Отчёт о проделанной работе
с использованием оборудования ИВЦ НГУ

1. Тема работы

Исследование методов автоматического реферирования

2. Состав коллектива

Батура Татьяна Викторовна, старший научный сотрудник ЛМСС ИСИ СО РАН, доцент кафедры СИ ФИТ НГУ, t.batura@g.nsu.ru

Научные интересы: автоматическая обработка текстов, компьютерная лингвистика, семантический анализ, извлечение информации из текстов

Березин Сергей Андреевич, студент ММФ НГУ, BDA&AI, 2 курс магистратуры, гр. 20155, email: s.berezin@g.nsu.ru

Ваулин Данил Николаевич, студент ФИТ НГУ, кафедра систем информатики, 4 курс, гр. 18204, email: d.vaulin@g.nsu.ru

Паульс Алексей Евгеньевич, студент ФИТ НГУ, кафедра систем информатики, 2 курс магистратуры, гр. 20222, email: a.pauls@g.nsu.ru

Шварц Никита Андреевич, аспирант ФИТ НГУ, кафедра систем информатики, 1 курс, email: shvarts.nikita.an@gmail.com

3. Аннотация

Исследован метод настройки префикса (prefix-tuning) для моделей BART и mBART. Эксперименты проводились на 3 датасетах: CNN/Daily Mail (новостные тексты на английском языке), Gazeta (новостные тексты на русском языке), RuSERRC (научные тексты на русском языке). Метод настройки префикса показал сравнительно хорошие результаты при использовании на русскоязычных текстах в условиях малого количества данных. Количество примеров при обучении может быть уменьшено в 50 раз без значительной потери качества генерируемых коротких рефератов.

Разработан новый метод абстрактной суммаризации (MNELM), учитывающий ключевые термины в научных текстах. Предлагаемая модель показывает хорошие результаты в метриках суммирования по сравнению с обычным подходом и быстрее сходится. Предварительное обучение помогает модели сосредоточиться на словах, специфичных для предметной области, тогда как базовая модель учится восстанавливать в основном общеупотребимые слова.

Предложен метод предобучения языковой модели с помощью семантической сегментации текста. Применение семантической сегментации немного ухудшает качество и заметно увеличивает размер саммари, но при этом справляется с главной задачей — позволяет обрабатывать длинные тексты полностью, в отличие от других существующих на сегодняшний день языковых моделей.

4. Информация о гранте

Грант РФФИ 19-07-01134, “Создание моделей, методов и программных средств анализа текстов на естественном языке для использования в интеллектуальных информационных системах” (2019–2021), руководитель Батура Татьяна Викторовна.

5. Научное содержание работы

5.1. Постановка задач

Описание работы 1:
Название (тема) работы: «Разработка программного модуля автоматического реферирования с применением метода настройки префикса»

Постановка задачи (что именно должно быть сделано, какие результаты должны быть получены):
Рассматривается классическая задача автоматического реферирования, когда на вход подается длинный текст, на выходе модель генерирует краткий реферат. Будет обучена языковая модель BART с применением техники “prefix-tuning” для работы с научными текстами. Предполагается, что данный метод позволит использовать значительно меньшее количество данных для обучения (по сравнению с другими методами) без потери качества. Для проверки данной гипотезы будет проведен ряд экспериментов в условиях ограниченного количества данных, в том числе для русского языка.

Описание работы 2:
Название (тема) работы: «Analysis of modern algorithms for named entity recognition and text summarization»

Постановка задачи (что именно должно быть сделано, какие результаты должны быть получены):
Будет разработан метод абстрактной суммаризации, учитывающий ключевые термины в тексте. Идея состоит в том, чтобы применить методы извлечения сущностей на этапе предварительного обучения модели для суммаризации. Это позволит добавить в языковую модель дополнительную информацию о ключевых терминах и тем самым достичь лучшего качества генерируемых рефератов.

Описание работы 3:
Название (тема) работы: «Исследование и реализация методов автоматического реферирования на основе нейронных сетей»

Постановка задачи (что именно должно быть сделано, какие результаты должны быть получены):
Будет предложена модификация метода Pegasus (Pre-training with Extracted Gap-sentences for Abstractive Summarization, пояснение в ссылке 3 обзора), позволяющая управлять процессом генерации реферата посредством заданного контекста. Эффективность предложенного метода будет проверена на больших датасетах научных текстов таких как ArXiv и PubMed.

