Оригинальная статья / Original article

https://doi.org/10.21869/2223-1560-2022-26-2-159-171

Увеличение производительности языковых моделей «трансформер» в информационных вопросно-ответных системах

Д. Т. Галеев [1] ✉, В. С. Панищев [1], Д. В. Титов [1]

[1] Юго-Западный государственный университет
ул. 50 лет Октября, д. 94, г. Курск 305040, Российская Федерация

✉ e-mail: ra3wvw@mail.ru

Резюме

Цель исследования. *Целью работы является увеличение производительности вопросно-ответных информационных систем на русском языке. Научная новизна работы состоит в увеличении производительности для модели RuBERT, которая была обучена для нахождения ответа на вопрос в тексте. Поскольку более производительная языковая модель позволяет обрабатывать большее количество запросов за то же самое время, результаты работы могут найти применение в различных информационных вопросно-ответных системах, для которых важна скорость отклика.*

Методы. *В настоящей работе используются методы обработки естественного языка, машинного обучения, уменьшения размера искусственных нейронных сетей. Языковая модель была настроена и обучена при помощи библиотек машинного обучения Torch и Onnxruntime. Оригинальная модель и набор данных для обучения были взяты в библиотеке Huggingface.*

Результаты. *В результате исследования была увеличена производительность работы языковой модели RuBERT при помощи методов уменьшения размера нейронных сетей, таких как дистилляция знаний и квантизация, а также при помощи экспорта модели в формат ONNX и её запуска в среде выполнения ONNX.*

Заключение*. В результате, модель, к которой одновременно были применены дистилляция знаний, квантизация и ONNX оптимизация, получила увеличение производительности в ~4.6 раза (с 66.57 до 404.46 запросов в минуту), при этом размер модели уменьшился в ~13 раз (с 676.29 Мб до 51.66 Мб). Обратной стороной полученной производительности стало ухудшение показателей EM (с 61.3 до 56.87) и F-мера (с 81.66 до 76.97).*

Ключевые слова: *машинное обучение; глубокое обучение; нейронные сети; обработка естественного языка; трансформер.*

Конфликт интересов: *Авторы декларируют отсутствие явных и потенциальных конфликтов интересов, связанных с публикацией настоящей статьи.*

Для цитирования*:* Галеев Д. Т., Панищев В. С., Титов Д. В. Увеличение производительности языковых моделей «трансформер» в информационных вопросно-ответных системах // Известия Юго-Западного государственного университета. 2022; 26(2): 159-171. https://doi.org/10.21869/2223-1560-2022-26-2-159-171.

Поступила в редакцию 19.05.2022 *Подписана в печать 05.07.2022* *Опубликована 29.07.2022*

Increased Performance of Transformers Language Models in Information Question and Response Systems

Denis T. Galeev [1] ✉, Vladimir S. Panishchev [1], Dmitry V. Titov [1]

[1] Southwest State University
50 Let Oktyabrya str. 94, Kursk 305040, Russian Federation

✉ e-mail: ra3wvw@mail.ru

Abstract

Purpose of research. *The purpose of this work is to increase the performance of question and response information systems in Russian. Scientific novelty of the work is to increase the performance for RuBERT model, which was trained to find the answer to the question in the text. As far as a more efficient language model allows more requests to be processed in the same time, the results of this work can be used in various information question and response systems for which response speed is important.*

Methods. *The present work uses methods of processing natural language, machine learning, reducing the size of artificial neural networks. The language model was configured and trained using Torch and Onnxruntime machine learning libraries. The original model and training dataset were taken from the Huggingface Library.*

Results. *As a result of the study, the performance of RuBERT language model was increased using methods to reduce the size of neural networks, such as distillation of knowledge and quantization, as well as by exporting the model to ONNX format and running it in ONNX runtime.*

Conclusion*. As a result, the model, to which knowledge distillation, quantization and ONNX optimization were simultaneously applied, received a performance increase of ~ 4.6 times (from 66.57 to 404.46 requests per minute), while the size of the model decreased ~ 13 times (from 676.29 MB to 51.66 MB). The downside of obtained performance was EM deterioration (from 61.3 to 56.87) and F-measure (from 81.66 to 76.97).*

Keywords: *machine learning; deep learning; neural networks; natural language processing; transformer.*

Conflict of interest. *The authors declare the absence of obvious and potential conflicts of interest related to the publication of this article.*

For citation: Galeev D. T., Panishchev V. S., Titov D. V. Increased Performance of Transformers Language Models in Information Question and Response Systems. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2022; 26(2): 159-171 (In Russ.). https://doi.org/ 10.21869/2223-1560-2022-26-2-159-171.

