

УДК 004.912

doi: 10.15622/rcai.2025.031

АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ КЛЮЧЕВЫХ СЛОВ ДЛЯ РУССКОЯЗЫЧНЫХ НАУЧНЫХ СТАТЕЙ С ИСПОЛЬЗОВАНИЕМ ПСЕВДОРАЗМЕТКИ И КОНТРАСТИВНОГО ОБУЧЕНИЯ

К.Ш. Яушев (*kyaush@mail.ru*)^A

Н.В. Лукашевич (*louk_nat@mail.ru*)^B

^A Московский государственный технический университет
им. Н.Э. Баумана, Москва

^B Московский государственный университет
им. М.В. Ломоносова, Москва

В работе предложен многоэтапный подход к автоматическому порождению ключевых слов для русскоязычных научных статей. Метод основан на дообучении трансформеров с использованием псевдоразметки и контрастивного обучения, а также включает фильтрацию порождённых кандидатов. Реализованы две стратегии генерации псевдоразметки и архитектура с биэнкодером для отбора релевантных ключевых слов. Эксперименты на корпусе математики и компьютерных наук демонстрируют превосходство предложенного подхода над классическими и нейросетевыми методами по метрикам F1, ROUGE-1 и BERTScore.

Ключевые слова: генерация ключевых слов, автоматическая разметка, контрастивное обучение, фильтрация ключевых слов.

Введение

В последние годы объём научных публикаций растёт стремительно [Cicero, 2025], что усложняет систематизацию и поиск информации. В условиях информационной перегрузки ключевые слова играют важную роль для навигации в библиографических системах (Scopus, Web of Science, eLIBRARY) и влияют на индексацию и цитируемость научных работ [Гендина и др., 2018].

Размеченные авторами ключевые слова могут быть слишком специфичными для узкой тематики статьи или, напротив, чрезмерно общими, что затрудняет их использование в поисковых системах и при тематиче-

ской классификации. Это делает актуальной задачу автоматической генерации ключевых слов, позволяющую формировать более сбалансированный и релевантный набор терминов для поиска и анализа.

Автоматическая генерация ключевых слов для русскоязычных текстов остаётся сложной из-за богатой морфологии, синтаксического разнообразия и отсутствия явных ключевых концептов в тексте [Glazkova et. al., 2024]. Для успешного решения задачи модели должны уметь выявлять скрытые семантические связи – неявные смысловые отношения между терминами, тематическими группами и контекстно связанными выражениями. Такие связи определяются на основе распределённых представлений и совместной встречаемости в корпусе. Современные большие языковые модели (LLM) на основе трансформеров доказали эффективность в генерации текстов и извлечении смысловых единиц [Vaswani et. al., 2017], однако их применение для русского языка ограничено дефицитом размеченных данных [Glazkova et. al., 2025].

В этой работе предложен многоэтапный метод, сочетающий псевдоразметку и контрастивное обучение, направленный на улучшение генерации как явно представленных, так и отсутствующих ключевых слов при ограниченных размеченных ресурсах.

1. Близкие работы

Методы выделения ключевых слов делятся на экстрактивные и абстрактивные. Экстрактивные (статистические: TF-IDF [Salton et. al., 1975], YAKE! [Campos et. al., 2020]; графовые: TextRank [Mihalcea et. al., 2004], TopicRank [Bougouin et. al., 2013]) выбирают значимые слова из текста, отличаются простотой и интерпретируемостью, но не могут генерировать отсутствующие в документе слова [Glazkova et. al., 2024].

Нейросетевые трансформеры (BERT [Devlin et. al., 2019], T5 [Raffel et. al., 2020], mBART [Tang et. al., 2021]) обучены на больших корпусах и способны выявлять глубокие семантические связи, генерируя новые релевантные ключевые слова. Среди русскоязычных моделей выделяются ruT5 [Zmitrovich et. al., 2024], Vikhr [Nikolich et. al., 2024] и Saiga [Gusev, 2023]. Главная сложность — генерация точных по смыслу ключевых слов при ограниченном объёме данных.

Ранее в русскоязычной тематике изучались модели mT5 и mBART, подтвердившие потенциал генеративных подходов [Glazkova et. al., 2024]. Инструктивные LLM-модели показали высокую эффективность в few-shot режиме, но требуют значительных ресурсов [Glazkova et. al., 2025].

