ISSN 1816-0301 (Print)
ISSN 2617-6963 (Online)

УДК 004.8
https://doi.org/10.37661/1816-0301-2020-17-4-61-72

Поступила в редакцию 17.07.2020
Received 17.07.2020

Принята к публикации 21.09.2020
Accepted 21.09.2020

Моделирование языка и двунаправленные представления кодировщиков: обзор ключевых технологий

Д. И. Качков

Белорусский государственный университет, Минск, Беларусь
E-mail: dmitriydikanskiy@gmail.com

Аннотация. Представлен очерк развития технологий обработки естественного языка, которые легли в основу BERT (Bidirectional Encoder Representations from Transformers) − языковой модели от компании Google, демонстрирующей высокие результаты на целом классе задач, связанных с пониманием естественного языка. Две ключевые идеи, реализованные в BERT, – это перенос знаний и механизм внимания. Модель предобучена решению нескольких задач на обширном корпусе неразмеченных данных и может применять обнаруженные языковые закономерности для эффективного дообучения под конкретную проблему обработки текста. Использованная архитектура Transformer основана на внимании, т. е. предполагает оценку взаимосвязей между токенами входных данных. В статье отмечены сильные и слабые стороны BERT и направления дальнейшего усовершенствования модели.

Ключевые слова: информатика, информационные технологии, языковые модели, обработка естественного языка, механизм внимания, архитектура Transformer, модель BERT

Для цитирования. Качков, Д. И. Моделирование языка и двунаправленные представления кодировщиков: обзор ключевых технологий / Д. И. Качков // Информатика. – 2020. − Т. 17, № 4. – С. 61–72. https://doi.org/10.37661/1816-0301-2020-17-4-61-72

Language modeling and bidirectional coders representations: an overview of key technologies

Dzmitry I. Kachkou

Belarusian State University, Minsk, Belarus
E-mail: dmitriydikanskiy@gmail.com

Abstract. The article is an essay on the development of technologies for natural language processing, which formed the basis of BERT (Bidirectional Encoder Representations from Transformers), a language model from Google, showing high results on the whole class of problems associated with the understanding of natural language. Two key ideas implemented in BERT are knowledge transfer and attention mechanism. The model is designed to solve two problems on a large unlabeled data set and can reuse the identified language patterns for effective learning for a specific text processing problem. Architecture Transformer is based on the attention mechanism, i.e. it involves evaluation of relationships between input data tokens. In addition, the article notes strengths and weaknesses of BERT and the directions for further model improvement.

Keywords: informatics, information technology, language models, natural language processing, attention mechanism, transformer architecture, model BERT.

For citation. Kachkou D. I. Language modeling and bidirectional coders representations: an overview of key technologies. *Informatics*, 2020, vol. 17, no. 4, pp. 61−72 (in Russian). https://doi.org/10.37661/1816-0301-2020-17-4-61-72

Введение. Естественный язык является основным средством коммуникации для человека, и проблема его автоматической обработки – одно из актуальных направлений научных исследований в области искусственного интеллекта. Системы, способные обрабатывать естественный язык, позволяют решать широкий спектр задач, в том числе извлечения информации об окружающем мире из огромных массивов текстов, автоматического перевода с одного языка на другой без участия человека, разработки эффективных интерфейсов взаимодействия между человеком и компьютером и т. д.

Среди различных инструментов обработки естественного языка можно выделить класс средств, базирующихся на языковых моделях – системах, которые были предобучены на больших корпусах текстов рассчитывать распределение вероятности появления тех или иных токенов (слов) на заданных позициях в предложении. Как показали исследования, нейронные языковые модели могут эффективно дообучаться под широкий спектр задач, связанных с интерпретацией информации на естественном языке. По состоянию на май 2020 г. большинство систем, демонстрирующих наилучший результат при решении конкретных задач обработки естественного языка, так или иначе основаны на языковой модели BERT, разработанной в компании Google.

Механизм внимания. В 2014 г. был предложен механизм seq2seq (sequence-to-sequence), идея которого описана в работе [1]. Решение состоит из двух рекуррентных нейронных сетей, представляющих собой кодировщик (encoder) и декодировщик (decoder) соответственно. Задача кодировщика заключается в том, чтобы поставить в соответствие исходной последовательности переменной длины некоторый вектор состояния фиксированной размерности. Декодировщик, в свою очередь, разворачивает этот вектор в целевую последовательность, не имеющую фиксированной длины. Обучение проходит одновременно для обеих сетей: ставится задача максимизировать условную вероятность целевой последовательности по заданной исходной последовательности.