Received 19.05.2022 *Accepted 05.07.2022* *Published 29.07.2022*

***

Введение

Появление большого количества онлайн сервисов способствовало бурному росту команд для поддержки пользователей данных сервисов. Поддержка пользователей обычно представлена в виде чата на сайте сервиса, в котором пользователи могут задавать появившиеся у них вопросы или обсуждать проблемы, которые возникли у них во время взаимодействия с сервисом. Часто в этих чатах пользователи общаются непосредственно с людьми, которые были наняты для помощи пользователям. Однако сейчас можно заметить, как множество компаний работают над внедрением искусственного интеллекта во взаимодей-

ствие с клиентами [1], а также над замещением реальных людей в службе поддержки на различные модели машинного обучения [1]. Поскольку данные модели должны уметь находить нужную для пользователей информацию в различных базах знаний, то одной из важных задач в обработке естественного языка является задача нахождения ответа на вопрос в тексте.

Под поиском ответа на вопрос в тексте будем подразумевать наличие текста и вопроса к тексту, а также того, что система должна выбрать в качестве ответа на вопрос непрерывный фрагмент из данного текста. В англоязычной литературе данный тип задачи называется «Извлечение ответа на вопрос» (Extractive question answering). Для моделей, которые используют только кодировщик, задача сводится к тому, что необходимо ответить на два вопроса для каждого слова в тексте: «Является ли данное слово из текста началом ответа на заданный вопрос?», «Является ли данное слово из текста концом ответа на заданный вопрос?»:

$$-\sum_{k}(\log p_{start}(y_k^{start} \mid C, Q; \vartheta, w_{start}) + \\ + \log p_{end}(y_k^{end} \mid C, Q; \vartheta, w_{end})),$$

где y_k^{start} и y_k^{end} – позиции начала и конца ответа на вопрос для примера с индексом k, C – текст, Q – вопрос, Θ - параметры языковой модели, w_{start} и w_{end} – обучаемые параметры для предсказания позиции начала и конца ответа на вопрос. Вероятности p_{start} и p_{end} при использовании модели BERT определяются следующим образом:

$$p^{start}(y) = softmax\left(\left[\left\{w_{start}, BERT_0^N\right\}, \left\{w_{start}, BERT_1^N\right\}, ..., \left\{w_{start}, BERT_1^N\right\}, ..., \left\{w_{start}, BERT_L^N\right\}\right]\right),$$

$$p^{end}(y) = softmax\left(\left[\left\{w_{end}, BERT_0^N\right\}, \left\{w_{end}, BERT_1^N\right\}, ..., , ..., \left\{w_{end}, BERT_L^N\right\}\right]\right)$$

где softmax – функция софтмакс, $\{\cdot,\cdot\}$ – скалярное произведение, $BERT_i^N$ – выход с последнего слоя N для i-ого токена во входной последовательности, L – число токенов во входной последовательности.

Появление архитектуры «трансформер» [2], показавшей лучшие результаты по сравнению с присутствующими языковыми моделями, послужило началом для появления огромного количества моделей, использующих данную архитектуру. Модели использующие наработки из [2] могут быть полными «трансформерами» [3-6], могли использовать только логику кодировщика [7-10] или использовать только логику декодера [11-15].

Недостатком использования архитектуры «трансформер» является наличие большого количества параметров у модели и, следовательно, большая сложность вычисления результата. Это может быть критично для крупных онлайн сервисов, поскольку они имеют большое количество клиентов, следовательно большое количество запросов в службу поддержки. Время отклика должно быть максимально быстрым. Поэтому еще одной важной задачей в обработке естественного языка является увеличение

производительности языковых моделей при сохранении эффективности их работы.