Также развивается направление, в котором генеративные подходы дополняются механизмами фильтрации и ранжирования: например, с использованием бинкодеров для отбора релевантных кандидатов по семантической близости [Choi et. al., 2023] или применения псевдоразметки для расширения обучающей выборки за счёт неразмеченных текстов [Kang et. al., 2024].

Настоящее исследование развивает эти направления, предлагая методы с псевдоразметкой и контрастивным обучением для повышения качества генерации при дефиците размеченных данных.

2. Методы автоматической разметки и архитектура обучения

Для преодоления дефицита размеченных данных предлагается двух-этапный подход, основанный на идеях [Kang et. al., 2024] и адаптированный под особенности русскоязычных научных текстов. На первом этапе проводится предобучение генеративной модели на корпусе, обогащённом псевдоразметкой, на втором этапе тонкая настройка на размеченных данных, чтобы развить способность модели не только извлекать, но и порождать ключевые слова.

Псевдоразметка формируется по двум стратегиям. Первая основана на *маскировании* слабо релевантных, но явно присутствующих ключевых слов. Для каждой аннотации библиотекой `ruTermExtract`¹ извлекается упорядоченный по релевантности список ключевых слов (именных форм) в корректных словоформах, учитывая морфологические особенности русского языка. Релевантность определяется косинусной близостью векторных представлений ключевых слов и текста, полученных с помощью модели E5². Первые 5 наиболее релевантных ключевых слов считаются эталонными и сохраняются в тексте, а позиции с 6-й по 10-ю маскируются специальным токеном. Такой отбор исключает заведомо нерелевантные выражения и фокусирует модель на восстановлении терминов, связанных по смыслу, но не очевидных, что способствует генерации отсутствующих в тексте ключевых слов.

Вторая стратегия обучает генерацию релевантных, но явно непредставленных ключевых слов. Для этого используется «глобальная коллекция» – топ-5 ключевых слов, извлечённых `ruTermExtract` из остальных документов корпуса. Поиск семантически близких кандидатов выполняется методом HNSW [Malkov et. al., 2018], выбираются 5 наиболее близких по косинусному сходству и отсутствующих в исходном тексте ключевых слов. Они добавляются как целевые метки для обучения абстрактивной генерации.

Рассмотрим статью И.А. Чиждова и Н.П. Засца «Моделирование процесса теплопроводности многослойной конструкции для выполнения тепловизионного контроля» [Чиждов и др., 2015]. Разметка выполняется по заголовку и аннотации. Ниже приведён текст с псевдоразметкой (цитата по [Чиждов и др., 2015, с. 1]):

¹ <https://github.com/igor-shevchenko/rutermextract>.

² <https://huggingface.co/d0rj/e5-large-en-ru>.

[Моделирование процесса] [теплопроводности многослойной конструкции] для выполнения [тепловизионного контроля]. В статье представлен <этап моделирования> протекания [тепловых потоков] в многослойной конструкции, применяемые <численные решения> для <математического моделирования>. Перечислены учитываемые в модели <параметры> и проведён <анализ факторов> [теплового неразрушающего контроля].

В квадратных скобках представлены явно присутствующие ключевые слова; в угловых – маскируемые, предназначенные для генерации.

Ключевые слова, найденные глобально в коллекции, отсутствующие в тексте аннотации: *моделирование процесса теплопередачи, тепловизионный анализ, моделирование процессов теплопереноса, исследование проблем теплопередачи, численное моделирование процесса теплопередачи.*

Для повышения устойчивости к шуму и улучшения различения релевантных и нерелевантных ключевых слов применяется архитектура с элементами контрастивного обучения (рис. 1), предложенная в [Choi et. al., 2023]. Она включает экстрактор-генератор и ранкер.