В терминах задачи о сопоставлении двух последовательностей могут быть описаны различные проблемы естественной обработки языка, например автоматического перевода [1, 2] или генерация ответных реплик в диалоге [3]. Более того, источником вектора состояния может быть кодировщик любой природы, поэтому подобная архитектура оказалась применима и в таких задачах, как генерация подписи для картинки [4], построение комментария к изменениям в исходном коде программ [5] и составление связанного текста на основе данных таблицы [6].

Использование рекуррентных нейронных сетей обусловлено их способностью сохранять некоторое скрытое состояние, которое характеризует происходящее во входной последовательности. Таким образом обеспечивается обработка предложений произвольной длины [7, с. 232–233].

Уже в работе [8] отмечается основной недостаток подобной архитектуры – необходимость упаковывать произвольное количество информации, содержащейся в исходной последовательности, в вектор фиксированной длины. С увеличением длины входного предложения эффективность построенной модели снижается. Авторы предложили усовершенствовать систему, внедрив механизм внимания. Идея подобного механизма вдохновлена человеческой природой: известно, что при взгляде на рисунок человек не анализирует его полностью во всех подробностях, а фокусируется на отдельных участках, которые, как ему кажется, несут наибольшее количество полезной информации [7, с. 331–332]. Попытки имитировать внимание предпринимались в машинном обучении при анализе изображений, определение значимых участков картинки позволяло сэкономить время на обработке незначимых, периферийных фрагментов [7, с. 333]. Эта концепция переносится и на задачи обработки естественного языка: от механизма внимания требуется оценить, какие составляющие длинного исходного предложения являются наиболее существенными при решении текущей задачи.

Предложенный механизм был реализован следующим образом. Кодировщик передает декодировщику не единый вектор состояния, а множество векторов-аннотаций, построенных с помощью двунаправленной рекуррентной нейронной сети [9] для каждого слова исходного предложения. Декодировщик в ходе своей работы вычисляет актуальный вектор контекста как взвешенную сумму полученных от кодировщика векторов. Используемые при суммировании

веса – аналог внимания: они вычисляются на основе текущего внутреннего состояния рекуррентной нейронной сети декодировщика для каждого вектора-аннотации и определяют, насколько значимо соответствующее этому вектору слово на данном этапе перевода. Предложенная модель показала значительный прирост в качестве перевода длинных предложений.

Исследование нейронного машинного перевода с использованием механизма внимания проводилось в ряде работ. Например, в работе [10] рассмотрены два архитектурных решения. Первое основано на подходе, представленном в [8], и отличается от него лишь деталями. Второе решение, названное авторами локальным вниманием (local attention), предполагает построение текущего вектора контекста, основываясь на подмножестве наиболее значимых векторов-аннотаций. В качестве наиболее значимых предполагается рассматривать векторы-аннотации, соответствующие словам $\left[p_t - D, p_t + D\right]$, где p_t – центральный элемент окна, выбираемый с помощью обученного компонента системы, а D – фиксированная ширина окна. Кроме того, было отмечено, что локально близкие слова в целевом предложении с большой долей вероятности локально близки и в исходной. Поэтому на каждом шаге перевода при вычислении очередных весов внимания имеет смысл учитывать веса внимания предыдущего шага. В работе [11] была рассмотрена возможность посимвольного перевода, в [12] исследователи из Facebook применили нейронные сети с механизмом внимания для автореферирования предложений, в [13, 14] внимание используется для решения задачи распознавания речи, в [15] – для построения модели, способной отвечать на вопросы. Наконец, в работе [16] были описаны детали реализации Google's Neural Machine Translation – системы с вниманием, которая в 2016 г. легла в основу сервиса Google Translate.

С ростом научного интереса к механизму внимания приходило понимание большого потенциала данного архитектурного решения. Основная масса работ использовала его в связке с рекуррентными нейронными сетями, в частности с LSTM-ячейками [17] или их модификацией GRU [18]. Однако вычисления в рекуррентных нейронных сетях имеют низкий уровень параллелизма, поскольку сама архитектура требует последовательной обработки поступающих единиц [19]. Эта проблема мотивировала поиск путей усовершенствования подхода. Например, в работах [20, 21] вместо рекуррентных нейронных сетей использовались одномерные сверточные сети (convolutional neural network) [22]. В статье [23] исследователи из Google применили механизм внимания для декомпозиции задач при решении проблемы сопоставления двух утверждений в терминах «следствие», «противоречие» или «нейтральная связь». Авторы показали, что подобная архитектура допускает высокий уровень параллелизации вычислений, а качество полученных результатов не ниже достигнутых с помощью других методов.

