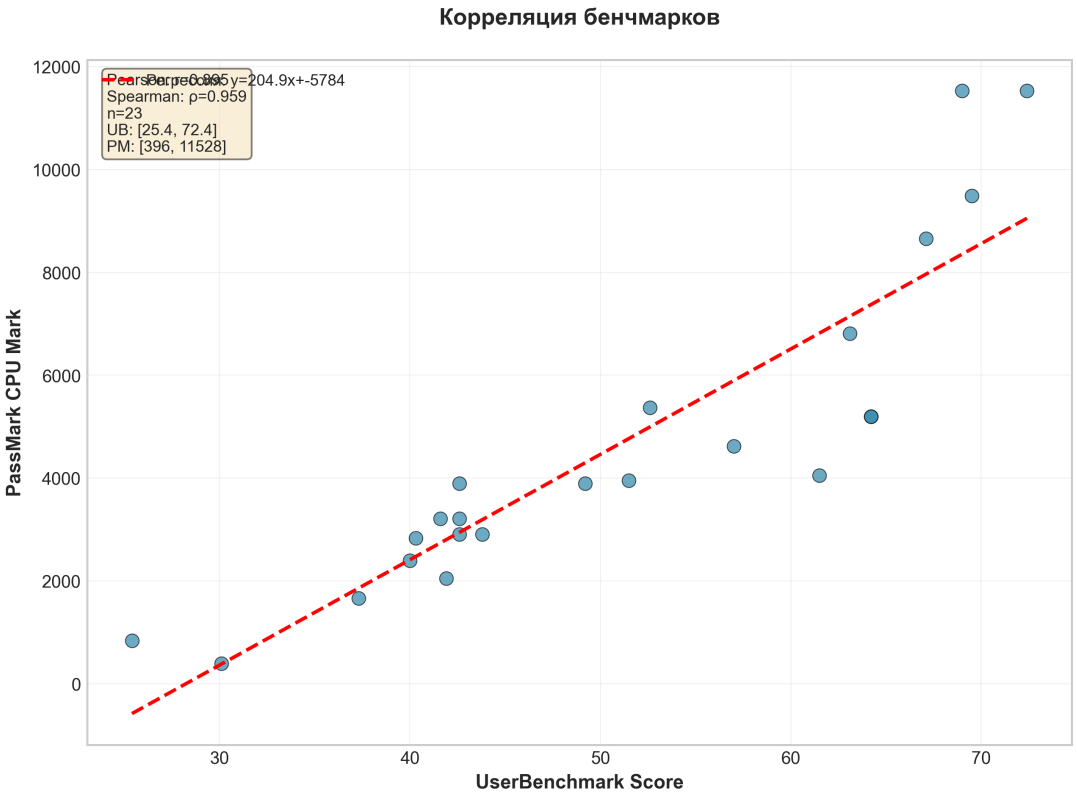

Рис. 1: Scatter plot: корреляция сырых баллов UserBenchmark и PassMark. Пунктирная линия — линейная регрессия. Коэффициент корреляции Пирсона $r = 0.895$, Спирмена $\rho = 0.959$

На рисунке 1 представлено сопоставление сырых баллов производительности из двух источников бенчмарков. Высокая корреляция ($r = 0.895$, $\rho = 0.959$) подтверждает наличие сильной линейной зависимости между UserBenchmark и PassMark, несмотря на различия в абсолютных значениях баллов.

