МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**
Кафедра математической кибернетики и компьютерных наук

# РАЗРАБОТКА МОДЕЛЕЙ ПОИСКОВОЙ СИСТЕМЫ НА ОСНОВЕ РАНЖИРОВАНИЯ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Янченко Вадима Александровича

Научный руководитель
доцент, к. ф.-м. н. _____ С. В. Папшев

Заведующий кафедрой
к. ф.-м. н., доцент _____ С. В. Миронов

Саратов 2026

Информационный поиск (information retrieval, IR) является одной из ключевых задач компьютерной лингвистики и информационных технологий. Ее цель заключается в поиск наиболее релевантных документов из базы данных по запросу пользователя. Постоянный рост объёма текстовых данных приводит к необходимости создавать методы, которые будут всё эффективнее находить релевантную информацию.

Классические модели IR, основанные на статистике встречаемости слов в документах, до сих пор используются благодаря своей простоте и интерпретируемости. Однако такие модели обеспечивают низкое качество поиска, так как связь между запросом и документом часто не зависит только от совпадения слов. Появление трансформерных моделей привело к повышению качества анализа текста и, как следствие, поиска. Однако подобные модели остаются вычислительно сложными, что не позволяет их применять в качестве основного компонента поисковой системы.

Одним из решений этой проблемы является использование поисковых систем, выполняющих свою работу в два этапа. На первом этапе с помощью быстрой модели происходит первичное извлечение небольшого количества документов, а на втором этапе — их упорядочивание по степени релевантности к запросу более сложной моделью. Второй этап называется ранжированием и позволяет лучше выявить семантическую связь между запросом и документами, что улучшает качество итоговой выдачи.

Одна из проблем в обучении моделей ранжирования заключается в необходимости наличия большого количества данных. Создание подобных датасетов вручную требует значительных временных и финансовых затрат, а также требует привлечения экспертов предметной области. Высокое качество работы больших языковых моделей (LLM) позволяет их использовать для генерации таких данных. Это делает возможным обучение моделей ранжирования даже при отсутствии крупных вручную размеченных корпусов. Для повышения качества синтетических данных используются специальные методы разработки промптов, включающие подробные инструкции, примеры и ограничения на формат ответа.

Актуальность заключается в необходимости разработки поисковых систем, обеспечивающих высокое качество поиска при ограниченном количестве размеченных данных. Использование сгенерированных данных в обучении мо-

дели ранжирования может сильно упростить построение поисковых систем для узкоспециализированных предметных областей.

Целью данной работы является разработка и анализ поисковой системы новостей, использующей модели ранжирования, обученные на синтетических данных.

Для достижения цели были поставлены следующие задачи:

1. изучить современные методы информационного поиска и подходы к ранжированию документов;
2. исследовать архитектуры моделей ранжирования на основе трансформеров;
3. реализовать систему полнотекстового поиска на основе поисковой системы Elasticsearch с использованием алгоритма BM25;
4. разработать алгоритм генерации пользовательских запросов и автоматической разметки релевантности документов к запросу с помощью больших языковых моделей;
5. сформировать датасет для обучения моделей ранжирования;
6. обучить модели архитектур Cross-encoder и ColBERT;
7. провести анализ качества поисковой системы, использующих обученные модели ранжирования.

В рамках данной работы была реализована система поиска новостей, состоящая из двух этапов: этапа отбора и ранжирования. В качестве модели, используемой на стадии отбора, была выбрана модель BM25. В стадии ранжирования использовались обученные Cross-encoder и ColBERT модели. Обучение этих моделей происходило на синтетических данных, полученных с помощью модели семейства Qwen. После обучения модели ранжирования сравнивались на различных метриках.

В качестве новостных документов были выбраны все новости Саратовского государственного университета в период с 02.04.2007 по 12.05.2022, размещенные на официальном сайте университета `https://www.sgu.ru/`. Каждый документ содержал следующие поля:

— уникальный идентификатор новости;
— заголовок новости;
— необработанный текст новости;

Для реализации этапа первичного поиска использовалась система

Elasticsearch, использующая алгоритм полнотекстового поиска BM25. После создания индекса в него были загружены все документы корпуса.

Для хранения документов был создан специализированный индекс, содержащий текст новости, заголовок, дату публикации и служебные поля. При индексировании использовались только поля заголовка и текста документа, поскольку именно они участвуют в процессе поиска. Остальные поля сохранялись для последующего отображения результатов и формирования обучающих данных.