5.2. Современное состояние проблемы (на момент начала работы)

1. Среди современных хорошо зарекомендовавших себя методов абстрактной суммаризации можно выделить BART [Lewis M., Liu Y., Goyal N., Ghazvininejad M., Mohamed A., Levy O., Stoyanov V., Zettlemoyer L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. Annual Meeting of the Association for Computational Linguistics, 2020, pp. 7871–7880.]. Эта сравнительно небольшая языковая модель показывает стабильно хорошее качество на разных датасетах (ROUGE-1 – 45.14, ROUGE-2 – 22.27, ROUGE-L – 37.25). Модель имеет стандартную архитектуру “Трансформер” (класса “кодировщик-декодировщик”) и может применяться не только для задачи генерации рефератов, но и для других задач обработки текстов, таких как машинный перевод, классификация текстов, генерация диалогов.

2. В задачах генерации текста техника настройки префикса (prefix-tuning) [Li X.L., Liang P. Prefix-tuning: Optimizing continuous prompts for generation. 2021. arXiv preprint arXiv:2101.00190.] – это более легкая альтернатива дообучению (fune-tuning), при которой большинство параметров предобученной модели остается "замороженной", а в процессе обучения оптимизируется только непрерывный специфический для конкретной задачи вектор небольшого размера, называемый "префиксом". Этот вектор воспринимается как "виртуальные токены". В отличие от техники prompt learning префикс состоит целиком из свободных параметров, которые не соответствуют реальным токенам, а в отличие от популярного дообучения (fine-tuning) происходит обновление не всех параметров “Трансформера”, оптимизируется только та часть, которая входит в префикс, что позволяет не хранить копии моделей для каждой задачи отдельно. Применение prefix-tuning позволяет достигать лучшего качества суммаризации при наличии сравнительно небольшого количества данных.

3. На сегодняшний день одним из лучших методов абстрактной суммаризации текстов является Pegasus [Zhang J., Zhao Y., Saleh M., Liu P. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In International Conference on Machine Learning. 2020. pp. 11328-11339]. Модель имеет архитектуру “Трансформер” с дополнительной задачей для предобучения. Обычно “Трансформеры” предобучаются

на некоторых задачах в режиме “без учителя”, т.е. на автоматически размеченных данных. В нашем случае это Masked Language Modelling и Gap Sentence Generation. Первая задача заключается в замене некоторых слов на специальные токены и обучении модели восстанавливать такие токены (заменять их на исходные). Для это модели нужно найти и запомнить зависимости между словами, в некотором роде – выучить семантику языка. Вторая задача очень похожа, но вместо отдельных слов на специальные токены заменяются целые предложения. Причем предложения не случайные, а “важные”. На этом уровне модель обучается восстанавливать такие “важные” предложения на основе других предложений. Такая техника позволяет “управлять” суммаризацией и делать больший акцент на ключевых словах, тем самым вынуждая языковую модель уделять больше внимания критически важным словам при обучении и впоследствии при работе выделять их в текстах.

4. Важное усовершенствование механизма внимания – «разреженное» внимание. Одна из проблем архитектур типа Трансформер заключается в том, что они имеют ограничение по длине входного текста. Архитектура BigBird [Zaheer M., Guruganesh G., Dubey K.A., Ainslie J., Alberti C., Ontanon S., Pham P., Ravula A., Wang Q., Yang L., Ahmed A. Big Bird: Transformers for Longer Sequences. In NeurIPS. 2020.] частично решает эту проблему – с помощью более стохастичного механизма внимания можно при том же размере и сравнимом качестве модели обрабатывать тексты на порядок длиннее, что критически важно в задачах суммаризации длинных текстов. На данный момент одновременное использование Pegasus и BigBird показывает один из лучших результатов в решении задачи суммаризации (ROUGE-1 – 46.63, ROUGE-2 – 19.02, ROUGE-L – 41.77 на датасете ArXiv).

5.3. Подробное описание работы, включая используемые алгоритмы

Работа 1

Метод настройки префикса (prefix-tuning) [Li X.L., Liang P. Prefix-Tuning: Optimizing Continuous Prompts for Generation. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021. pp. 4582–4597.] основан на гипотезе о том, что есть возможность управлять генерацией текста предобученной языковой модели с помощью подсказок без изменения параметров самой модели. Вместо дискретной оптимизации можно оптимизировать инструкции как непрерывные векторные представления слов (word embeddings). Под префиксом понимается простая полносвязная сеть (перцептрон), которая считается до подачи входной последовательности в модель.

