УДК 005.94
DOI: 10.26102/2310-6018/2026.58.7.014

Вычислительно эффективная коррекция раскладки ENG→RU в высоконагруженных информационно-поисковых системах электронной коммерции

Ф.В. Краснов✉

ООО «Ви.Тех», Ковров, Российская Федерация

Резюме. Современные поисковые системы электронной коммерции функционируют в условиях высокой пользовательской активности и необходимости обработки коротких, структурно насыщенных запросов, содержащих технические обозначения и артикулы. Особую сложность представляет сегмент «сделай сам» (DIY), где значительная доля запросов включает буквенно-цифровые токены («M8x50», «LED 12W», «SSD 1TB»). Одним из существенных источников деградации качества поиска являются ошибки раскладки клавиатуры (ENG→RU), при которых латинские символы интерпретируются как кириллические, формируя нераспознаваемые комбинации (например, «ыыВ 1ЕИ» вместо «SSD 1TB»). Такие искажения нарушают лексическую и семантическую целостность запроса, приводят к снижению релевантности выдачи и негативно отражаются на конверсии. В работе предложен вычислительно эффективный подход к коррекции раскладки клавиатуры, ориентированный на эксплуатацию в высоконагруженных информационно-поисковых системах с ограниченными ресурсами. Метод объединяет детерминированные таблицы сопоставления клавиш, контекстное сопоставление токенов и легковесные модели машинного обучения, обеспечивая баланс точности и производительности. Экспериментальная оценка на доменно-специфичных наборах данных показала повышение точности поиска на 25–30 % по сравнению с базовыми орфографическими методами при среднем времени отклика менее 10 мс на запрос. Полученные результаты подтверждают масштабируемость и промышленную применимость предложенного решения в серверных и облачных IR-платформах.

Ключевые слова: информационный поиск, электронная коммерция, коррекция раскладки клавиатуры, высоконагруженные системы, вычислительная эффективность, технические поисковые запросы.

Для цитирования: Краснов Ф.В. Вычислительно эффективная коррекция раскладки ENG→RU в высоконагруженных информационно-поисковых системах электронной коммерции. *Моделирование, оптимизация и информационные технологии.* 2026;14(7). URL: https://moitvivt.ru/ru/journal/article?id=2256 DOI: 10.26102/2310-6018/2026.58.7.014

Computationally efficient ENG→RU keyboard layout correction in high-load e-commerce information retrieval systems

F.V. Krasnov✉

Vi.Tech LLC, Kovrov, the Russian Federation

Abstract. Modern e-commerce search systems operate under conditions of high user activity and the need to process short, structurally dense queries containing technical specifications and product identifiers. The "do-it-yourself" (DIY) segment presents particular challenges, as a substantial proportion of queries include alphanumeric tokens (e.g., "M8x50", "LED 12W", "SSD 1TB"). A significant source of retrieval quality degradation arises from keyboard layout errors (ENG→RU), in which Latin characters are interpreted as Cyrillic ones, producing unrecognized combinations (e.g., "ыыВ 1ЕИ" instead of "SSD 1TB"). Such distortions disrupt the lexical and semantic integrity of queries, reduce retrieval relevance, and negatively affect conversion rates. This study proposes a

computationally efficient approach to ENG→RU keyboard layout correction designed for deployment in high-load information retrieval systems operating under limited computational resources. The method integrates deterministic key-mapping tables, contextual token matching, and lightweight machine learning models to achieve a balance between accuracy and performance. Experimental evaluation on domain-specific datasets demonstrates a 25–30 % improvement in retrieval accuracy compared to baseline spelling-correction methods, while maintaining an average response latency below 10 ms per query. The results confirm the scalability and industrial applicability of the proposed solution in server-side and cloud-based IR platforms.

Keywords: information retrieval, e-commerce, keyboard layout correction, high-load systems, computational efficiency, technical search queries.

For citation: Krasnov F.V. Computationally efficient ENG→RU keyboard layout correction in high-load e-commerce information retrieval systems. *Modeling, Optimization and Information Technology.* 2026;14(7). (In Russ.). URL: https://moitvivt.ru/ru/journal/article?id=2256 DOI: 10.26102/2310-6018/2026.58.7.014

Введение

Платформы электронной коммерции, особенно в сегменте «сделай сам», часто обрабатывают короткие запросы с техническими обозначениями – «bolt M8x50», «drill 18V» и т. п. Ошибки раскладки клавиатуры (ENG→RU) приводят к тому, что латинские символы интерпретируются как кириллические, делая запросы нераспознаваемыми для поискового индекса. Это снижает релевантность и конверсию до 40–50 %, согласно аналитике крупных маркетплейсов.