Развёртывание Elasticsearch выполнялось с использованием технологии контейнеризации Docker. Такой подход упрощает переносимость системы и обеспечивает воспроизводимость экспериментов. Для хранения индекса использовалось отдельное постоянное хранилище, что позволяло сохранять данные между перезапусками контейнера.

Поиск реализовывался через механизм multi_match, позволяющий одновременно учитывать совпадения по нескольким текстовым полям. Полю заголовка назначался повышенный вес, поскольку заголовок обычно наиболее точно отражает содержание документа.

В результате Elasticsearch вычислял оценки релевантности документов и возвращал наиболее подходящие результаты. Проведённые эксперименты показали, что BM25 эффективно работает при поиске по ключевым словам, однако не способен учитывать семантическую близость между запросом и документом.

Поскольку готовые размеченные данные отсутствовали, обучающий датасет формировался автоматически с использованием больших языковых моделей.

На первом этапе для новостных документов генерировались пользовательские запросы. Для этого использовалась модель Qwen3.5-9B, которой передавались заголовок и текст новости. При генерации учитывались особенности реальных поисковых запросов. Поэтому запросы могли содержать разговорные конструкции, неполные формулировки и различные способы описания одного и того же события. Для повышения разнообразия первоначально генерировалось десять вариантов запросов, после чего модель выполняла их дополнительную оценку и отбирала пять наиболее качественных. При выборе учитывались естественность формулировки, соответствие содержанию документа и отсутствие дублирования. В результате было сформировано 14905 поисковых запросов.

Для каждого сформированного запроса выполнялся поиск двадцати наиболее релевантных документов реализованным ранее поиском. Если исходный документ отсутствовал в поисковой выдаче, он добавлялся принудительно. Это позволяло гарантировать наличие положительного примера для каждого запроса и предотвращало формирование выборки, состоящей исключительно из нерелевантных документов. Таким образом, получилось $303038$ пар, состоящих из запроса и документа.

Для автоматической разметки релевантности использовалась языковая модель Qwen2.5-7B-Instruct. На вход модели подавались текст запроса, заголовок и содержимое документа. Для уменьшения вычислительных затрат использовались первые 300 слов текста документа. Результатом работы модели являлась оценка релевантности по шкале от 0 до 3:

0 — документ нерелевантен;

1 — документ имеет лишь тематическое пересечение с запросом;

2 — документ релевантен частично или содержит значимую информацию по запросу;

3 — документ полностью соответствует запросу.

В результате был получен датасет из 303038 троек вида «запрос–документ–оценка релевантности». Количество записей с оценкой релевантности $0$ равно $172308$ ($56.86\%$ от всей выборки), в свою очередь, количество записей с оценкой $1$ и $2$ равняется $59345$ ($19.58\%$) и $58774$ ($19.39\%$). Записей же с самой лучшей оценкой всего $12611$ ($4.16\%$ от общего количества).

Результаты показали преобладание нерелевантных документов, что соответствует реальным условиям работы поисковых систем и является важным фактором при обучении моделей ранжирования.

Дополнительный анализ сформированного набора данных показал, что для документов, на основе которых генерировались запросы, преимущественно присваивались оценки релевантности 2 и 3. Это свидетельствует о том, что автоматически сформированные запросы в большинстве случаев корректно отражают содержание соответствующих документов и могут использоваться для построения обучающих датасетов без привлечения ручной разметки.

После формирования датасета было выполнено обучение Cross-encoder модели ранжирования на основе предобученной модели `BAAI/bge-reranker-v2-m3`. Данная модель поддерживает более $100$ языков,

включая русский язык, и ориентирована на задачи информационного поиска и переранжирования документов.

Исходная четырёхуровневая шкала релевантности была преобразована в бинарную задачу классификации. Документы с оценками $0$ и $1$ относились к классу нерелевантных, тогда как документы с оценками $2$ и $3$ считались релевантными. Подобное преобразование обусловлено тем, что оценки $0$ и $1$ соответствуют отсутствию или слабой тематической связи между запросом и документом, тогда как оценки $2$ и $3$ характеризуют документы, содержащие значимую информацию по пользовательскому запросу. Кроме того, бинаризация позволила упростить процесс обучения и сделать распределение классов более устойчивым.