Материалы и методы

Методы уменьшения размера нейронных сетей. Одним из подходов к увеличению скорости работы модели является уменьшение размера модели. С уменьшением размера модели происходит уменьшение количества операций, которые необходимо выполнить модели чтобы получить результат.

Основными методами уменьшения размера модели являются квантизация (quantization), прунинг (pruning) и дистилляция знаний (knowledge distillation).

Квантизация означает уменьшение численной точности весов модели. Например, изначально вес модели содержал значение типа float, то после квантизации данный вес модели будет содержать значение типа integer.

Прунинг предполагает фильтрацию частей модели, чтобы сделать ее меньше и быстрее. Популярным методом является прунинг весов модели. Данный метод удаляет веса, значения которых близки к 0. Это позволяет избавить модель от весов, значимость которых для предсказывания невелика.

Дистилляция знаний заключается в том, что вначале обучается большая модель, а затем на основе предсказаний данной модели-учителя обучается более легковесная модель-ученик [16]. Данный подход показал хорошие результаты [17]. Модели-ученики могут совсем незначительно уступать в результатах моделям-учителям, но при этом быть в несколько раз меньше и работать быстрее. По этой причине появилось большое количество дистиллированных версий популярных сетей (таких как tinyBERT, distilBERT, distilRoBERTa [18]).

Экспорт модели в формат ONNX. ONNX (Open Neural Network Exchange) [19] представляет собой открытый формат, созданный для представления моделей машинного обучения. ONNX определяет общий набор строительных блоков моделей машинного обучения. Модели в формате ONNX можно запускать в специальных средах выполнения ONNX. Данные среды выполнения содержат множество оптимизаций на аппаратном уровне. В результате модели в данной среде работают быстрее. Помимо этого, среды выполнения ONNX позволяют производить оптимизацию моделей: удалять избыточные операции, объединять некоторые операции вместе и т. д.

Набор данных для обучения сети. Основным набором данных для тренировки вопросно-ответных систем на русском языке является SberQuAD [20]. SberQuAD содержит 45328 тренировочных наборов из текста, вопроса, и ответа, 5036 валидационных наборов и 23936 проверочных наборов. К сожалению, ответы на проверочные данные не представлены публично, поэтому результаты работы модели сравниваются на валидационном наборе. Конкретный экземпляр набора данных был взят из

библиотеки Huggingface. Каждый экземпляр данных содержит следующие поля:

– context. В данном поле находится текст, к которому будет задан вопрос.

– question. В данном поле содержится вопрос к тексту из поля context.

– answers. В данном поле находится ответ на вопрос из поля question к тексту из поля context. В данном поле присутствуют два поля answer_start и text. Поле answer_start содержит индекс начала ответа на вопрос, а поле text содержит полный текст ответа.

Пример тренировочных данных представлен на рис. 1.

```
{
 'id': 15008,
 'title': 'SberChallenge',
 'context': 'Корпоративный сайт - содержит полную информацию о компании-владельце, услугах/
    продукции, событиях в жизни компании. Отличается от сайта-визитки и представительского сайта
    полнотой представленной информации, зачастую содержит различные функциональные инструменты
    для работы с контентом (поиск и фильтры, календари событий, фотогалереи, корпоративные блоги,
    форумы). Может быть интегрирован с внутренними информационными системами компании-владельца (
    КИС, CRM, бухгалтерскими системами). Может содержать закрытые разделы для тех или иных групп
    пользователей - сотрудников, дилеров, контрагентов и пр.',
 'question': 'С чем может быть интегрирован корпоративный сайт?',
 'answers': {
    'text': ['с внутренними информационными системами компании-владельца (КИС, CRM, бухгалтерскими
        системами'],
    'answer_start': [389]
  }
}
```

Рис. 1. Пример тренировочных данных

Fig. 1. Example of training data

Показатели. Основными показателями качества работы вопросно-ответных моделей являются EM и F-мера [20].

Exact match (EM) — точное совпадение, доля ответов системы, которые полностью совпадают с одним из правильных ответов с точностью до пунктуации и регистра.