Цель исследования – разработка и оценка вычислительно эффективных алгоритмов коррекции раскладки для высоконагруженных IR-систем. В отличие от орфографических методов, ориентированных на естественный язык, подход нацелен на восстановление коротких технических токенов. Работа объединяет контекстное сопоставление, эвристические правила и легкие модели машинного обучения, формируя основу для масштабируемой обработки запросов в реальном времени.

Обзор литературы. Проблема обработки шумных запросов активно исследуется в контексте поисковых систем и машинного перевода. Ранние работы по статистической транслитерации [1] и коррекции имен собственных [2] улучшали поиск в многоязычных сценариях, но не учитывали визуальные ошибки клавиатурной раскладки. Многоязычные интерфейсы и модели ввода [3] рассматривали транслитерацию для индийских языков, однако их применение к ENG→RU ограничено.

Для русского языка проводились исследования по коррекции орфографии [4, 5], устойчивые к шуму, но не адаптированные к коротким токенам и техническим обозначениям. Нейронные подходы [6, 7] обеспечивают высокую точность, но требуют значительных вычислительных ресурсов, что делает их непрактичными в условиях реального времени. Работа [8], ориентированная на моделирование произношения для улучшения орфографической коррекции, демонстрирует эффективность для английского языка, но ее фонетическая направленность ограничивает применимость к визуальным ошибкам раскладки типа ENG→RU. В электронной коммерции подтверждено влияние корректировки запросов на рост конверсии [9], однако специфика коротких технических токенов в сегменте DIY остается недостаточно изученной.

Таким образом, существующие решения, включая Yandex Speller [5], не масштабируются под серверные сценарии с высокой нагрузкой. Расширение запросов через векторные представления слов [10] улучшает релевантность, но редко учитывает ограничения ресурсов.

Для коротких технических токенов коррекция орфографии малоприменима, так как транслитерированные ошибки не соответствуют лингвистическим паттернам [11].

Алгоритмы раскладочной коррекции, напротив, детерминированы и эффективно восстанавливают исходный токен на основе фиксированных таблиц сопоставления [12]. Это определяет актуальность легковесных решений, устойчивых к шуму и пригодных для промышленного внедрения.

Материалы и методы

Предложенная методика базируется на многоуровневом подходе к автоматической коррекции раскладки ввода в рамках конвейеров (pipelines) информационного поиска. Первичный этап обработки предполагает идентификацию ошибочных вхождений – процедуру детекции токенов, подлежащих исправлению. Реализация данного этапа осуществляется посредством компактной нейросетевой модели, прошедшей процедуру обучения на специализированном массиве доменных данных из сегмента DIY (Do-It-Yourself).

Средний словарь каталога насчитывает 1–3 млн токенов, а с учетом длинного хвоста низкочастотных комбинаций их число в запросах достигает 10 млн в сутки. Распределение отклоняется от закона Ципфа, что требует динамических фильтров и отказа от статичных словарей.

Для формализации процесса детекции вводится метрика аномальности токена w_i, основанная на логарифмическом правдоподобии символьных n-грамм для выбранного языка:

$$P(L|w_i) = \frac{1}{|w_i|}\sum log\, P\left(c_j|c_{j-k}, \dots, c_{j-1}, L\right), \quad (1)$$

где c_j – символы токена, а L – предполагаемый язык (RU/EN). Оценка вероятностей P производится на основе метода максимального правдоподобия (MLE) по репрезентативному корпусу очищенных поисковых запросов и названий товаров. Для обеспечения вычислительной эффективности в условиях высокой нагрузки используется порядок n-грамм $k = 3$ (триграммы). Данный выбор обусловлен балансом между точностью детекции и объемом оперативной памяти, необходимым для хранения предварительно вычисленных таблиц вероятностей. Решение о классификации токена как кандидата на исправление принимается на основе следующего условия $is\_candidate(w_i) = 1 if P(RU|w_i) < \tau(|w_i|) else 0$, где $\tau(|w_i|)$ – адаптивный порог аномальности, значение которого устанавливается в зависимости от длины токена. Использование функциональной зависимости порога τ от длины $|w_i|$ необходимо для компенсации высокой естественной вариативности коротких токенов (например, артикулов или технических аббревиатур), где статистическая предсказуемость символьных цепочек ниже, чем в полнотекстовых словах. Финальная калибровка параметров τ выполняется эмпирически в ходе оффлайн-валидации для достижения компромисса между полнотой детекции и минимизацией доли ложных срабатываний (FPR).