После подготовки данных датасет был разделён на обучающую, валидационную и тестовую выборки в соотношении $60\%$, $20\%$ и $20\%$. В качестве групп использовались идентификаторы запросов.

Для обучения использовалась библиотека `sentence-transformers`. Обучение выполнялось в течение двух эпох. Размер батча составлял $8$ примеров. Значение параметра learning rate было выбрано равным $2 \cdot 10^{-5}$, а коэффициент регуляризации weight decay — $0.01$. Для ускорения вычислений использовался режим смешанной точности `fp16`, позволяющий уменьшить объём используемой видеопамяти и сократить время обучения.

Во время обучения регулярно выполнялась оценка качества модели на валидационной выборке. Логирование процесса обучения осуществлялось средствами TensorBoard. В качестве основной функции потерь использовалась бинарная кросс-энтропия.

На начальном этапе обучения наблюдались колебания значения loss, что является типичным поведением для трансформерных моделей. Далее значения функции потерь на обучающей и валидационной выборках постепенно уменьшались, что свидетельствует об успешном обучении модели.

Помимо Cross-encoder модели была обучена модель архитектуры ColBERT. В качестве базовой модели использовалась мультиязычная модель `BAAI/bge-m3`, поддерживающая русский язык и предназначенная для решения задач информационного поиска.

Также как и в случае подготовки датасета для обучения Cross-encoder модели оценка релевантности документа на запрос преобразуется следующим

образом:

— оценка 0 и 1 преобразуется в 0;
— оценка 2 и 3 преобразуется в 1.

На основе размеченного датасета формировались триплеты вида «запрос — релевантный документ — нерелевантный документ». В качестве положительного примера использовался документ, для которого первоначально был сгенерирован запрос, а отрицательными примерами являлись документы с низкими оценками релевантности.

Всего было сформировано 227519 обучающих триплетов. После этого данные разделялись на обучающую, валидационную и тестовую выборки.

Для обучения использовалась библиотека PyLate, реализующая поддержку моделей архитектуры ColBERT. В качестве функции потерь применялся contrastive loss, обеспечивающий увеличение сходства между запросом и релевантным документом и уменьшение сходства между запросом и нерелевантными документами.

Обучение проводилось в течение двух эпох. Для ускорения вычислений использовался режим смешанной точности `fp16`. Контроль процесса обучения осуществлялся с использованием TensorBoard.

В процессе обучения модель демонстрировала устойчивую сходимость. Значения функции потерь на обучающей и валидационной выборках постепенно уменьшались, при этом существенного расхождения между ними не наблюдалось. Полученные результаты свидетельствуют об отсутствии выраженного переобучения и подтверждают способность модели эффективно решать задачу ранжирования для ранее не встречавшихся запросов.

Для оценки качества поиска были сгенерированы запросы для 1993 документов, отсутствовавших в обучающих выборках. Сравнение проводилось между тремя конфигурациями поисковой системы:

— поисковая система, использующая только алгоритм BM25, обозначаемая как BM25;
— поисковая система, использующая алгоритм BM25 для отбора документов и обученную Cross-encoder модель ранжирования, обозначаемая как BM25 + Cross-encoder;
— поисковая система, использующая алгоритм BM25 для отбора документов и обученную ColBERT модель ранжирования, обозначаемая как BM25 +

ColBERT.

В качестве основных метрик использовались средняя обратная позиция (MRR), средняя позиция релевантного документа в выдаче, доля попаданий в топ-5 и время обработки запроса.

Для поисковой системы BM25 документ, для которого был сгенерирован запрос, оказывается на первой позиции в $28.2\%$ случаев. В топ-5 документ попадает в $46.4\%$ случаев, а значение MRR составляет $0.37$. При этом средняя позиция целевого документа равняется $33.3$. Подобные результаты демонстрируют ограничения классических алгоритмов поиска: алгоритм BM25 эффективно находит документы по совпадению слов, однако не способен учитывать семантическую близость текстов.

Применение Cross-encoder модели ранжирования в дополнении к алгоритму BM25 приводит к заметному улучшению качества поиска. Доля случаев, когда документ, на основе которого был сгенерирован запрос, оказывается на первой позиции, возрастает до $32.2\%$, а попадание в топ-5 увеличивается до $62.1\%$. Значение MRR возрастает до $0.46$, а средняя позиция целевого документа уменьшается больше чем в три раза и составляет $9.1$. Кроме того, количество запросов, в которых релевантный документ находится ниже десятой позиции, снижается с $918$ до $505$. Это подтверждает, что Cross-encoder модель позволяет существенно улучшить качество поиска.