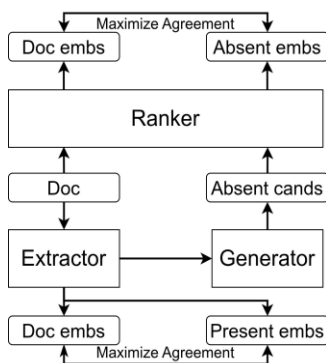


Рис. 1. Архитектура генерации и фильтрации ключевых слов

Экстрактор-генератор (рис. 2) – энкодер-декодерная модель, одновременно обучаемая на задаче извлечения и генерации с помощью комбинированной функции потерь с контрастивной частью (NT-Xent) и максимальным правдоподобием (MLE) [Chen et. al., 2020]:

$$(2.1)$$

Это позволяет объединить точность экстракции с обобщающей способностью генерации. Для формирования негативных примеров в контрастивном обучении применяется *hard negative mining*: в качестве «труд-

ных» негативов отбирается не более 5-ти кандидатных ключевых слов, извлечённые с помощью `ruTermExtract`, которые отсутствуют в эталонной разметке (псевдоразметке) датасета, но имеют высокое семантическое сходство с релевантными ключевыми словами. Такой подход помогает модели точнее проводить границу между действительно релевантными и нерелевантными ключевыми словами, минимизируя влияние тривиальных различий.

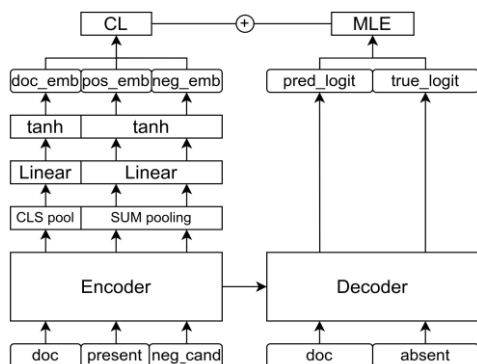


Рис. 2. Архитектура экстрактора-генератора с контрастивным модулем

Ранкер – биэнкодерная модель, контрастивно обученная различать релевантные и нерелевантные ключевые слова. После генерации кандидатных ключевых слов (например, с применением `beam search`) ранкер отбирает наиболее подходящие, фильтруя шум и снижая вероятность появления «галлюцинаций».

Общая стратегия обучения включает два этапа. Сначала модели обучаются на псевдоразмеченных данных, затем проводится тонкая настройка на вручную размеченном корпусе научных аннотаций, что позволяет повысить точность и адаптировать модели к специфике предметной области.

3. Эксперименты

3.1. Данные

В качестве основного размеченного корпуса для обучения и тестирования использовался датасет `Math&CS` состоящий из 8348 аннотаций русскоязычных научных статей из области математики и компьютерных наук. Разделение на обучающую (5844) и тестовую (2504) выборки выполнено авторами датасета. В среднем каждая аннотация содержит от 3-х до 5-ти

³ https://huggingface.co/datasets/aglazkova/keyphrase_extraction_russian.

(4.34±1.5) эталонных ключевых слов. Важной особенностью корпуса является то, что 53.66% эталонных ключевых слов отсутствуют явно в тексте аннотаций, что подчеркивает необходимость применения генеративных подходов.

Для псевдоразметки использовался обширный корпус ruSciBench⁴, содержащий более 190 тысяч аннотаций научных статей из различных областей.

3.2. Используемые модели и их параметры

Для всестороннего анализа были выбраны модели из трех категорий:

1. Классические алгоритмы: YAKE!⁵ и ruTermExtract⁶.
2. Нейросетевые модели: mT5-base 580M⁷, mBART-large 610M⁸, e5-large-en-ru 366M⁹ (ранкер) а также несколько конфигураций модели ru-mbart-summ 380M¹⁰, дообученной на задаче суммаризации:
 - &mask: обучение с маскированием;
 - &generator: модель с контрастивной функцией потерь;
 - &generator&ranker: полная модель с генератором и ранкером.
3. Инструктивные нейросетевые модели: Mistral-7B-Instruct¹¹, Vikhr-7B-Instruct¹² и Saiga-Mistral-7B-Lora¹³ в режимах zero-shot и few-shot.

В качестве базовой была выбрана модель ru-mbart-summ – дообученная версия mbart_ru_sum_gazeta¹⁴, изначально обученной на новостном корпусе и превосходящей T5 и GPT-3 по метрикам суммаризации. Дополнительное обучение на расширенном русскоязычном наборе повысило её способность обобщать тексты разных доменов. Архитектура «энкодер–декодер» и специализация на сжатии содержания делают её подходящей для генерации ключевых слов, поскольку эта задача требует выделения и формулирования основной идеи текста.