Ключевой элемент методологии – разграничение задач обнаружения и исправления транслитерации:

1. Обнаружение фокусируется на токенах, возникших из-за ошибочной раскладки ENG→RU, и использует паттерны несоответствия кириллических и латинских символов.

2. Исправление выполняется по фиксированным маппингам клавиш $f: C_{err} \rightarrow C_{corr}$, восстанавливая исходный смысл токена.

Контекст запроса используется для минимизации ложных срабатываний. Оценка финального исправления $\hat{w}_i$ для токена w_i в контексте соседних слов $Q \setminus \{w_i\}$ вычисляется через максимизацию апостериорной вероятности:

$$\hat{w}_i = \text{argmax}_{w' \in \{w_i, f(w_i)\}}[\alpha \cdot \text{Sim}(E(w'), E_{cat}) + (1-\alpha) \cdot P(w'|Q \setminus \{w_i\})], \quad (2)$$

где $E(w')$ – векторное представление токена, E_{cat} – центроид векторов категории, α – весовой коэффициент значимости семантики над языковой моделью. Центроид векторов категории E_{cat} рассчитывается как среднее арифметическое эмбеддингов наименований N наиболее релевантных или частотных товаров в соответствующем разделе каталога: $E_{cat} = \frac{1}{N}\sum E(p_j)$, где $E(p_j)$ – векторное представление j-го товара в категории. Использование центроида позволяет оценить семантическую близость исправленного токена к тематике товаров, которые пользователь рассчитывает найти в выбранном сегменте. Весовой коэффициент α измеряется в диапазоне [0, 1]. При $\alpha \to 1$ система отдает приоритет семантической согласованности токена с товарной категорией (что критично для технических обозначений в DIY), а при $\alpha \to 0$ решение принимается преимущественно на основе вероятности появления слова в естественном языке. Оптимальное значение α определяется в процессе оффлайн-калибровки на исторических логах поисковых сессий путем максимизации целевой функции качества ранжирования при заданных ограничениях на долю ложных исправлений.

Контекстное сопоставление задействует векторные представления и анализ n-грамм для оценки семантической согласованности с каталогом товаров. Особое значение это приобретает в сегменте DIY, где технические обозначения (например, М8х50) могут совпадать по форме с ошибками (Ь8ч50). Для оценки эффективности исправления всего запроса Q используется функция потерь, минимизирующая ожидаемое расстояние до эталонного поискового интента:

$$L(Q, Q_{corr}) = 1 - \frac{\sum_{j=1}^{k} \text{rel}(d_j) \cdot \text{IDCG}_j^{-1}}{max\text{DCG}_k}, \quad (3)$$

где $\text{rel}(d_j)$ – релевантность документов, возвращенных по исправленному запросу.

Метод поддерживает частичную коррекцию: в запросах типа «купить LED лампу» корректируется лишь один токен. Это обеспечивает адаптивность в реальных пользовательских сценариях. Итеративный цикл – детекция → исправление → валидация – обеспечивает баланс точности и производительности при минимальных вычислительных затратах.

Дополнительно к описанному конвейеру, методика включает этап адаптивной калибровки пороговых параметров и весовых коэффициентов, обеспечивающий устойчивость системы в условиях изменяющегося пользовательского поведения и ассортимента каталога. В частности, порог аномальности токена и коэффициент значимости семантической компоненты оптимизируются не статически, а на основе скользящего окна поисковых сессий. Для этого используется процедура оффлайн-валидации на логах пользовательских взаимодействий (клики, добавления в корзину, конверсии), где целевая функция формулируется как компромисс между снижением числа ложных исправлений и ростом метрик ранжирования. Оптимизация параметров может быть представлена в виде задачи максимизации взвешенной функции качества:

$$\theta^* = argmax_\theta\big(\lambda \cdot NDCG@k(\theta) - (1-\lambda) \cdot FPR(\theta)\big), \quad (4)$$

где θ – вектор настраиваемых параметров (порог аномальности, вес семантики), $\lambda \in [0,1]$ – коэффициент баланса, FPR – доля ложных срабатываний.