F-мера – представляет собой совместную оценку полноты и точности. Данный показатель вычисляется по следующей формуле:

$$F = 2 \cdot \frac{Точность \cdot Полнота}{Точность + Полнота}.$$

Полнота (recall) вычисляется по следующей формуле:

$$Полнота = \frac{TP}{TP + FN}.$$

Точность (precision) вычисляется по следующей формуле:

$$Точность = \frac{TP}{TP + FP},$$

где TP – Истинноположительные предсказания модели (модель правильно отнесла объект к классу); FN – Ложноотрицательные предсказания модели (модель неправильно не отнесла объект к классу); FP – Ложноположительные предсказания модели (модель неправильно отнесла объект к классу).

Выбор основной языковой модели. Выбор модели был осуществлён на сайте библиотеки Huggingface. В качестве основной модели-учителя было принято

решение выбрать модель ruBert-base [21]. Данная модель показала одни из лучших результатов в обработке данных из набора SberQuAD.

Дистилляция. В качестве модели-ученика для дистилляции была выбрана Geotrend/distilbert-base-ru-cased [22]. Основным её плюсом является малый размер, который составляет 205.62 Мб, против 676.29 Мб у ruBert-base.

Проведение дистилляции с моделью-учителем ruBert-base и моделью-учеником Geotrend/distilbert-base-ru-cased на наборе данных SberQuAD заняло 4 часа 15 минут.

Квантизация. Языковые модели «трансформер» обычно имеют веса в формате чисел с плавающей запятой с размером 32 бит. В проводимых экспериментах формат весов моделей был изменен на целочисленный с размером 8 бит. Данная операция позволяет увеличить скорость вычислений предсказаний модели, а также уменьшить вес самой модели. Также стоит отметить, что во всех экспериментах применялась динамическая квантизация.

ONNX. Модели из библиотеки Huggingface поддерживают экспорт в формат ONNX. Данная операция позволяет запускать данные модели в оптимизированных средах выполнения. Также данные среды позволяют оптимизировать некоторые операции модели. Далее в статье запуск модели в ONNX среде выполнения и применение оптимизаций операций к этой модели будет называться ONNX оптимизация.

Результаты и их обсуждение

Описание стенда для обучения нейронной сети. Обучение и дистилляция моделей проводились на сервисе Google Colab. Данный сервис предоставляет во временное (12 часов) бесплатное пользование компьютеры с производительными видеокартами (уровня NVIDIA Tesla K80).

Эксперименты проводились на компьютере MacBook Pro (Retina, 15-inch, Mid 2015) с ЦП 2,2 GHz Quad-Core Intel Core i7 и ОЗУ 16 GB 1600 MHz DDR3.

Проведение экспериментов. За основу была взята модель ruBert-base. После дистилляции появилась еще обученная модель Geotrend/distilbert-base-ru-cased. Для каждой из этих моделей было решено провести следующие операции:

Квантизацию.

ONNX оптимизацию.

Квантизацию и ONNX оптимизацию вместе.

Для каждой модели эксперимента были получены:

Показатели EM (Exact match) и F-мера

Полное время обработки валидационного (5389 вопросов) набора данных из SberQuAD (минуты).

Производительность (количество обработанных запросов в минуту).

Размер модели (Мб).

Результаты работы моделей представлены в табл. 1, рис. 2 и рис. 3.

Обсуждение результатов экспериментов. Дистилляция знаний из модели

ruBert-base в модель Geotrend/distilbert-base-ru-cased позволила почти в 1.7 раза сократить время обработки (с ~81 минуты до 47 минут), а также уменьшить размер модели в ~3.3 раза (с 676.29 до 205.62 Мб).