Порядок обучения модели с применением метода настройки префикса:

1. Замораживаются веса предобученной языковой модели;
2. Инициализируются веса сети префикса;
3. Цикл обучения:
    a. Подсчет сети префикса (вход префикса одинаковый для каждой входной последовательности);

b. Выход префикса вместе с входной последовательностью подается на вход языковой последовательности;
c. По выходу языковой модели считается ошибка (перекрёстная энтропия);
d. Обратный проход с подсчетом градиентов;
e. Оптимизация весов префикса.

Таким образом, при обучении модели методом настройки префикса, веса языковой модели не изменяются, оптимизируются только веса сети префикса, в связи с этим данный метод подходит в условиях малого количества данных для обучения, так как количество оптимизируемых параметров в сети префикса значительно ниже, чем в сети языковой модели.

При использовании тонкой настройки оптимизируются миллионы параметров, поэтому для избегания переобучения требуется большое количество данных, вследствие чего модель может обучаться неделями. При использовании метода настройки префикса оптимизируются десятки-сотни тысяч параметров, поэтому это позволяет использовать меньшее количество данных, и модель не уйдет в переобучение. Однако качество в некоторых случаях может быть ниже.

Эксперименты проводились на 3 датасетах: CNN/Daily Mail, Gazeta, RuSERRC.

Первым этапом исследования были эксперименты, целью которых являлась проверка работоспособности метода. Данный метод разрабатывался для модели с архитектурой Transformer (encoder-decoder), реализованной в пакете hugging-face transformers в виде модели BART. Для удобства дальнейшей разработки метод настройки префикса был реализован для свежей на момент разработки версии пакета transformers (с 3.6.1 до 3.9.2). Для обучения модели был использован корпус данных CNN/Daily Mail версии 3.0.0. Обучение производилось на различном количестве данных: 2 тысячи примеров, 4 тысячи примеров. Количество циклов обучения: 10.

Вторым этапом были эксперименты на русскоязычных данных. Модель BART предобучена только на английском языке, что не подходит для решения поставленной задачи. Для решения проблемы были рассмотрены два варианта: предобучение Bart на русскоязычных данных; реализация метода для модели mBart.

Предобучение BART для русскоязычных данных является сложной задачей, требующей отдельного исследования, поэтому был выбран вариант реализации метода настройки префикса для модели mBART, так как она поддерживает русский язык. Архитектурно модели BART и mBART схожи, но их реализации отличаются: отличается порядок некоторых слоев, а также размерности этих слоев. В рамках данной работы метод настройки префикса был реализован для модели mBART и протестирован.

В качестве данных для обучения был выбран корпус данных Gazeta (новостные тексты на русском языке) версии 2.0. Обучение проводилось на разном количестве данных: 56 тысяч примеров, 5 тысяч примеров, 1 тысяча примеров, 500 примеров.

Третьим этапом были эксперименты на собранном и подготовленном корпусе научных статей на русском языке (RuSERRC).

Работа 2

Разработан новый метод абстрактной суммаризации, учитывающий ключевые термины в тексте. Основная идея состоит в том, чтобы на этапе предварительного обучения модели для суммаризации применить метод извлечения сущностей. Такой прием позволяет добавить в языковую модель дополнительную информацию о ключевых терминах и тем самым достичь лучшего качества генерируемых рефератов. Метод состоит из трех этапов: во-первых, модель

NER обучается на наборе данных, специфичном для предметной области; затем эта обученная модель NER используется для неконтролируемого предварительного обучения языковой модели в стиле MLM; наконец, предварительно обученная модель точно настроена для задачи суммаризации.

Для обучения модели распознавания именованных сущностей (Named Entity Recognition, NER) применялась модель RoBERTa [Liu Y. et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach. 2019. https://arxiv.org/abs/1907.11692]. После обучения в течение 7 эпох был получен показатель F1 0,76 на тестовом наборе данных.