Можно сказать, что механизм внимания стал результатом естественного развития архитектуры seq2seq. Текст – сложная структура, в которой удаленность двух токенов друг от друга не является исчерпывающим фактором, позволяющим однозначно определять степень их взаимосвязи. Так, например, для разрешения анафоры, т. е. для определения значения встреченного в тексте местоимения, необходимо установить, на какой именно объект оно указывает. Однако определяющее выражение может располагаться на значительном удалении от местоимения, в том числе в другом предложении. Поэтому для качественного решения задач обработки естественного языка необходим более глубокий анализ входных последовательностей. В частности, таковым мог быть переход от линейных входных последовательностей к более сложным конструкциям, таким как синтаксические деревья, выражающие зависимость одних членов предложения от других. Механизм внимания, однако, более информативен, поскольку предоставляет численное выражение связи между двумя токенами, тогда как в древовидной структуре зависимость может быть выражена только наличием либо отсутствием ребра между компонентами. Кроме того, механизм внимания более абстрактен и может быть применен в иных сферах машинного обучения, например при обработке изображений. В том числе его можно адаптировать к другим формам входных данных, не представляющих собой упорядоченную последовательность или матрицу.

С другой стороны, рассчитанные с помощью механизма внимания оценки связи между токенами зависят от качества обучающей выборки и эффективности самого процесса обучения. Таким образом, внимание является дополнительным набором параметров в архитектуре нейронной сети, подбираемых в ходе обработки обширного массива текстов, а не некоторой попыткой

смоделировать внимание человека. Здесь можно провести параллель с системами автоматического перевода: нейронные системы показали лучшую эффективность, чем архитектуры, основанные на правилах и словарях, однако с точки зрения своего устройства и процесса обучения они имеют мало общего с процессом имитации переводческой деятельности человека.

Архитектура Transformer. В 2017 г. сотрудники Google представили публикацию [24], в которой также отказались от использования рекуррентных нейронных сетей. Авторы предложили архитектуру Transformer, основанную на механизме внимания. Кратко принцип работы предложенной архитектуры изложен ниже.

Кодирующий компонент Transformer состоит из стека идентичных по структуре кодировщиков, аналогично при декодировании используется стек декодеров одинакового строения. Глубина обоих стеков совпадает, в публикации она равна шести единицам.

При обработке предложения первый кодировщик получает векторы, соответствующие входящим в него словам. Каждый вектор строится как сумма векторного представления слова и вектора, кодирующего положение рассматриваемого слова в предложении. Для каждого входящего вектора, представляющего слово w_i, путем умножения на обученные матрицы строится тройка векторов: вектор запроса q_i, вектор ключа k_i и вектор значения v_i. Скалярное произведение вектора запроса q_i и вектора ключа слова k_j рассматривается как коэффициент внимания (self-attention), которое следует уделить слову w_j при анализе слова w_i. Результирующий вектор для слова w_i вычисляется как взвешенная сумма векторов значений, где в качестве весов используются нормализованные коэффициенты внимания.

Подобное вычисление проводится параллельно несколько раз (в публикации [24] – восемь) с использованием различных наборов матриц для генерации q_i, k_i и v_i. Данный подход (multi-head attention) позволяет смоделировать различные аспекты внимания. Например, одна цепочка вычислений акцентирует внимание на семантической составляющей слов, что, в частности, позволит справиться с проблемой разрешения анафор [25], другая – на грамматической составляющей, что поможет сохранить связь в словосочетаниях. Векторы, полученные в каждой цепочке вычислений, конкатенируются в один, который после нормализации (layer normalization) [26] передается во второй слой кодировщика – нейронную сеть прямого распространения.

Вывод второго слоя кодировщика – новые векторные представления слов предложения, которые вновь нормализуются и передаются на вход следующему кодировщику в стеке. Аналогичный процесс повторяется для всех оставшихся кодировщиков.

Процесс декодирования близок к процессу кодирования и проходит в три этапа. На первом этапе применяется механизм внимания, аналогичный работе кодировщика, но с естественным ограничением: в качестве входных слов целевого предложения используются уже построенные слова. Второй этап – повторное применение механизма внимания, однако для построения вектора ключа и вектора запроса используются результаты работы стека кодировщиков. Наконец, третий этап – использование нейронной сети прямого распространения. Таким образом последовательно отрабатывают все декодировщики. Вывод последнего декодировщика проходит через линейный слой, который строит новый вектор. Размерность его равна размеру словаря, известного модели. После нормализации значения каждой компоненты такого вектора интерпретируются как условные вероятности того, что соответствующее компоненте слово должно оказаться на следующей позиции строящегося целевого предложения.