Настройка параметров следовала рекомендациям из оригинальных работ [Glazkova et. al., 2025]. Для энкодер–декодерных моделей (mT5, mBART, ru-mbart-summ и их модификаций) использовались: 10 эпох, максимальная длина входа 256 токенов, learning rate $4 \cdot 10^{-5}$, оптимизаторы Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 4 \cdot 10^{-8}$) и AdamW (с weight_decay =

⁴ https://huggingface.co/datasets/mlsa-iai-msu-lab/ru_sci_bench.

⁵ <https://github.com/boudinfl/pke>.

⁶ <https://github.com/igor-shevchenko/rutermextract>.

⁷ <https://huggingface.co/google/mt5-base>.

⁸ <https://huggingface.co/facebook/mbart-large-50>.

⁹ <https://huggingface.co/d0rj/e5-large-en-ru>.

¹⁰ <https://huggingface.co/d0rj/ru-mbart-large-summ>.

¹¹ <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2>.

¹² https://huggingface.co/Vikhrmodels/Vikhr-7B-instruct_0.4.

¹³ https://huggingface.co/IlyaGusev/saiga_mistral_7b_lora.

¹⁴ https://huggingface.co/IlyaGusev/mbart_ru_sum_gazeta.

0.01 для моделей с дополнительной головой). Параметры генерации: `repetition_penalty = 1.4`, `num_beams = 100`, `no_repeat_ngram_size = 2`, `num_return_sequences = 16` для моделей с ранжированием. Для инструктивных моделей (Mistral, Vikhr, Saiga) применялись zero-/few-shot режимы с промптами на русском по шаблону [Glazkova et al., 2025], ограничение генерации 100 токенами и температура 0.5. Контрастивное обучение генераторов использовало $\tau = 0.1$ для потерь NT-Xent и $\gamma = 0.3$ при комбинировании с MLE (формула 2.1). При тестировании экстрактор выбирал 5 лучших кандидатов от `ruTermExtract`, генератор ограничивался 5 ключевыми словами, прочие генеративные модели – 10.

3.3. Метрики оценки

Качество генерации оценивалось с помощью трех метрик, рассчитанных для топ-10 сгенерированных ключевых слов:

1. `F1-score@10`: Гармоническое среднее точности и полноты для точных совпадений лемматизированных ключевых слов.
2. `ROUGE-1@10`: F1-мера перекрытия униграмм между сгенерированным и эталонным списками. Перед вычислением метрики ключевые слова лемматизировались и объединялись в одну строку через пробел.
3. `BERTScore@10`: Семантическая F1-мера, основанная на косинусном сходстве BERT¹⁵-эмбеддингов токенов. Перед вычислением метрики ключевые слова объединялись в одну строку через запятую.

4. Результаты и обсуждение

4.1. Сравнительный анализ моделей

Сводные результаты экспериментов приведены в табл. 1. В скобках указаны результаты из предыдущего исследования [Glazkova et al., 2025] для сравнения. При этом прямое сопоставление ограничено: в [Glazkova et al., 2025] для экстрактивных методов выбиралось лучшее значение среди топ-5, топ-10 и топ-15, тогда как в данной работе все модели оценивались при фиксированном топ-10. Кроме того, в [Glazkova et al., 2025] метрика ROUGE-1 вычислялась без лемматизации, а ключевые слова объединялись через запятую, тогда как в настоящем исследовании применялась лемматизация, а объединение выполнялось через пробел. Генеративные модели в предыдущей работе формировали не более 10 ключевых слов. Эти различия в методологии частично объясняют, почему при одинаковых моделях наши значения метрик, в том числе BERTScore, могут быть ниже, чем в [Glazkova et al., 2025].

¹⁵ <https://huggingface.co/google-bert/bert-base-multilingual-cased>.