Таблица 1. Результаты работы моделей

Table 1. Model Results

Версия модели / Model version	EM	F-1	Размер модели (Мб) / Model size	Время обработки валидационного набора данных (мин) / Treatment Time of Validation Data Set (min)	Количество запросов в валидационном наборе данных / Number of queries in validation dataset	Производительность (запросы в минуту) / Productivity (requests per minute)
ruBert-base	61.3	81.66	676.29	80.95	5389	66.57
ruBert-base + квантизация	59.21	80.54	433.34	36.73		146.71
ruBert-base + ONNX оптимизация	61.3	81.66	676.31	30		179.63
ruBert-base + ONNX оптимизация + квантизация	61.51	81.96	169.55	28		192.46
distilbert-base-ru-cased	58.57	78.42	205.62	47		114.65
distilbert-base-ru-cased + квантизация	52.28	72.62	84.15	20.15		267.44
distilbert-base-ru-cased + ONNX оптимизация	58.57	78.42	205.63	18		299.38
distilbert-base-ru-cased + квантизация + ONNX оптимизация	56.87	76.97	51.66	17.7		304.46

Квантизация для обеих моделей позволила чуть больше чем в 2 раза уменьшить время обработки (с ~81 минуты до ~37 минут для ruBert-base и с 47 минут до ~21 минуты для Geotrend/distilbert-base-ru-cased), но в обоих случаях пострадали показатели EM и F-мера. Для модели ruBert-base падение

метрик является незначительным (с EM 61.3 и F-мера 81.66 до EM 59.21 и F-мера 80.54). В случае с моделью Geotrend/distilbert-base-ru-cased наблюдается максимальное падение показателей среди всех методов оптимизации (с EM 58.57 и F-мера 78.42 до EM 52.28 и F-мера 72.62). После квантизации модель ruBert-base стала на 35% меньше (с 676.29 Мб до 433.34 Мб), а модель Geotrend/distilbert-base-ru-cased стала на 59% меньше (с 205.62 Мб до 84.15 Мб).Применение ONNX оптимизации позволило сохранить показатели EM и F-мера, а также размер модели неизменными, но уменьшить время обработки более чем в 2 раза. Для модели ruBert-base время обработки сократилось на 62% (с ~81 минуты до 30 минут), а для модели Geotrend/distilbert-base-ru-cased - на 61% (с 47 минут до ~20 минут).