Параметры модели подбираются с применением стратифицированной выборки по частотным кластерам запросов, что позволяет избежать деградации качества на длинном хвосте. Такой подход обеспечивает переносимость методики между различными доменами электронной коммерции без необходимости полной переобучаемости модели.

Важным компонентом является также оптимизация вычислительного исполнения методики в высоконагруженной среде. Для обеспечения требований по латентности применяется совокупность инженерных решений: кэширование результатов детекции для высокочастотных токенов, использование компактных представлений символьных n-грамм, а также предварительная материализация кандидатов исправления для наиболее вероятных клавиатурных преобразований. Формально ожидаемое время обработки запроса может быть оценено как:

$$T_{query} = \sum\left(p_i \cdot T_{corr} + (1 - p_i) \cdot T_{pass}\right), \quad (5)$$

где N – число токенов в запросе, p_i – вероятность необходимости коррекции i-го токена, T_{corr} – время полной обработки (детекция + исправление + валидация), T_{pass} – время пропуска без изменений.

Конвейер реализуется в виде асинхронных стадий с возможностью раннего выхода при достижении достаточной уверенности в корректности исходного токена, что минимизирует среднее время обработки. Дополнительно применяется батчинг и векторизация операций на уровне CPU-инструкций, обеспечивая обработку пиковых нагрузок без существенного увеличения инфраструктурных затрат и сохраняя предсказуемость времени отклика.

Результаты

Для проверки эффективности предложенных алгоритмов сформирован корпус из реальных и синтетических поисковых запросов сегмента DIY. Основу составили анонимизированные логи поисковых сессий и данные открытых каталогов, совокупный словарь которых составил порядка 2,5 млн уникальных токенов. Итоговый корпус содержит около 10 млн записей, распределенных по ключевым товарным категориям: «Инструменты» (около 35 %), «Электроника» (25 %), «Стройматериалы» (20 %) и прочие.

Аугментация выполнялась по фиксированным маппингам ENG→RU («S»→«Ы», «D»→«В»), с варьированием количества искаженных токенов. Выбор детерминированного подхода к генерации синтетического шума обоснован спецификой исследуемой проблемы: в отличие от классических опечаток, возникающих из-за фонетического или когнитивного сходства, ошибки непереключенной раскладки при вводе технических артикулов строго привязаны к физической геометрии клавиатуры. Следовательно, фиксированный аппаратный маппинг является адекватным и репрезентативным приближением реального паттерна пользовательских ошибок в сегменте.

Распределение по числу ошибок демонстрирует выраженный «длинный хвост»: большинство запросов корректны, но редкие сильно искаженные примеры критически влияют на качество поиска.

Таблица 1 демонстрирует распределение классов по количеству ошибок раскладки в сформированном экспериментальном корпусе. Следует отметить, что данная выборка не является репрезентативным срезом общего поискового трафика, а представляет собой целевой набор данных для оценки модуля коррекции. Включение 20,6 % корректных запросов (класс 0) обусловлено необходимостью жесткого контроля метрики ложных срабатываний (False Positive Rate). Поскольку в промышленной эксплуатации алгоритм применяется к подмножеству запросов, предварительно отобранных триггерами как потенциально ошибочные, выбранное распределение соответствует реальной операционной ситуации: по результатам внутренней разметки, порядка 20 % запросов, классифицированных системой как подозрительные, в

действительности не содержат ошибок раскладки. Таким образом, структура датасета позволяет объективно оценить не только полноту исправления, но и надежность работы системы в условиях отсутствия искажений.

Таблица 1 – Распределение классов ошибок
Table 1 – Distribution of error classes

Количество ошибок в поисковом запросе	Доля в датасете
0	0,2063
1	0,0312
2	0,0727
3	0,1173
4	0,1523
5	0,1674
6	0,1463
7	0,0844
8	0,0222

Корпус разделен на обучающую (70 %), валидационную (15 %) и тестовую (15 %) выборки.