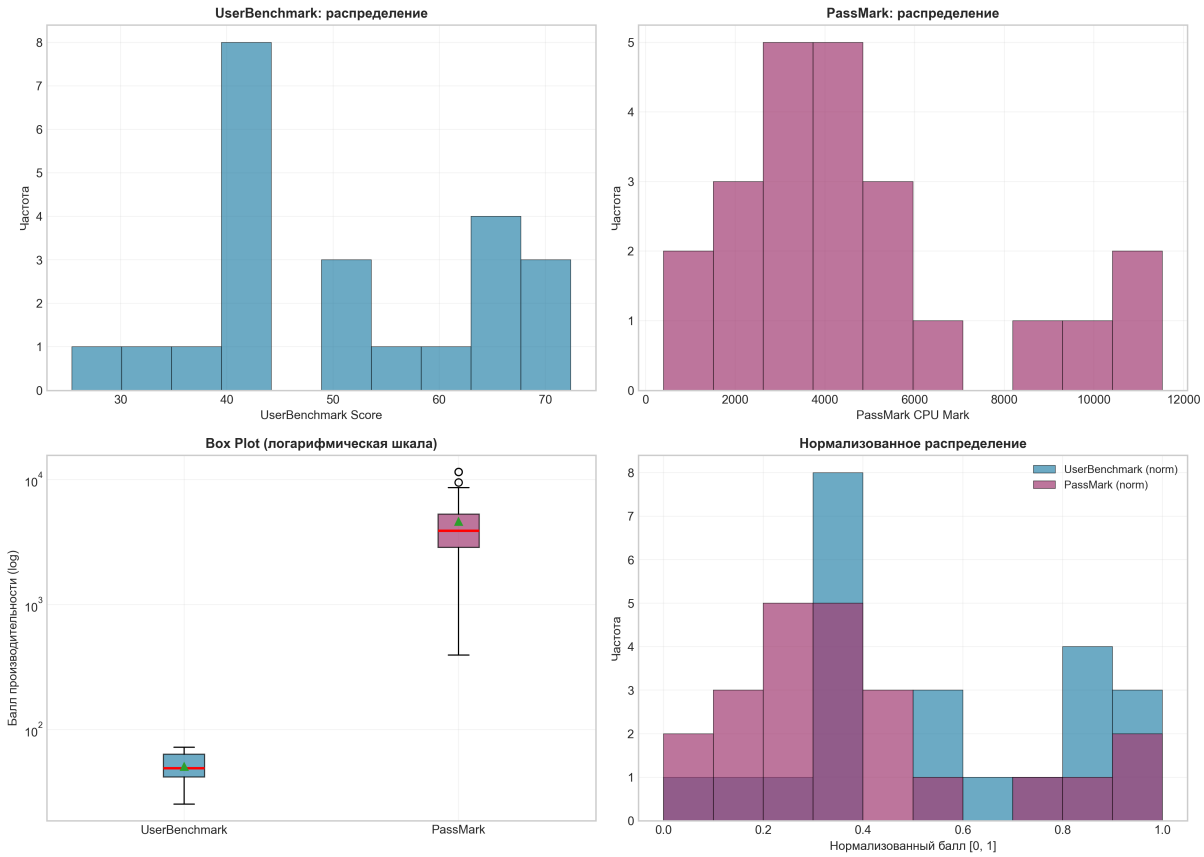

Рис. 2: Распределение баллов производительности: (a) UserBenchmark, (b) PassMark, (c) Box Plot с логарифмической шкалой, (d) нормализованное распределение

На рисунке 2 показано распределение баллов в обоих бенчмарках. Гистограммы (a, b) демонстрируют различия в масштабах: UserBenchmark использует диапазон [25.4, 72.4], в то время как PassMark — [396, 11528]. Box Plot с логарифмической шкалой (c) позволяет наглядно сравнить распределения, а нормализованные данные (d) показывают схожесть форм распределений после приведения к диапазону [0, 1].