В качестве базовой модели суммаризации рассматривалась модель BART. Она относится к классу Masked Language Model (MLM), т.к. использует операцию “маскирования”. Ранее обученную модель для NER мы применяем к научным текстам из датасета ArXiv и заменяем найденные именованные сущности (в нашем случае это научные термины) токенами [mask]. Таким образом, мы привлекаем внимание модели к терминам, а не просто к случайным словам, большинство из которых являются общеупотребимой лексикой. Предобучение выполнялось на 215912 научных текстах в течение 1 эпохи, learning_rate = $5 \cdot 10^{-5}$, linear_scheduler_with_gamma = 0.5 каждые 10000 шагов.

Работа 3

Популярные языковые модели ограничены максимальным размером входной последовательности, а значит длинные тексты невозможно обработать полностью без дополнительных действий. Предложен метод предобучения языковой модели с помощью семантической сегментации текста, который потенциально позволит не только справляться с длинными текстами, но может также повысить качество итоговой суммаризации. Для семантической сегментации использовалась модель HierarchicalBERT [Lukasik M. et al. Text segmentation by cross segment attention. Proceedings of EMNLP. 2020. pp. 4707–4716.].

В качестве основного BERT-а первого уровня (генератора эмбеддингов) используется уже предобученный SentenceRuBERT[1] — RuBERT, дообученный на русскоязычной части датасета XNLI [Conneau A. et al. XNLI: Evaluating cross-lingual sentence representations. arXiv preprint arXiv:1809.05053. 2018.]. Были также протестированы альтернативные варианты генерации эмбеддингов предложений на основе RuBERT — усреднение/взятие максимумов каждой компоненты векторов токенов, входящих в предложение, и вектор токена [CLS]. Сравнивать разные способы генерации эмбеддингов будем с помощью конечной задачи — семантической сегментации. Для начальной инициализации весов взяты веса RuBERT.

Параметры процедуры дообучения mBART для задачи сегментации и задачи суммаризации представлены в таблицах 1 и 2 соответственно.

Таблица 1. Параметры процедуры обучения модели сегментации

Параметр	Значение	Параметр	Значение
Количество эпох	12	Максимальная длина выходной последовательности	64

[1] https://huggingface.co/DeepPavlov/rubert-base-cased-sentence/

Размер батча для обучения	32	learning_rate	5e-5
Размер батча для валидации	32	adam_eps	1e-6
Максимальная длина входной последовательности	64	fp16	Да

Таблица 2. Параметры процедуры дообучения для задачи суммаризации

Параметр	Значение	Параметр	Значение
Количество эпох	2	attention_dropout	0.1
Размер батча для обучения	4	adam_eps	1e-6
Размер батча для валидации	8	fp16	Да
Максимальная длина входной последовательности	600	freeze_encoder	Да
Максимальная длина выходной последовательности	200	freeze_embeds	Да
learning_rate	3e-5	src_lang	ru_RU
warmup_steps	500	tgt_lang	ru_RU
label_smoothing	0.1	gradient_accumulation_steps	1
dropout	0.1		

5.4. Полученные результаты

Работа 1

Результаты эксперимента с моделью **BART на данных CNN/Daily Mail** (новостные тексты на английском языке) приведены в таблице 3. Для сравнения в таблице приведены эталонные результаты, взятые из статьи BART, где описана архитектура данной языковой модели. Обучение производилось на полном корпусе данных (порядка 270 тысяч примеров) и большом количестве циклов обучения.

Таблица 3. Результаты BART для CNN/Daily Mail

№ эксперимента	Rouge-1	Rouge-2	Rouge-L
1	*36,76*	15,69	25,98
2	31,02	*16,09*	*26,20*

3	**37,41**	**16,36**	**26,56**
Эталонные результаты	44,16	21,28	40,90

Эксперимент номер 1 проводился с 2 тысячами примеров для обучения, а также с случайным массивом чисел для входа префикса. Эксперимент номер 2 проводился также с 2 тысячами примеров, но в качестве входных данных для префикса выступает упорядоченный массив чисел. При случайном входном векторе результаты получились лучше. Третий эксперимент запускался со случайным входным вектором, так как этот результат дал лучшие метрики, и 4 тысячами примеров для обучения.

Результаты экспериментов с моделью **mBART на данных Gazeta** приведены в таблице 4. Эталонные результаты взяты из статьи, где описан процесс обучения модели mBART на корпусе данных Gazeta методом тонкой настройки (fine-tuning), модель обучалась на полном корпусе данных.