Модель ранжирования архитектуры ColBERT совместно с алгоритмом BM25 демонстрирует наилучшие результаты. Доля попадания целевого документа на первую позицию составляет $54.8\%$, в топ-5 документ оказывается в $82.5\%$ случаев. Значение MRR равно $0.671$, а средняя позиция целевого документа составляет $4.5$. Также уменьшается количество случаев, когда релевантный документ оказывается ниже десятой позиции ($202$ запросов против $918$ у BM25 и $505$ у BM25 + Cross-encoder).

Также был проведён анализ времени обработки запросов. Наиболее быстрой оказалась система BM25 со средним временем ответа около $0.1$ секунды. Для системы BM25 + Cross-encoder среднее время обработки запроса составило $1.4$ секунды. Система BM25 + ColBERT работала быстрее и обеспечивала среднее время ответа около $1$ секунды.

Для дополнительной оценки качества поиска было проведено тематическое моделирование новостного корпуса средствами Gensim и BigARTM. На

основе запросов были построены матрицы распределения тем, отражающие соответствие между темой документа, для которого был сгенерирован запрос, и доминирующими темами документов, возвращаемых поисковой системой.

Анализ результатов для системы на основе BM25 показал наличие заметного смешения тематик в выдаче (рис. 0.1). Значения главной диагонали, отражающие соответствие целевой тематике, варьируются в диапазоне от 16% до 26%

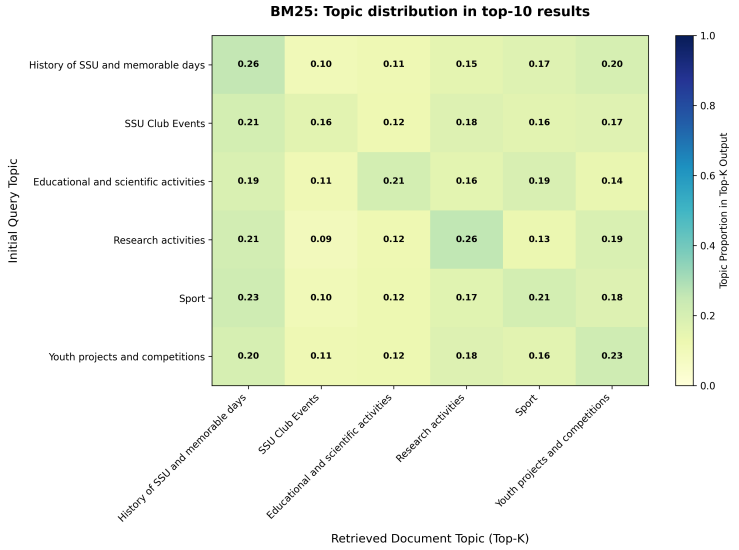


Рисунок 0.1 – Матрицы распределения тем документов в топ-10 выдаче системы на основе BM25

Использование модели ранжирования Cross-Encoder позволило повысить тематическую согласованность выдачи (рис. 0.2). В результатах поиска увеличилась доля документов, относящихся к тематике исходного запроса.

Наиболее качественные результаты были получены при использовании модели ColBERT (рис. 0.3). Данная модель обеспечила наиболее сбалансированное распределение тематик и наилучшее соответствие между содержанием запроса и тематикой найденных документов.

Полученные результаты подтверждают, что применение моделей ранжирования не только улучшает количественные метрики поиска, но и повышает тематическую точность поисковой выдачи.

Основные результаты сравнения представлены в таблице 0.1. Проведённые эксперименты показали, что обе обученные модели улучшают качество поиска относительно базового алгоритма BM25. Наиболее эффективной оказалась система, сочетающая алгоритм BM25 и модель ранжирования архитектуры