Оценка проводилась в условиях, имитирующих реальные IR-системы: нагрузка до 1000 запросов/сек на облачных узлах с ограниченными ресурсами. Сравнивались три подхода:

1) стандартная орфографическая коррекция (базовый уровень);
2) эвристический алгоритм раскладочной коррекции;
3) гибридная модель (эвристика + ML).

Метрики включали точность, полноту, F1-меру, а также пропускную способность и среднюю задержку отклика.

Гибридная модель показала рост релевантности на 25–30 % по сравнению с базовым методом. Среднее время отклика – 8–9 мс (95-й перцентиль < 17 мс), что соответствует требованиям SLA (Service Level Agreement) для коммерческих платформ. Пропускная способность достигала 115 запросов/сек при нагрузке без сбоев.

Использование легковесной модели fastText [13] и оптимизированных структур данных в памяти обеспечило загрузку ЦПУ менее 70 %. Для сравнительного анализа в качестве альтернативы использовался сервис Yandex Speller, доступ к которому осуществлялся через публичный JSON-over-HTTP API. В ходе эксперимента был зафиксирован эффект низкой эффективности кэширования ответов внешнего сервиса: доля пропусков (cache miss) превысила 80 %. Это обусловлено спецификой сегмента DIY, характеризующегося высокой вариативностью технических поисковых запросов и выраженным «длинным хвостом» уникальных артикулов, что делает использование статических словарей и кэшей неэффективным.

В данных условиях производительность предложенного метода оказалась выше на 350–400 %. Столь значительный разрыв объясняется отсутствием сетевых задержек и высокой локальной скоростью обработки в компилируемом окружении (Go), что критично для соблюдения жестких SLA в высоконагруженных системах. При этом точность исправления коротких токенов увеличилась с 0,76 до 0,97 по метрике F1-score. Результаты подтверждают, что специализированное гибридное решение превосходит

универсальные лингвистические модели при работе с разреженными и технически насыщенными данными (Рисунок 1).