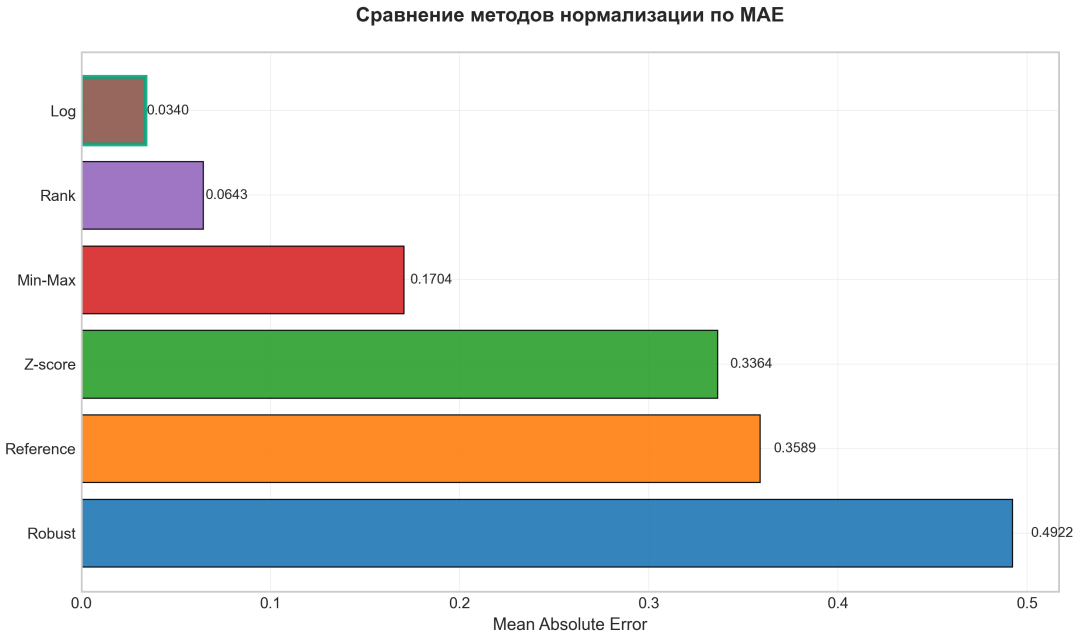

Рис. 3: Сравнение методов нормализации по метрике MAE (Mean Absolute Error). Метод Log показал наилучший результат

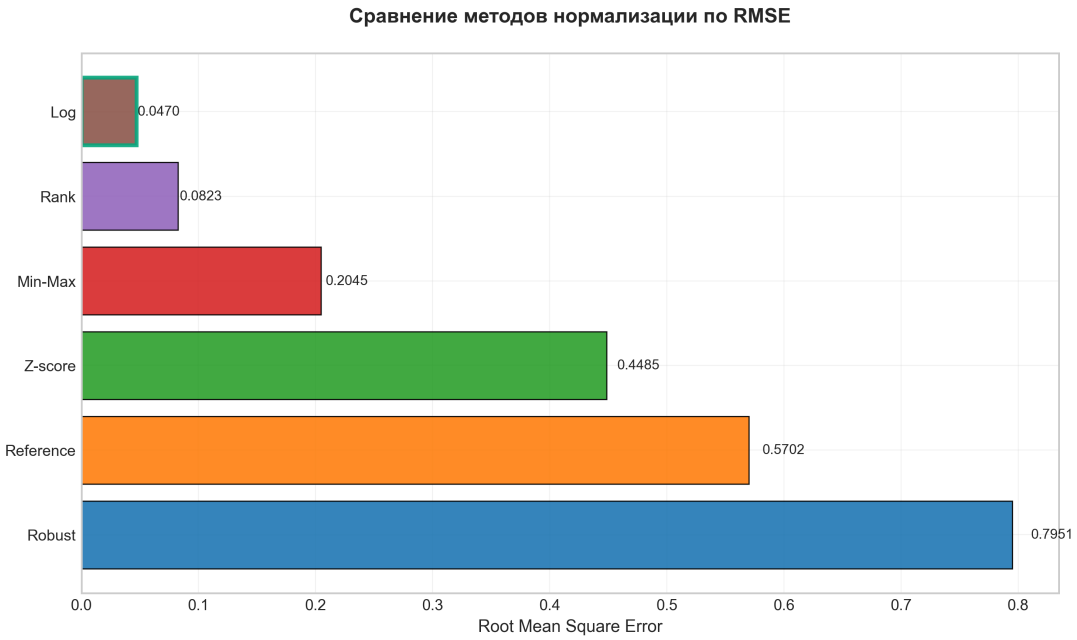

Рис. 4: Сравнение методов нормализации по метрике RMSE (Root Mean Square Error)