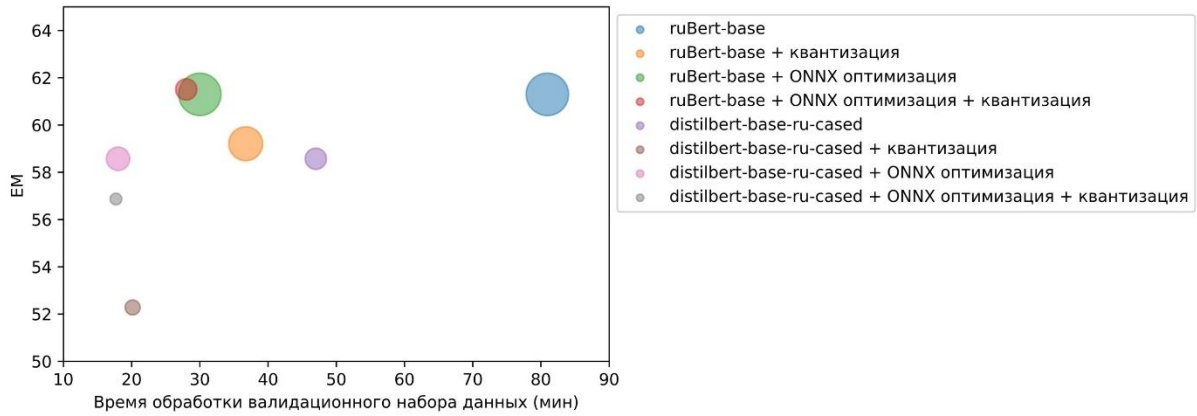

Рис. 2. Показатель EM и время обработки для проведенных экспериментов

Fig. 2. EM metric and processing time for the experiments performed

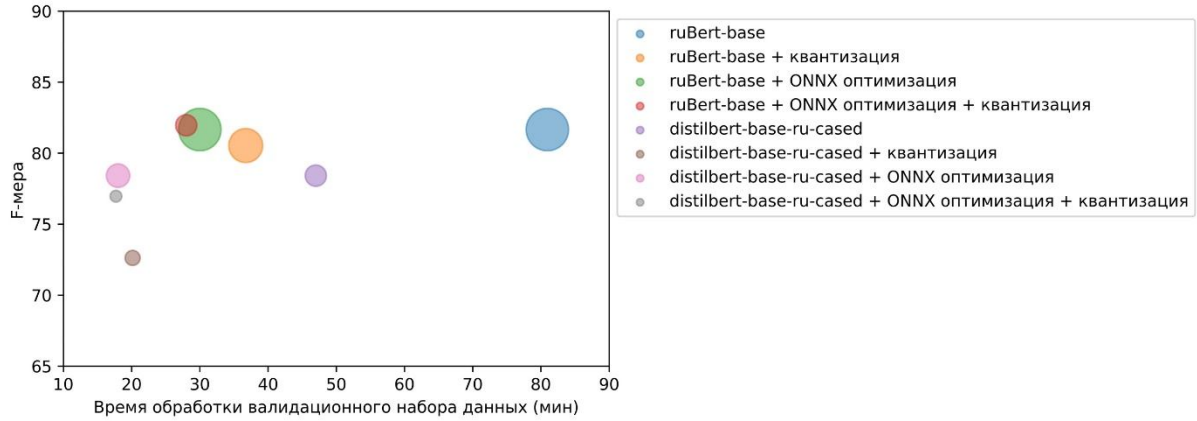

Рис. 3. Показатель F-мера и время обработки для проведенных экспериментов

Fig. 3. F_1 metric and processing time for experiments performed

Добавление квантизации к ONNX оптимизации незначительно снизило время обработки, но позволило еще сильнее уменьшить размер модели. После ONNX оптимизации с квантизацией модель ruBert-base стала на 75% меньше

(с 676.29 Мб до 169.55 Мб), а модель Geotrend/distilbert-base-ru-cased - на 74% меньше (с 205.62 Мб до 51.66 Мб). Однако, стоит отметить, что если в случае с моделью ruBert-base показатели EM и F-мера немного выросли (с EM 61.3 и F-мера 81.66 до EM 61.51 и F-мера 81.96), то в случае с моделью данные показатели ухудшились (с EM 58.57 и F-мера 78.42 до EM 56.87 и F-мера 76.97).

Худшие результаты по времени обработки валидационного набора данных показала оригинальная модель ruBert-base (~91 минута). Лучшие результаты показала модель Geotrend/distilbert-base-ru-cased в эксперименте с ONNX оптимизацией (18 минут) и в эксперименте с ONNX оптимизацией и квантизацией (~18 минут).

Худшие результаты относительно размера модели также показала оригинальная модель ruBert-base (676.29 Мб). Лучшие результаты показала модель distilbert-base-ru-cased в эксперименте с квантизацией (84.15 Мб) и в эксперименте с ONNX оптимизацией и квантизацией (51.66 Мб).

Модель distilbert-base-ru-cased из эксперимента с ONNX оптимизацией и квантизацией имеет наименьший размер (51.66 Мб) и наименьшее время выполнения (~18 минут) среди всех других комбинаций оптимизаций. Данная модель смогла обработать валидационный набор данных на 78% быстрее по сравнению с оригинальной моделью (ruBert-base). Но данная модель имеет одни из худших метрик EM 56.87 и F-мера 76.97. Однако стоит отметить, что, учитывая высокую скорость работы данное ухудшение качества модели может быть приемлемым.

Выводы

В результате, модель, к которой одновременно были применены дистилляция знаний, квантизация и ONNX оптимизация, получила увеличение производительности в ~4.6 раза (с 66.57 до 404.46 запросов в минуту), при этом размер модели уменьшился в ~13 раз (с 676.29 Мб до 51.66 Мб). Обратной стороной полученной производительности стало ухудшение показателей EM (с 61.3 до 56.87) и F-мера (с 81.66 до 76.97).

Список литературы

1. Рябинов А.В., Уздяев М.Ю., Ватаманюк И.В. Применение многозадачного глубокого обучения в задаче распознавания эмоций в речи // Известия Юго-Западного государственного университета. 2021; 25(1): 82-109. https://doi.org/10.21869/2223-1560-2021-25-1-82-109

2. Vaswani A. et al. Attention is all you need // Advances in Neural Information Processing Systems 2017-December, 5999–6009 (Neural information processing systems foundation, 2017).