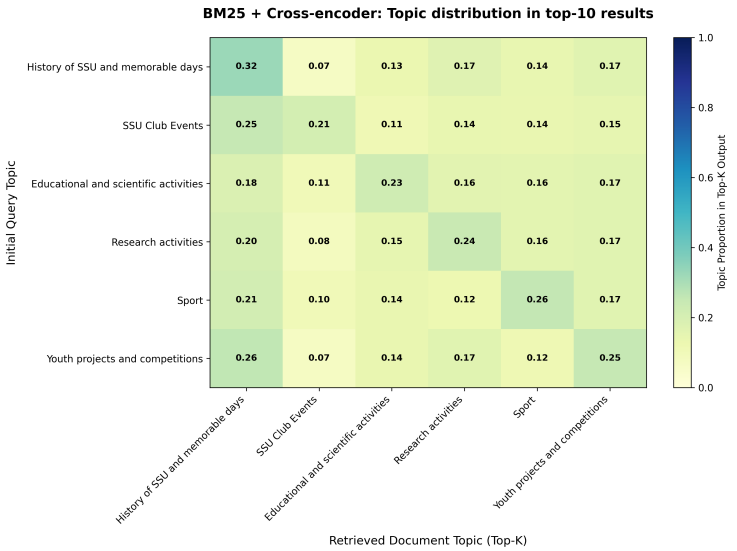


Рисунок 0.2 – Матрицы распределения тем документов в топ-10 выдаче системы, состоящей из BM25 + Cross-Encoder

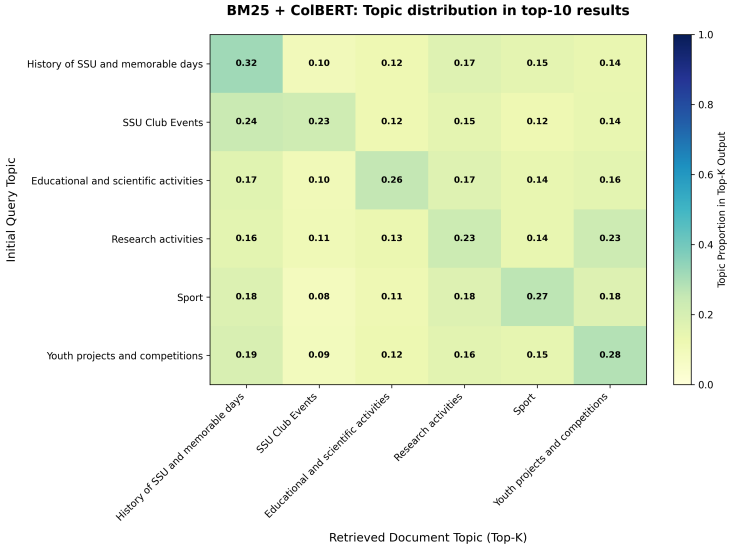


Рисунок 0.3 – Матрицы распределения тем документов в топ-10 выдаче системы, состоящей из BM25 + ColBERT

ColBERT. Использование моделей ранжирования влияет в лучшую сторону на распределение тем в выдаче. Наиболее сбалансированное распределение значений тем наблюдалось у системы, использующей алгоритм BM25 и модель ранжирования ColBERT.

Таким образом, была разработана и исследована поисковая система, использующая двухэтапную архитектуру. На первом этапе осуществлялся отбор документов с помощью алгоритма BM25 через систему Elasticsearch, а на втором этапе выполнялось ранжирование документов с использованием обученных Cross-encoder и ColBERT моделей.

Таблица 0.1 – Значение основных метрик для всех рассматриваемых конфигураций поисковых систем

| **Конфигурация поисковой системы** | **MRR** | **Средняя позиция документа** | **Попадание документа в топ-5 (%)** | **Среднее время обработки запроса (сек.)** |
|---|---|---|---|---|
| BM25 | 0.37 | 33.3 | 46.4% | 0.1 |
| BM25 + Cross-encoder | 0.46 | 9.1 | 62.1% | 1.4 |
| BM25 + ColBERT | 0.67 | 4.5 | 82.5% | 1 |

Обучающие данные формировались с помощью больших языковых моделей. Через модель семейства Qwen были сгенерированы пользовательские запросы, а затем выполнена разметка релевантности документов, выданных алгоритмом BM25, на запрос. В результате был сформирован набор данных, содержащий более 300 тысяч троек, состоящих из запроса, документа и оценки релевантности документа относительно запроса. На полученном датасете были обучены две модели ранжирования архитектуры Cross-encoder и ColBERT.

Результаты работы демонстрируют, что применение моделей ранжирования позволяет существенно повысить качество информационного поиска. Несмотря на отсутствие вручную размеченного корпуса, модели, обученные на сгенерированных данных, показали высокую эффективность и обеспечили повышение качества поисковой выдачи.