Таблица 1

Модель	F1@10	ROUGE1@10	BERTScore@10
ruTermExtract	10.19 (11.02)	<u>29.97</u> (15.12)	71.26 (75.95)
YAKE!	04.04 (06.06)	<u>26.27</u> (06.47)	<u>70.56</u> (69.13)
mT5-base	4.42 (13.41)	<u>17.99</u> (15.14)	67.86 (76.07)
mBart-large	16.43 (16.84)	<u>33.65</u> (19.26)	72.34 (78.66)
ru-mbart-summ	16.46	33.61	74.32
&mask	17.80	34.91	75.30
&gen	02.04	06.41	77.71
&gen&rank	15.43	33.38	77.95
Mistral&few-shot	13.37 (15.08)	14.45 (16.30)	<u>76.11</u> (74.85)
Vikhr&few-shot	14.57 (15.18)	17.67 (19.62)	77.14 (77.48)
Saiga&few-shot	15.04 (20.16)	17.98 (22.37)	78.11 (79.50)

В данном исследовании модификация ru-mbart-summ&mask показала лучшие значения F1 и ROUGE-1, что подтверждает эффективность двух-этапного обучения с маскированием. Подход обеспечивает генерацию точных и релевантных ключевых слов даже при ограниченном объёме размеченных данных.

Модель &gen&rank, несмотря на умеренные значения F1 и ROUGE-1, достигла наивысшего BERTScore среди моделей с учителем в данной работе. Однако по BERTScore абсолютным лидером остаётся Saiga в режиме few-shot из [Glazkova et al., 2025], что может быть связано с различиями в настройках генерации (например, температурой), вариативностью выходов инструктивных LLM и более агрессивной стратегией перефразирования, которая повышает семантическое сходство.

Следует отметить ограниченность сравнений: модель с фильтрацией (&ranker) не была применена к конфигурации с маскированием, которая лидирует по F1/ROUGE-1. Комбинирование этих подходов, вероятно, позволило бы достичь баланса между точностью и семантической насыщенностью, что представляет интерес для дальнейшей работы.

4.2. Примеры успешных и неудачных предсказаний

Для анализа был взят пример из статьи «Процедурное программирование, операторы ветвления и циклов в системе компьютерной алгебры» [Дубовицкий, 2012].

Эталонные ключевые слова: *процедурное программирование, оператор ветвления, операторы циклов, система компьютерной алгебры, система компьютерной алгебры mathpar*. Сводные результаты экспериментов представлены в табл. 2.

Таблица 2

Модель	Сгенерированные ключевые слова
YAKE!	процедурный программирование, оператор ветвление, компьютерный алгебра mathpar, создание оператор ветвление
ruTermExtract	процедурный программирование, компьютерный алгебра, система, алгоритм, создание оператор ветвление
mT5-base	информационная система, веб-технология, информатика, метод ветвления, цикл
mBart-large	алгебра, ветвление, цикл, программа, процедурный программирование
ru-mbart-summ	процедурное программирование, система компьютерной алгебры, структура системы, операторы ветвления и циклов
ru-mbart-summ&mask	процедурное программирование, система компьютерной алгебры, структура системы, операторы ветвления и циклов
ru-mbart-summ&gen	компьютерная алгебра, процедурное программирование, процедуры, алгоритмы, цикл, алгебраические операторы, архитектура системы
ru-mbart-summ&gen&rank	компьютерная алгебра, процедурное программирование, процедуры, алгоритмы, цикл, алгебраические системы, архитектура системы, ветвление, алгебраические методы, оператор ветвления
Mistral&few-shot	процедурное программирование, операторы ветвления и циклов, алгоритмы, компьютерная алгебра, mathpar

Классические методы, такие как YAKE! и ruTermExtract, показывают ограниченную способность к морфологически точной генерации и склонны к порождению лексически искажённых форм (оператор ветвление, компьютерный алгебра). Это отражается в низких значениях F1 и ROUGE-1, несмотря на поверхностную релевантность отдельных терминов.

Модель ru-mbart-summ&mask демонстрирует наилучшее соответствие эталону как в лексическом, так и в семантическом плане. Её предсказания почти полностью совпадают с референсными ключевыми словами, что объясняет высокие значения F1 и ROUGE-1 и делает её лидером по этим метрикам.

Модель ru-mbart-summ&gen&rank обеспечивает высокий уровень семантического разнообразия: она охватывает больше аспектов, включая термины второго порядка (алгоритмы, архитектура системы). Однако избыточность и наличие пересечений между фразами снижают точность на уровне F1/ROUGE, несмотря на один из самых высоких BERTScore, отражающий её семантическое богатство.