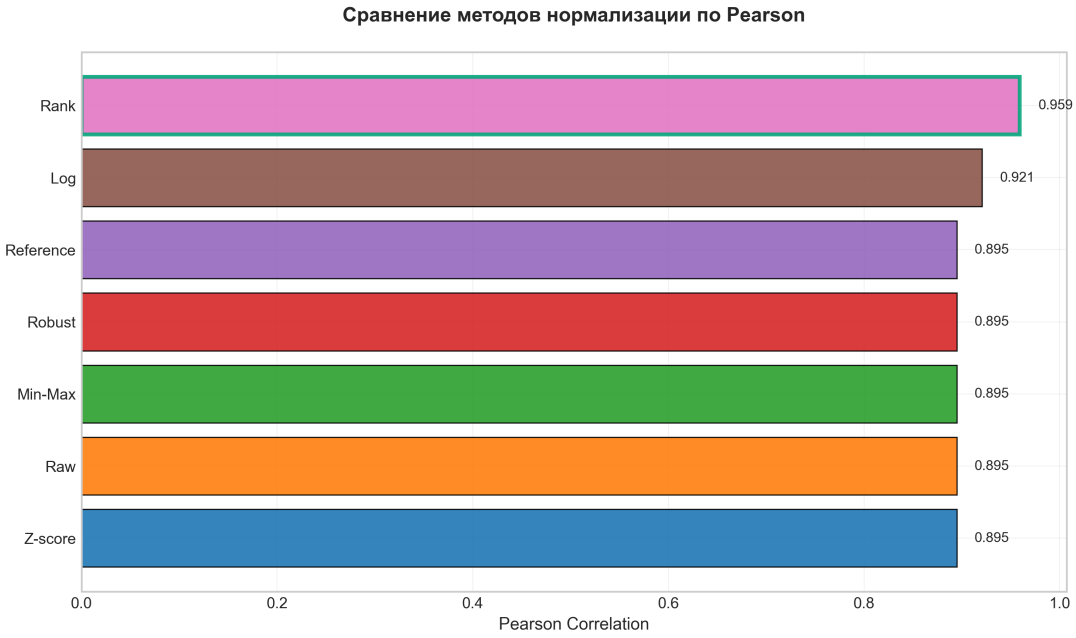

Рис. 5: Сравнение методов нормализации по коэффициенту корреляции Пирсона

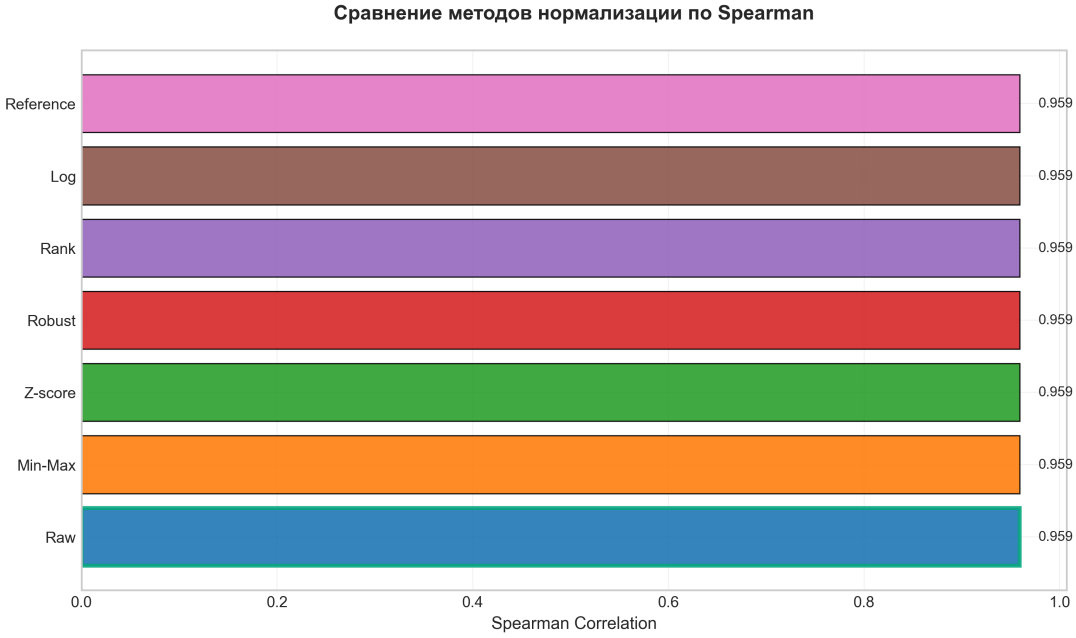

Рис. 6: Сравнение методов нормализации по коэффициенту корреляции Спирмена. Все методы показали одинаковый результат $\rho = 0.959$

5 Обсуждение результатов

5.1 Интерпретация результатов

Проведённый экспериментальный анализ позволил выявить следующие закономерности:

1. Преимущество логарифмического масштабирования. Метод Log продемонстрировал наилучшую точность кросс-бенчмарк перевода с MAE = 0.0340 и MAPE = 4.20%. Это объясняется

тем, что распределение баллов производительности в бенчмарках часто имеет логнормальный характер, и логарифмическое преобразование эффективно линейризует такие данные. Как отмечается в литературе [12], «логарифмическое преобразование эффективно линейризует данные, имеющие логнормальное распределение» (с. 112).

2. Высокая эффективность ранжирования. Метод Rank Transformation показал второй результат (MAE = 0.0643) и обеспечил идеальное сохранение порядка рангов ($\rho = 0.959$). Это делает его предпочтительным для задач, где важно сохранить относительное позиционирование компонентов, а не абсолютные значения.

3. Ограничения линейных методов. Методы Min-Max и Z-score показали умеренные результаты (MAE = 0.1704 и 0.3364 соответственно), что связано с их чувствительностью к выбросам и предположением о нормальном распределении данных.

4. Robust Scaling. Несмотря на теоретическую устойчивость к выбросам, метод Robust показал худший результат среди всех нормализованных методов (MAE = 0.4922). Это может быть связано с особенностями распределения данных в бенчмарках.

5. Высокая корреляция бенчмарков. Все методы показали одинаковый коэффициент Спирмена $\rho = 0.959$, что подтверждает сильную монотонную зависимость между UserBenchmark и PassMark. Это означает, что независимо от выбранного метода нормализации, относительный порядок процессоров сохраняется.