3. Lewis M. et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. in 7871–7880 (Association for Computational Linguistics (ACL), 2020).

4. Raffel C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer // Journal of Machine Learning Research 21, (2020).

5. Zhang J., Zhao Y., Saleh M., Liu P. J. PEGASUS: Pre-Training with extracted gap-sentences for abstractive summarization // 37th International Conference on Machine Learning, ICML 2020 PartF168147-15, 11265–11276 (International Machine Learning Society (IMLS), 2020).

6. Qi W. et al. ProphetNet: Predicting future n-gram for sequence-to-sequence pre-training // Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020 2401–2410 (Association for Computational Linguistics (ACL), 2020).

7. Devlin J., Chang M. W., Lee K., Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding // NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference 1, 4171–4186 (Association for Computational Linguistics (ACL), 2019).

8. Lan Z. et al. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. in ICLR (OpenReview.net, 2020).

9. Liu Y. et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach. CoRR abs/1907.11692, (2019).

10. Clark K., Luong M.-T., Le Q. V., Manning, C. D. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. CoRR abs/2003.10555, (2020).

11. Dai Z. et al. Transformer-XL: Attentive language models beyond a fixed-length context // ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference 2978–2988 (Association for Computational Linguistics (ACL), 2020).

12. Keskar N. S., McCann B., Varshney L. R., Xiong C., Socher R. CTRL: A Conditional Transformer Language Model for Controllable Generation. CoRR abs/1909.05858, (2019).

13. Radford A., Narasimhan K., Salimans T., Sutskever I. (OpenAI Transformer): Improving Language Understanding by Generative Pre-Training. OpenAI 1–10 (2018).

14. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, I. S. Language Models are Unsupervised Multitask Learners. OpenAI Blog 1, 1–7 (2020).

15. Brown T. B. et al. Language models are few-shot learners // Advances in Neural Information Processing Systems 2020-December, (Neural information processing systems foundation, 2020).

16. Hahn S., Choi H. Self-knowledge distillation in natural language processing // International Conference Recent Advances in Natural Language Processing, RANLP 2019-September, 423–430 (Incoma Ltd, 2019).

17. Sanh V., Debut L., Chaumond J., Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. CoRR abs/1910.01108, (2019).

18. Li T., El Mesbahi Y., Kobyzev I., Rashid A., Mahmud A., Anchuri N., Hajimolahoseini H., Liu Y., Rezagholizadeh M. A Short Study on Compressing Decoder-Based Language Models. CoRR abs/2110.08460 (2021).

19. Le T. D. et al. Compiling ONNX Neural Network Models Using MLIR. CoRR abs/2008.08272, (2020).

20. Efimov P., Chertok A., Boytsov L., Braslavski, P. SberQuAD – Russian Reading Comprehension Dataset: Description and Analysis // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 12260 LNCS, 3–15 (Springer Science and Business Media Deutschland GmbH, 2020).

21. Kuratov Y., Arkhipov M. Adaptation of deep bidirectional multilingual transformers for Russian language // Komp'juternaja Lingvistika i Intellektual'nye Tehnologii 2019-May, 333–339 (ABBYY PRODUCTION LLC, 2019).

22. Abdaoui A., Pradel C., Sigel G. Load What You Need: Smaller Versions of Mutilingual BERT. in 119–123 (Association for Computational Linguistics (ACL), 2020).

References

1. Ryabinov A.V., Uzdiaev M.Yu., Vatamaniuk I.V. [Applying Multitask Deep Learning to Emotion Recognition in Speech]. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University* 2021;25(1):82-109. (In Russ.) https://doi.org/10.21869/2223-1560-2021-25-1-82-109.

2. Vaswani A. et al. Attention is all you need. *Advances in Neural Information Processing Systems 2017-December*, 5999–6009 (Neural information processing systems foundation, 2017).

3. Lewis M. et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. in 7871–7880 (Association for Computational Linguistics (ACL), 2020).

4. Raffel C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21, (2020).

5. Zhang J., Zhao Y., Saleh M., Liu P. J. PEGASUS: Pre-Training with extracted gap-sentences for abstractive summarization. *37th International Conference on Machine Learning*, ICML 2020 PartF168147-15, 11265–11276 (International Machine Learning Society (IMLS), 2020).