Инструктивные модели, такие как Mistral&few-shot, показывают высокое качество генерации: предсказания совпадают с эталоном и грамматически, и семантически. Это подтверждается их сильными значениями BERTScore. Однако, в отличие от ru-mbart-summ&mask, они уступают по F1 и ROUGE, что может быть связано с большей вариативностью выходов и нестабильностью при генерации.

Заключение

В работе представлен и экспериментально оценён комплексный подход к автоматическому порождению ключевых слов для аннотаций русскоязычных научных статей. Основной вклад заключается в разработке и анализе архитектур, эффективно работающих при ограниченном количестве размеченных данных.

Двухэтапная стратегия обучения – предобучение на псевдоразмеченных данных и донастройка на небольшом объёме эталонных аннотаций – существенно повышает качество генерации. Модель ru-mbart-summ&mask достигла наивысших значений F1 и ROUGE-1, демонстрируя высокую лексическую точность и стабильность, что делает её предпочтительной для задач с приоритетом точного совпадения с референсными ключевыми словами.

Контрастная архитектура с генерацией кандидатов и последующей фильтрацией ранкером показала наивысший BERTScore, что подтверждает эффективность подхода, где генерация обеспечивает разнообразие, а фильтрация – семантическую релевантность.

Примечательно, что компактная ru-mbart-summ с минимальными изменениями достигает BERTScore, сопоставимого с крупными инструктивными моделями (например, Mistral 7B в few-shot режиме), обеспечивая высокое качество при умеренных вычислительных затратах.

Среди ограничений – отсутствие этапа фильтрации в конфигурации с маскированием и зависимость качества от точности псевдоразметки. Дальнейшие исследования следует направить на интеграцию фильтрации во все генеративные конфигурации, улучшение качества псевдоразметки для снижения уровня шума, проведение кросс-доменных экспериментов (например, в медицине, химии), углублённую настройку инструктивных моделей для повышения F1 и ROUGE, а также тесты на статистическую значимость (t-критерий Стьюдента) и привлечение экспертной оценки моделей с близкими метриками.

В целом, разработанные методы демонстрируют значительный шаг вперед в решении задачи автоматической генерации ключевых слов для русского языка и могут быть использованы для улучшения систем индексации и поиска в научных электронных библиотеках.