Эксперименты показали, что архитектура Transformer может обучаться значительно быстрее, чем решения с использованием рекуррентных и сверточных нейронных сетей, достигая при этом более высоких результатов.

Архитектура Transformer получила широкое распространение в области естественного языка. В частности, она применялась для генерации речи [27], автоматического реферирования текста [28], поддержания диалога [29], решения текстовых математических задач [30]. Оказалось, что подобная архитектура применима в том числе за пределами задач обработки естественного языка: на базе Transformer были построены система рекомендаций [31] и генератор музыки [32].

Были предложены также усовершенствования архитектуры [33–36]. В 2020 г. компания Microsoft объявила о создании модели на базе Transformer, содержащей 17 млрд параметров и предназначенной для синтеза речи (URL: https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/).

Языковые модели. До 2017 г. в сфере обработки естественного языка в качестве переиспользуемых компонентов обычно выступали предобученные векторные представления слов, в частности word2vec [37] и GloVe [38]. В их основе лежало явление, известное в литературе под названием «дистрибутивная гипотеза» [39]. Согласно этому явлению схожие по своему значению слова встречаются в схожих контекстах. Соответственно, механизм работы word2vec и GloVe – обучение на большом корпусе текстов системы, способной ставить в соответствие слову вектор, некоторым образом характеризующий множество контекстов, в которых это слово встречается. Таким образом, согласно гипотезе схожим по смыслу словам будут сопоставлены близкие векторы. Использование word2vec или GloVe в качестве первого слоя нейронной сети повысило качество решения различных задач, отмечают авторы в работе [40]. Тем не менее значительное количество параметров, регулирующих остальную часть системы, требовалось обучать с нуля. Кроме того, строгое сопоставление слова и вектора имело свои недостатки. В частности, как подчеркнуто в [41], эти модели строят единый вектор для многозначных слов (например, слово «киви» может обозначать фрукт, птицу, а в некоторых контекстах использоваться для обозначения жителей Новой Зеландии), что негативно сказывается на качестве работы.

В то же время в другой области применения машинного обучения – компьютерном зрении – доказал свою эффективность прием переноса знаний (transfer learning) [42, с. 43–44] между моделями. Результаты показали, что если обучить сверточную нейронную сеть классифицировать изображения на большом объеме данных, например датасете ImageNet [43], то переиспользование этой сети с небольшим количеством дополнительных слоев позволяет быстро достичь высоких результатов при решении других задач, в частности при определении позы человека на фотографии [44] и обнаружении объектов на картинке [45]. Подход оказался настолько успешен, что в скором времени использование предобученных нейронных сетей де-факто стало стандартом в исследованиях, связанных с распознаванием изображений [46].

Попытки применить схожий подход к обработке естественного языка также оказались успешными. В работе [47] было применено обучение с частичным привлечением учителя (semi-supervised learning): на первом шаге система, имеющая архитектуру seq-2-seq, обучалась без учителя, на втором – дообучалась под задачу классификации. В качестве первого шага авторы предложили два приема: предсказание следующего слова в предложении и самодекодирование, при котором декодировщику требовалось восстановить из контекстного вектора исходное предложение. В работе [40] модель seq-2-seq обучалась автоматическому переводу, после чего обученные кодировщики дообучались под решение других задач. Авторы рассматривали векторы, построенные кодировщиками, как контекстные (context vectors) – альтернативу фиксированным векторным представлениям слов word2vec и GloVe, учитывающую контекст встреченных слов. В статье [48] подобные контекстные векторы обучались с помощью двунаправленной рекуррентной нейронной сети. Идея получила свое развитие в основанных на рекуррентных нейронных сетях моделях ULMFiT [49] и ELMo (Embeddings from Language Models) [50], которые показали лучшие для своего времени результаты на целом классе различных задач обработки естественного языка. В частности, разработчики ULMFiT сравнили свою предобученную на корпусе WikiText-103 [51] модель с нейронной сетью, обучающейся на размеченных данных с нуля. Эксперименты показали, что для достижения равной точности при анализе тональности текста модели ULMFiT требуется в 10 раз меньше размеченных данных, чем непредобученной системе; для определения темы текста – в 20 раз меньше.