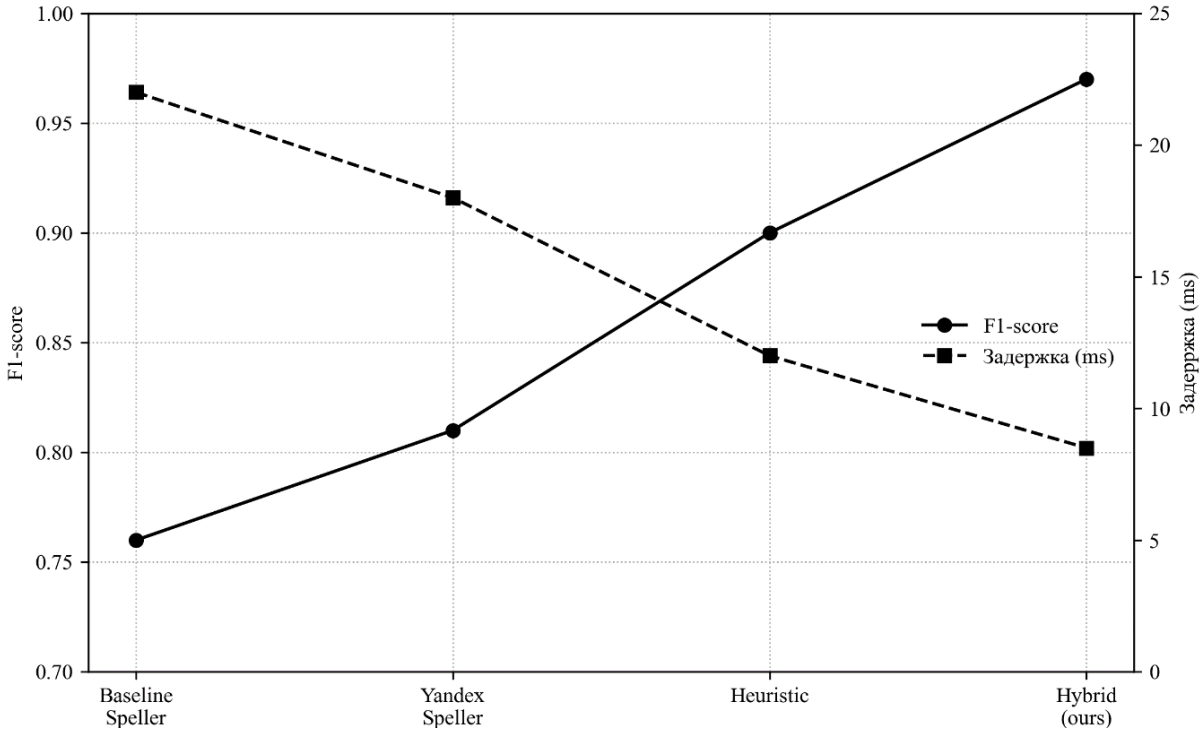

Рисунок 1 – Сравнение различных подходов к коррекции раскладки
Figure 1 – Comparison of layout correction approaches

Обсуждение

Полученные результаты демонстрируют, что предложенный комбинированный подход обеспечивает высокий уровень детерминизма и предсказуемости поведения системы при минимальном потреблении вычислительных ресурсов. В отличие от полноценных нейросетевых архитектур общего назначения, решение ориентировано на строго формализованную задачу коррекции раскладки и потому избегает избыточной параметризации. Это особенно важно для серверных поисковых систем, функционирующих в режиме постоянной высокой нагрузки и ограниченных по CPU- и memory-бюджету.

Отказ от тяжелых трансформерных архитектур, таких как mT5 [14] и BART [15], позволил устранить характерные для них латентные задержки, превышающие 50 мс при инференсе на стандартных серверных конфигурациях. Несмотря на высокую выразительную способность трансформеров, их применение к задаче визуально-детерминированных ошибок раскладки оказывается методологически избыточным. Аналогично, субсловная токенизация SentencePiece [16] продемонстрировала ограниченную применимость в условиях обработки коротких технических обозначений («M8x50», «LED5W»), поскольку разбиение на подслова нарушает атомарность доменных токенов и усложняет процедуру обратной реконструкции исходной формы.

Таким образом, для высоконагруженных информационно-поисковых систем электронной коммерции оптимальной является архитектура, сочетающая:

– фиксированные таблицы маппинга символов ENG→RU,
– контекстное сопоставление на уровне n-грамм и категориальных признаков,

– легкие векторные модели fastText [13], интегрированные в компилируемое окружение Go.

Рост метрики F1 на отложенной выборке с 0,90 до 0,97 обусловлен не только качеством модели, но и оптимизацией всего конвейера обработки: фильтрацией шумовых токенов, балансировкой классов, настройкой порогов детекции и процедурой пост-валидации. Существенный вклад внесла квантизация параметров модели, позволившая уменьшить ее размер приблизительно на 30 % без статистически значимого снижения точности. Это повысило кэш-локальность и снизило накладные расходы на загрузку модели в память.

Следует подчеркнуть, что аналогичные показатели трудно воспроизвести в трансформерных архитектурах вследствие их высокой параметрической сложности и чувствительности к масштабированию. При увеличении числа запросов в секунду нелинейный рост вычислительных затрат приводит к деградации SLA. Предложенная архитектура, напротив, демонстрирует линейную масштабируемость и устойчивость к флуктуациям трафика, обеспечивая соблюдение требований SLA < 20 мс даже при пиковых нагрузках.

Тем самым результаты подтверждают, что для узкоспециализированных задач коррекции раскладки рациональнее использовать оптимизированные детерминированно-гибридные методы, чем универсальные нейросетевые решения общего назначения.

Заключение

Работа представила вычислительно эффективный подход к коррекции раскладки клавиатуры ENG→RU в поисковых системах электронной коммерции. Метод объединяет эвристические правила и легкое машинное обучение для восстановления технических токенов в реальном времени.

Эксперименты показали увеличение точности поиска на 25–30 % и сокращение времени отклика до < 10 мс. Подход доказал масштабируемость и применимость в системах с ограниченными ресурсами, обеспечивая устойчивую работу под высокой нагрузкой.