6. Qi W. et al. ProphetNet: Predicting future n-gram for sequence-to-sequence pre-training. *Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020;* 2401–2410 (Association for Computational Linguistics (ACL), 2020).

7. Devlin J., Chang M. W., Lee K., Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1, 4171–4186 (Association for Computational Linguistics (ACL), 2019).

8. Lan Z. et al. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. in ICLR (OpenReview.net, 2020).

9. Liu Y. et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach. CoRR abs/1907.11692, (2019).

10. Clark K., Luong M.-T., Le Q. V., Manning C. D. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. CoRR abs/2003.10555, (2020).

11. Dai Z. et al. Transformer-XL: Attentive language models beyond a fixed-length context. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 2978–2988 (Association for Computational Linguistics (ACL), 2020).

12. Keskar N. S., McCann B., Varshney L. R., Xiong C., Socher R. CTRL: A Conditional Transformer Language Model for Controllable Generation. CoRR abs/1909.05858, (2019).

13. Radford A., Narasimhan K., Salimans T., Sutskever I. (OpenAI Transformer): Improving Language Understanding by Generative Pre-Training. OpenAI 1–10 (2018).

14. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, I. S. Language Models are Unsupervised Multitask Learners. OpenAI Blog 1, 1–7 (2020).

15. Brown T. B. et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems 2020-December*, (Neural information processing systems foundation, 2020).

16. Hahn S., Choi H. Self-knowledge distillation in natural language processing. *International Conference Recent Advances in Natural Language Processing, RANLP 2019-*September, 423–430 (Incoma Ltd, 2019).

17. Sanh V., Debut L., Chaumond J., Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. CoRR abs/1910.01108, (2019).

18. Li T., El Mesbahi Y., Kobyzev I., Rashid A., Mahmud A., Anchuri N., Hajimolahoseini H., Liu Y., Rezagholizadeh M. A Short Study on Compressing Decoder-Based Language Models. CoRR abs/2110.08460 (2021).

19. Le T. D. et al. Compiling ONNX Neural Network Models Using MLIR. CoRR abs/2008.08272, (2020).

20. Efimov P., Chertok A., Boytsov L., Braslavski P. SberQuAD – Russian Reading Comprehension Dataset: Description and Analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 12260* LNCS, 3–15 (Springer Science and Business Media Deutschland GmbH, 2020).

21. Kuratov Y., Arkhipov M. Adaptation of deep bidirectional multilingual transformers for Russian language. *Komp'juternaja Lingvistika i Intellektual'nye Tehnologii* 2019-May, 333–339 (ABBYY PRODUCTION LLC, 2019).

22. Abdaoui A., Pradel C., Sigel G. Load What You Need: Smaller Versions of Mutilingual BERT. in 119–123 (Association for Computational Linguistics (ACL), 2020).

Информация об авторах / Information about the Authors

Галеев Денис Талгатович, аспирант, Юго-Западный государственный университет», г. Курск, Российская Федерация, e-mail: ra3wvw@mail.ru, ORCID: https://orcid.org/0000-0002-7677-1800

Панищев Владимир Славиевич, кандидат технических наук, Юго-Западный государственный университет г. Курск, Российская Федерация, e-mail: gskunk@yandex.ru, ORCID: https://orcid.org/0000-0003-1772-7663

Титов Дмитрий Витальевич, доктор технических наук, доцент, Юго-Западный государственный университет, г. Курск, Российская Федерация, e-mail: titov.swsu@gmail.com

Denis T. Galeev, Post-Graduate Student, Southwest State University, Kursk, Russian Federation, e-mail: ra3wvw@mail.ru, ORCID: https://orcid.org/0000-0002-7677-1800

Vladimir S. Panishchev, Cand. of Sci. (Engineering), Southwest State University, Kursk, Russian Federation, e-mail: gskunk@yandex.ru, ORCID: https://orcid.org/ 0000-0003-1772-7663

Dmitry V. Titov, Dr. of Sci. (Engineering), Associate Professor, Southwest State University, Kursk, Russian Federation, e-mail: titov.swsu@gmail.com