Список литературы

- [Гендина и др., 2018] Гендина Н.И., Колкова Н.И. Методика формализованного аннотирования интернет-ресурсов // Научные и технические библиотеки. – 2018. – № 8. – С. 48-65. – doi: 10.33186/1027-3689-2018-8-48-65.
- [Гусев, 2023] Гусев И. Проект RuLM. [Электронный ресурс] // GitHub. 2023. – URL: <https://github.com/IlyaGusev/rulm> (дата обращения: 01.03.2025).
- [Дубовицкий, 2012] Дубовицкий Е.В. Процедурное программирование, операторы ветвления и циклов в системе компьютерной алгебры // Вестник российских университетов. Математика. – 2012. – Т. 17, № 2. – С. 598-602.
- [Чижов и др., 2015] Чижов И.А., Заец Н.П. Моделирование процесса теплопроводности многослойной конструкции для выполнения тепловизионного контроля [Электронный ресурс] // Приложение математики в экономических и технических исследованиях. – 2015. – № 5. – С. 139-145. – URL: <https://e.lanbook.com/journal/issue/295367> (дата обращения: 01.03.2025).
- [Bougouin et. al., 2013] Bougouin A., Boudin F., Daille B. TopicRank: Graph-Based Topic Ranking for Keyphrase Extraction // In: Proc. 6th International Joint Conference on Natural Language Processing (IJCNLP 2013), Nagoya, Japan, 2013. – P. 543-551.
- [Campos et. al., 2020] Campos R., Mangaravite V., Pasquali A., Jorge A.M., Nunes C., Jatowt A. YAKE! Keyword Extraction from Single Documents using Multiple Local Features // Information Sciences. – 2020. – Vol. 509. – P. 257-289. – doi: 10.1016/j.ins.2019.09.013.
- [Chen et. al., 2020] Chen T., Kornblith S., Norouzi M., Hinton G. A simple framework for contrastive learning of visual representations // In: Proc. 37th International Conference on Machine Learning (ICML 2020), Vienna, Austria, 2020. – P. 1597-1607. – doi: 10.1145/3701716.3715239.
- [Choi et. al., 2023] Choi M., Gwak C., Kim S., Kim S., Choo J. SimCKP: Simple Contrastive Learning of Keyphrase Representations // In: Proc. Findings of the Association for Computational Linguistics (EMNLP 2023), Singapore, Singapore, 2023. – P. 3003-3015. – doi: 10.18653/v1/2023.findings-emnlp.199.
- [Cicero, 2025] Cicero T. Forecasting the Scientific Production Volumes of G7 and BRICS Countries in a Comparative Analysis // Publications. – 2025. – Vol. 13(1). – Article 6. – doi: 10.3390/publications13010006.
- [Devlin et. al., 2019] Devlin J., Chang M., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // In: Proc. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Minneapolis, USA, 2019. – P. 4171-4186. – doi: 10.18653/v1/N19-1423.
- [Glazkova et. al., 2024] Glazkova A., Morozov D. Exploring Fine-tuned Generative Models for Keyphrase Selection: A Case Study for Russian // In: Proc. 26th International Conference on Data Analytics and Management in Data Intensive Domains (DAMDID 2024), Nizhny Novgorod, Russia, 2024. – URL: https://damdid2024.frccsc.ru/files/papers/DAMDID_2024_paper_11.pdf.
- [Glazkova et. al., 2025] Glazkova A., Morozov D., Garipov T. Key Algorithms for Keyphrase Generation: Instruction-Based LLMs for Russian Scientific Keyphrases // In: Proc. Analysis of Images, Social Networks and Texts (AIST 2024), Bishkek, Kyrgyzstan, 2025. – P. 107-119. – doi: 10.1007/978-3-031-88036-0_5.

- [Kang et. al., 2024] Kang B., Shin Y. Improving Low-Resource Keyphrase Generation through Unsupervised Title Phrase Generation // In: Proc. 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), Torino, Italia, 2024. – P. 8853-8865. – URL: <https://aclanthology.org/2024.lrec-main.775>.
- [Malkov et. al., 2018] Malkov Y.A., Yashunin D.A. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2020. – Vol. 42(4). – P. 824-836. – doi: 10.1109/TPAMI.2018.2889473.
- [Mihalcea et. al., 2004] Mihalcea R., Tarau P. TextRank: Bringing Order into Text // In: Proc. Empirical Methods in Natural Language Processing (EMNLP 2004), Barcelona, Spain, 2004. – P. 404-411. – URL: <https://aclanthology.org/W04-3252>.
- [Nikolich et. al., 2024] Nikolich A., Korolev K., Bratchikov S., Kiselev I., Shelmanov A. Vikhr: Constructing a State-of-the-art Bilingual Open-Source Instruction-Following Large Language Model for Russian // In: Proc. Fourth Workshop on Multilingual Representation Learning (MRL 2024), Miami, Florida, USA, 2024. – P. 189-199. – doi: 10.18653/v1/2024.mrl-1.15.
- [Raffel et. al., 2020] Raffel C., Shazeer N., Roberts A., Lee K., Narang S., Matena M., Zhou Y., Li W., Liu P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer // Journal of Machine Learning Research. – 2020. – Vol. 21(140). – P. 1-67.
- [Salton et. al., 1975] Salton G., Wong A., Yang C.S. A Vector Space Model for Automatic Indexing // Communications of the ACM. – 1975. – Vol. 18(11). – P. 613-620. – doi: 10.1145/361219.361220.
- [Tang et. al., 2021] Tang Y., Tran C., Li X., Chen P., Goyal N., Chaudhary V., Gu J., Fan A. Multilingual Translation from Denoising Pre-Training // In: Proc. Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021), Online, 2021. – P. 3450-3466. – doi: 10.18653/v1/2021.findings-acl.304.
- [Vaswani et. al., 2017] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I. Attention Is All You Need // In: Proc. 31st Conference on Neural Information Processing Systems (NIPS'17), Long Beach, California, USA, 2017. – P. 6000-6010. – doi: 10.5555/3295222.3295349.