Перспективы включают расширение набора раскладок (RU→ENG), внедрение самообучения и исследование влияния коррекции на downstream-задачи – расширение запросов, ранжирование и персонализацию поиска.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Lee J.S., Choi K.S. English to Korean statistical transliteration for information retrieval. *Comput Process Orient Lang*. 1998;12(1):17–37.
2. Pogrebnoi D., Funkner A., Kovalchuk S. RuMedSpellchecker: correcting spelling errors for natural Russian language in electronic health records using machine learning techniques. In: *International Conference on Computational Science, 3–5 July 2023, Prague, Czech Republic.* Cham: Springer Nature Switzerland; 2023. p. 213–227. https://doi.org/10.1007/978-3-031-36027-5_17
3. Prabhakar D.K., Pal S. Machine transliteration and transliterated text retrieval: a survey. *Sādhanā*. 2018;43(6). https://doi.org/10.1007/s12046-018-0879-4
4. Balabaeva K., Funkner A., Kovalchuk S. Automated spelling correction for clinical text mining in Russian. In: *Digital Personalized Health and Medicine, 17–19 November 2020, Amsterdam, Netherlands.* Amsterdam: IOS Press; 2020. p. 43–47. https://doi.org/10.3233/SHTI200119

5. Rozovskaya A. Spelling correction for Russian: a comparative study of datasets and methods. In: *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), 1–3 September 2021, Varna, Bulgaria.* Shoumen: INCOMA Ltd.; 2021. p. 1206–1216.
6. Bruch S., Lucchese C., Maistro M., et al. Special section on efficiency in neural information retrieval. *ACM Trans Inf Syst*. 2024;42(5):1–4. https://doi.org/10.1145/3641203
7. Chari A., Ounis I., MacAvaney S. Lost in transliteration: bridging the script gap in neural IR. In: *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, 13–18 July 2025, Padua, Italy.* New York: Association for Computing Machinery; 2025. p. 2900–2905.
8. Toutanova K., Moore R.C. Pronunciation modeling for improved spelling correction. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 7–12 July 2002, Philadelphia, USA.* Philadelphia: Association for Computational Linguistics; 2002. p. 144–151. https://doi.org/10.3115/1073083.1073110
9. Sachdeva N., McAuley J. How useful are reviews for recommendation? A critical review and potential improvements. In: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 25–30 July 2020, Xi'an, China.* New York: Association for Computing Machinery; 2020. p. 1845–1848. https://doi.org/10.1145/3397271.3401286
10. Mikolov T., Chen K., Corrado G., et al. *Efficient estimation of word representations in vector space.* arXiv. URL: https://doi.org/10.48550/arXiv.1301.3781 [Accessed 1st February 2026].
11. Belinkov Y., Bisk Y. *Synthetic and natural noise both break neural machine translation.* arXiv. URL: https://doi.org/10.48550/arXiv.1711.02173 [Accessed 1st February 2026].
12. Müller L., Juozapavičius A., Okhrimchuk V., et al. Dictionary attack with transformed Russian words using QWERTY keyboard layout. *Baltic Journal of Modern Computing*. 2025;13(4):919–932. https://doi.org/10.31219/osf.io/mfqrw
13. Joulin A., Grave E., Bojanowski P., et al. *Bag of tricks for efficient text classification.* arXiv. URL: https://doi.org/10.48550/arXiv.1607.01759 [Accessed 1st February 2026].
14. Xue L., Constant N., Roberts A., et al. *mT5: a massively multilingual pre-trained text-to-text transformer.* arXiv. URL: https://doi.org/10.48550/arXiv.2010.11934 [Accessed 1st February 2026].
15. Lewis M., Liu Y., Goyal N., et al. *BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.* arXiv. URL: https://doi.org/10.48550/arXiv.1910.13461 [Accessed 1st February 2026].
16. Kudo T., Richardson J. *SentencePiece: a simple and language independent subword tokenizer and detokenizer for neural text processing.* arXiv. URL: https://doi.org/10.48550/arXiv.1808.06226 [Accessed 1st February 2026].

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Краснов Федор Владимирович, кандидат технических наук, специалист по поисковым и рекомендательным системам в электронной коммерции, сотрудник ООО «Ви.тех», Ковров, Российская Федерация.
e-mail: fedor.krasnov@vi.ru
ORCID: 0000-0002-9881-7371

Fedor V. Krasnov, Candidate of Engineering Sciences, Search and Recommendation Systems Specialist in E-commerce, Employee of VI.Tech LLC, Kovrov, the Russian Federation.

Статья поступила в редакцию 27.02.2026; одобрена после рецензирования 13.04.2026; принята к публикации 18.07.2026.

The article was submitted 27.02.2026; approved after reviewing 13.04.2026; accepted for publication 18.07.2026.