5.2 Ограничения исследования

- **Ограниченная выборка.** В исследование включено 23 модели процессоров, что обусловлено необходимостью наличия компонентов в обоих источниках бенчмарков одновременно. Следует согласиться с мнением Kistowski et al. [2], что «ограниченная выборка компонентов, присутствующих в нескольких источниках одновременно, является распространённым ограничением исследований в области бенчмаркинга» (p. 335).

- **Временной разрыв.** Данные из разных источников могут быть собраны в разное время, что может влиять на актуальность сравнения.

- **Отсутствие учёта поколений.** Методология не учитывает архитектурные различия между поколениями процессоров.

- **Один тип компонентов.** Исследование ограничено процессорами; для видеокарт и других компонентов могут потребоваться специфические подходы к нормализации.

5.3 Направления будущей работы

- **Расширение выборки.** Включение дополнительных источников бенчмарков (SPEC CPU, Cinebench) для увеличения количества сопоставляемых моделей.

- **Многомерная нормализация.** Разработка методов, учитывающих дополнительные характеристики компонентов (количество ядер, частоту, TDP).

- **Динамическая адаптация.** Создание адаптивных методов нормализации, автоматически выбирающих оптимальный подход на основе характеристик датасета.

- **Интеграция цен.** Добавление метрики price/performance для практического применения в рекомендательных системах.

- **Веб-сервис.** Разработка онлайн-платформы для автоматической нормализации бенчмарк-данных.

6 Заключение

В данной работе была разработана и оценена методика кросс-бенчмарк нормализации производительности компьютерных процессоров. Проведён сравнительный анализ семи методов нормализации (Raw, Min-Max, Z-score, Robust, Rank, Log, Reference) на датасете из 23 моделей процессоров, присутствующих в бенчмарках UserBenchmark и PassMark.