Таблица 4. Результаты mBART для Gazeta

№ эксперимента	Rouge-1	Rouge-2	Rouge-L
1	**21,20**	*6,97*	**20,50**
2	19,48	6,30	18,91
3	18,05	5,95	17,66
4	*20,82*	**7,31**	*19,95*
Эталонные результаты	32,1	14,2	27,9

Эксперимент номер 1 проводился с 56 тысячами примеров для обучения. Эксперимент номер 2 и 3 проводились с 5 тысячами примеров для обучения, во 2 эксперименте количество циклов для обучения – 30, а в 3 эксперименте – 30. Большее количество циклов обучения дает лучшие метрики. Эксперимент номер 4 проводился с 1 тысячью примеров для обучения и количество циклов обучения – 40, поэтому модель дала сравнительные метрики с большим количеством данных при обучении.

Результаты экспериментов с моделью **mBART на данных RuSERRC** приведены в таблице 5.

Таблица 5. Результаты mBART для RuSERRC

№ эксперимента	Rouge-1	Rouge-2	Rouge-L
1	5,25	2,80	5,01
2	4,95	1,61	4,93
3	15,12	6,53	14,39

Эксперимент номер 1 проводился с количеством циклов обучения – 20, а количество примеров для обучения – 1100. Эксперимент номер 2 проводился с количеством циклов обучения – 40 и 1100 примерами для обучения. Результаты первых двух экспериментов оказались низкими. Связано это с тем, что данные сильно отличаются друг от друга: эталонные рефераты разнятся по длине в зависимости от журнала и выпуска. В третьем эксперименте была выбрана часть данных, у которых длина реферата меньше средней длины, таких примеров оказалось большинство, кроме того был проведен ручной анализ данных и чистка от лишних символов, что также дало

положительный эффект в эксперименте номер 3. Последний эксперимент проводился с количеством циклов обучения – 40.

Работа 2

Для экспериментов использовались два набора данных (SCIERC и ArXiv). Результаты приведены в таблице 6.

Таблица 6. Сравнение результатов суммаризации для моделей MNELM и MLM на основе BART

	20k steps	25k steps
	MNELM-pretrained	MLM-pretrained
ROUGE-1 F1	**0.36**	0.35
ROUGE-1 precision	**0.51**	0.49
ROUGE-1 recall	0.29	0.29
ROUGE-2 F1	**0.13**	0.12
ROUGE-2 precision	**0.21**	0.19
ROUGE-2 recall	0.10	0.10
ROUGE-L F1	**0.32**	0.31
ROUGE-L precision	**0.45**	0.43
ROUGE-L recall	**0.26**	0.25

Видно, что метрики для предлагаемого метода с языковой моделью MNELM получились выше, чем для базовой модели. Однако во время обучения нашей модели мы заметили, что увеличение общих показателей для суммаризации текста приводит к уменьшению включения именованных сущностей. Возможно, причиной этого является ограниченная длина сгенерированного резюме, а значит, количество терминов, включенных в резюме, будет тоже ограничено. Поэтому во время обучения было необходимо найти оптимальное соотношение, в котором модель имеет высокие значения метрики ROUGE для суммаризации и вместе с тем высокие значения F1-score для задачи извлечения терминов. Было замечено, что модель MNELM сходится быстрее, чем базовая модель. Полученные результаты уступают недавно опубликованным современным моделям, таким как PRIMER [Xiao W. et al. PRIMERA: Pyramid-based Masked Sentence Pre-training for Multi-document Summarization. Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022. Vol. 1, pp. 5245–5263.] (ROUGE 1 = 47,6; ROUGE 2 = 20,8) или DeepPyramidon [Pietruszka M. et al. Sparsifying Transformer Models with Trainable Representation Pooling. Proceedings of the 60th Annual Meeting of

the Association for Computational Linguistics. 2022. Vol. 1, pp. 8616–8633.] (ROUGE 1 = 47,2; ROUGE 2 = 20), но их способность сохранять термины в генерируемом тексте не изучалась.

Работа 3

Эксперименты проводились на датасете Gazeta (новостные тексты на русском языке). Он разбит на 3 части: обучающую (52400 примеров), тестовую (5770 примеров) и валидационную (5265 примеров). Оценка влияния семантической сегментации на суммаризацию была проведена на выбранном ранее датасете для суммаризации. Каждый текст был разбит на сегменты, затем сегменты подавались на вход модели для суммаризации, результаты склеивались в один текст. Такие саммари выходили длиннее обычных, но содержали больше информации.