В 2018 г. команда OpenAI представила языковую модель Generative Pre-Training (GPT) [52], основанную на архитектуре Transformer с единым стеком преобразователей, которая была предложена в [53]. Для тренировки GPT также применялось обучение с частичным привлечением учителя. Первый шаг в обучении модели – предобучение без учителя: система тренировалась предсказывать следующее слово в предложении на большом корпусе неразмеченных текстов. На втором шаге происходила тонкая настройка (fine-tuning): к модели добавлялась еще одна

нейронная сеть, веса которой дообучались под конкретную задачу. С помощью предложенного подхода авторам удалось превзойти лучшие для своего времени результаты сразу по нескольким задачам: ответы на вопросы по входным данным (reading comprehension), проверка грамматичности (linguistic acceptability), оценка семантической схожести (semantic similarity) и проверка двух текстов на логическое соответствие (textual entailment).

Связь между значением слова и контекстом, в котором оно употреблено, очевидна. Эта связь легла в основу методов построения векторного представления слов word2vec и GloVe. Можно сказать, что языковые модели развивают данную идею, поскольку они строят контекстно-зависимые векторные представления слов. В ходе обучения языковые модели обрабатывают огромное количество различных текстов и вычленяют разнообразные зависимости между токенами. Получив общее представление о том, как обычно токены располагаются в тексте, модель может эффективно дообучаться под конкретные задачи. Можно выделить два взаимообусловленных тезиса: с одной стороны, возможность переноса знаний позволяет использовать для обучения те задачи, для которых требуется неразмеченный корпус данных; с другой стороны, возможность использовать наразмеченные данные позволяет обучаться на практически неограниченном количестве текстов, повышая качество модели и эффективность переноса знаний. Работу языковых моделей можно сравнить с решением логических задач, в которых в качестве актантов и предикатов используются вымышленные лексемы, что, однако, не мешает сделать вывод об истинности или ложности некоторого высказывания.

Модель BERT. Компания Google предложила свою модель языка, получившую название BERT [54]. BERT также базируется на архитектуре Transformer и во многом схожа с GPT. Принципиальным ее отличием является метод обучения: вместо предсказывания последующего слова в последовательности модель BERT на стадии предобучения тренировалась определять закрытые маской слова в предложении. Впервые такая задача была предложена в работе [55]. Второй задачей на этапе предобучения было определение того, следовали ли два предложения в тексте одно за другим. В качестве источников неразмеченных данных в работе выступили корпус BookCorpus [56] и англоязычная Википедия.

Для проведения сравнительного анализа авторы разработали модель BERT-base, которая сопоставима по размерам и производительности с GPT. Вторая модель, BERT-large, имеет в три раза больше параметров – около 340 млн.

BERT показала впечатляющий результат, оказавшись лучшей в мире моделью для решения 11 различных задач. BERT-base по всем задачам оказалась эффективнее, чем GPT, что подтвердило эффективность используемого для предобучения подхода. На многозадачном тесте GLUE (General Language Understanding Evaluation) [57], цель которого – оценить понимание прочитанного компьютером, BERT-base и BERT-large набрали соответственно 79,6 и 82,1 балла из 100, тогда как предыдущим наивысшим достижением было 75,1 балла у GPT.

Модель BERT была выложена в открытый доступ, и в скором времени на ее основе появилось множество новых моделей:

– разработанная в Facebook модель RoBERTa (Robustly optimized BERT approach) [58], для которой был усовершенствован процесс обучения, в том числе увеличен объем неразмеченных данных;

– ALBERT (a lite BERT) [59], созданная совместно сотрудниками Google Research и Toyota Technological Institute, содержащая меньшее число параметров, чем оригинальная BERT, но при этом обучающаяся эффективнее;

– DistilBERT [60] – подвергнутая «дистилляции» (distillation) [61] BERT, которая имеет на 40 % меньше параметров и на 60 % быстрее работает, сохраняя при этом 97 % от качества работы исходный модели;

– TinyBERT [62] от Huawei – еще одна «дистиллированная» версия BERT;

– MT-DNN [63] от Microsoft, которая представляет усовершенствование многозадачной модели, предложенной в работе [64], с использованием BERT в качестве единого компонента. Эта модель на тесте GLUE превзошла средний человеческий результат (URL: https://docs.microsoft.com/archive/blogs/stevengu/microsoft-achieves-human-performance-estimate-on-glue-benchmark);

– StructBERT [65] от разработчиков Alibaba, при обучении которой использовались задачи предсказания порядка слов и предложений [66]. Таким образом авторы стремились сообщить модели больше информации о базовых языковых структурах;

– BioBERT [67], предназначенная для работы с текстами на биомедицинскую тематику;

– VilBERT (vision-and-language BERT) [68] – расширенная модель BERT, которая работает с парами «изображение – текст».