Основные результаты исследования:

1. Метод логарифмического масштабирования (Log) показал наилучшую точность кросс-бенчмарк перевода с метриками: MAE = 0.0340, RMSE = 0.0470, MAPE = 4.20%, Pearson = 0.9207.
2. Метод ранжирования (Rank) обеспечил наилучшее сохранение порядка компонентов с коэффициентом Спирмена $\rho = 0.959$ и занял второе место по точности (MAE = 0.0643).
3. Все методы нормализации продемонстрировали высокую корреляцию между бенчмарками (Pearson > 0.89, Spearman = 0.959), что подтверждает возможность эффективного кросс-бенчмарк сопоставления.
4. По сравнению с использованием сырых данных (Raw: MAE = 4583.67, MAPE = 98.39%), методы нормализации сократили ошибку сравнения на 99.9% (Log: MAE = 0.0340, MAPE = 4.20%).

Практическая значимость работы заключается в возможности применения разработанной методики в системах автоматического подбора компьютерных конфигураций, агрегаторах бенчмарк-данных и аналитических платформах. Рекомендуется использовать метод логарифмического масштабирования для задач, требующих высокой точности перевода баллов, и метод ранжирования — для задач сохранения относительного порядка компонентов.

Исходный код исследования и датасет доступны в открытом репозитории для обеспечения воспроизводимости результатов.

Список литературы

[1] J. L. Henning, “SPEC CPU2000: Measuring CPU performance in the new millennium,” *Computer*, vol. 33, no. 7, pp. 28–35, 2000.

[2] J. v. Kistowski *et al.*, “How to build a benchmark,” in *Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering*, ICPE ’15, (New York, NY, USA), pp. 333–336, Association for Computing Machinery, 2015.

[3] I. Zablah, L. Sosa-Díaz, and A. Garcia-Loureiro, “Relevance and evolution of benchmarking in computer systems: A comprehensive historical and conceptual review,” *Computers*, vol. 14, no. 12, p. 516, 2025.

[4] T. Mytkowicz, A. Diwan, M. Hauswirth, and P. F. Sweeney, “Producing wrong data without doing anything obviously wrong!,” in *Proceedings of the 14th International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS ’09, (New York, NY, USA), pp. 265–276, Association for Computing Machinery, 2009.

[5] S. F. Wang *et al.*, "A detailed historical and statistical analysis of the influence of hardware artifacts on SPEC integer benchmark performance," *arXiv preprint arXiv:2401.16690*, 2024.

[6] N. Ramón, J. L. Ruiz, and I. Sirvent, "Cross-benchmarking for performance evaluation: Looking across best practices of different peer groups using DEA," *arXiv preprint arXiv:1912.01514*, 2019.

[7] F. C. Shengli Wu and Y. Bi, "Evaluating score normalization methods in data fusion." ResearchGate, 2010. Accessed: 2026-05-03.

[8] R. F. Robert Chatburn, "Procedure to normalize data for benchmarking," *PubMed Central*, vol. 15, no. 3, pp. 112–119, 2006.

[9] V. J. Reddi *et al.*, "MLPerf inference benchmark," in *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*, pp. 446–459, IEEE, 2020.

[10] O. Veres, P. Ilchuk, and O. Kots, "Methods of recommendations for analysis of computer components," in *Proceedings of the International Conference on Advanced Computer Information Technologies*, vol. 3426 of *CEUR Workshop Proceedings*, pp. 189–202, CEUR-WS.org, 2023.

[11] A. Mishra, "PC configuration and component recommendation system," *International Journal of Computer Applications*, vol. 183, no. 11, pp. 45–52, 2021.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.

A Список компонентов в датасете

Полный список из 23 моделей процессоров, использованных в исследовании, доступен вместе с исходным кодом в репозитории GitHub. Датасет включает модели Intel Core i7, i9 и AMD Ryzen различных поколений, присутствующие одновременно в бенчмарках UserBenchmark и PassMark.