Было проведено сравнение результатов двух моделей: mBART-finetuned и mbart_ru_sum_gazeta без применения сегментации текстов и с применением сегментации текстов представлены в таблице 7.

Таблица 7. Сравнение результатов моделей для суммаризации с сегментацией и без

Модель	ROUGE-1 F1	ROUGE-2 F1	ROUGE-L F1	Средняя длина саммари в токенах
mBART-finetuned	18.75	4.76	18.06	110
mBART-finetuned + HierBERTSegmentator	**17.12**	**3.24**	**16.37**	**204**
mbart_ru_sum_gazeta	24.4	8.3	21.8	74
mbart_ru_sum_gazeta + HierBERTSegmentator	**22.4**	**6.9**	**20.5**	**189**

Сравнение показывает, что суммаризация на основе семантической сегментации генерирует более длинные саммари с немного меньшим качеством.

В Таблице 8 представлены результаты сравнения наших лучших моделей и известных моделей из открытых источников.

Таблица 8. Сравнение известных моделей для суммаризации и наших моделей

Модель	Датасет	ROUGE-1 F1	ROUGE-2 F1	ROUGE-L F1	Средняя длина саммари в токенах
mBART-finetuned	Gazeta	18.75	4.76	18.06	110
mBART-finetuned + HierBERTSegmentator	Gazeta	**17.12**	**3.24**	**16.37**	**204**

mBART-NER-random-finetuned	Gazeta	19.84	5.38	19.09	95
mbart_ru_sum_gazeta (наши измерения)	Gazeta	24.4	8.3	21.8	74
mbart_ru_sum_gazeta	Gazeta	32.4	14.3	28.0	92
rut5_base_sum_gazeta	Gazeta	32.2	14.4	28.1	82
rugpt3medium_sum_gazeta	Gazeta	26.2	7.7	21.7	61
PEGASUS_large (HugeNews)	XSum	47.21	24.56	39.25	-
BART	XSum	45.14	22.27	36.99	-
PEGASUS_large (HugeNews)	CNN/Daily Mail	44.17	21.47	41.11	-
BART	CNN/Daily Mail	44.16	21.28	40.90	-

Результаты наших моделей отличаются в худшую сторону по сравнению с моделями из открытых источников, обученными и протестированными на датасете Gazeta, при этом было проведено как обучение mBART на датасете Gazeta, так и повторное измерение результатов модели mbart_ru_sum_gazeta — в обоих случаях результаты отличаются от опубликованных в открытых источниках. Возможно, имеет место разная реализация метрик для оценки моделей или использование других гиперпараметров модели для генерации саммари. Также было проведено сравнение с результатами известных моделей на других датасетах (XSum и CNN/Daily Mail, оба датасета на английском языке). Видно, что наши модели заметно проигрывают в качестве.

Полученные результаты показывают, что работа по дообучению mBART с использованием NER имела смысл и принесла хорошие результаты — у модели mBART-NER-random-finetuned качество заметно лучше, чем у mBART, дообученного при тех же условиях. Но результаты как mBART-NER-random-finetuned, так и mBART-finetuned сильно отличаются от результатов, представленных в открытых источниках. Проблема может быть как в различной реализации метрик, так и в использовании различных параметров модели во время генерации саммари. Имеет смысл в будущем, для более точной оценки, попробовать другие метрики и другие параметры модели.

Применение семантической сегментации немного ухудшает качество и заметно увеличивает размер саммари, но при этом справляется с главной задачей — позволяет обрабатывать длинные тексты полностью, в отличие от других существующих языковых моделей.

6. Эффект от использования кластера в достижении целей работы

Обучение языковых моделей для задач автореферирования и извлечения сущностей из текстов связано с обработкой больших объемов информации и трудоемкими вычислениями. Решение этих задач возможно только с применением высокопроизводительных вычислительных систем (ВС).

7. Перечень публикаций, содержащих результаты работы

Ваулин Д.Н. Разработка программного модуля автоматического реферирования текстов с применением метода настройки префикса // Сборник тезисов МНСК-2022. (РИНЦ) (работа награждена дипломом 2 степени).