В мае 2020 г. среди лидеров в тесте GLUE большинство составляют модели, основанные на BERT.

Итак, можно выделить две ключевые идеи, лежащие в основе модели BERT: механизм внимания, на котором построена архитектура Transformer, и характерный для языковых моделей принцип переноса знаний. Как утверждалось выше, языковые модели обучаются искать закономерности между токенами в корпусе текстов. Механизм внимания ориентирован на поиск взаимосвязей между токенами в конкретных входных данных. Совершенно естественно, что два подхода соединились в эффективный ансамбль, ставший своего рода прорывом в сфере обработки естественного языка. Практика показывает, что увеличение числа параметров в архитектуре модели и увеличение обучающей выборки ведет к построению еще более качественных моделей. Развитие науки в этом направлении можно описать как итеративный процесс, включающий два этапа: создание более крупной и эффективной языковой модели, демонстрирующей лучшие результаты в сфере обработки естественного языка, и разработку более простых и быстрых моделей, способных демонстрировать схожий уровень.

Слабые стороны BERT. Следует отметить, что, несмотря на очень высокий процент правильных решений серии различных задач, BERT и другие языковые модели требуют дальнейшего усовершенствования. Как показывают исследования, после обучения эти системы ориентируются в том числе на ложные эвристики, обусловленные неудачным подбором данных в используемых датасетах [69]. Хотя BERT-система решила задачу понимания аргументации (argument reasoning comprehension) с точностью 77 %, что всего на 3 % меньше среднего человеческого уровня, авторы работы [69] утверждают, что в области понимания аргументации BERT не обучается ничему. При тестировании систем на новых, специально подготовленных тестовых данных эффективность языковых моделей оказывается существенно ниже [70, 71].

Представление о том, как слова взаимосвязаны друг с другом, позволяет на высоком уровне взаимодействовать с текстом и решать большинство задач естественной обработки языка. Тем не менее по своей сути это скорее копирование и вставка информации, а не понимание.

Работа BERT и других языковых моделей напоминает работу персонажа мысленного эксперимента о китайской комнате. В этом мысленном эксперименте, предложенном философом Джоном Сёрлом, описывается человек, не знающий китайских иероглифов. Он находится в запертой комнате, в которой также имеется подробная инструкция по манипуляции иероглифами. Вне комнаты находится наблюдатель, который через щель передает в комнату некоторое сообщение на китайском языке. Находящийся в комнате человек получает это сообщение и в соответствии с инструкцией перерисовывает некоторый ответ, который возвращает наблюдателю. В указанных условиях у наблюдателя может сложиться представление, что человек в комнате владеет китайским языком. На самом же деле он не имеет ни малейшего представления о теме разговора и просто выполняет инструкции.

Примечательно, что представление о мире, полученное языковыми моделями в ходе обучения, в некотором роде перекликается с постмодернистской концепцией Жака Деррида о мире как совокупности текстов. BERT, как и другие языковые модели, теоретически может извлекать из корпуса текстов знания: например, попытка предсказать последнее слово в фразе «самой быстрой птицей на свете является» будет обусловлена закономерностью в использовании слов, но может привести к верному ответу. Тем не менее понимание в том смысле, в котором этот термин применим к человеку, для автоматической языковой модели едва ли достижимо. Одной из проблем является интерпретация дейктических единиц: завершение предложения «в настоящий момент президентом США является» оказывается гораздо более сложной задачей, поскольку в корпусе текстов разных лет схожая фраза может завершаться различным образом.

«Представление» машины о мире можно несколько расширить с помощью взаимодействия системы с реальными или абстрактными явлениями и объектами. В этом случае знания модели

будут включать не только взаимосвязь токенов между собой, но и взаимосвязь отдельных токенов с объектами мира. В некотором смысле это будет сравнимо с изучением языка ребенком, который обучается ему параллельно с исследованием предметов окружающей действительности.

Вторая проблема заключается в том, что для предварительного обучения языковой модели BERT требуются огромные объемы текстов: в оригинальной работе были использованы корпус BookCorpus и англоязычная Википедия, в последующих публикациях (см., к примеру, [58]) множество текстов было расширено. Данный подход не применим в случае, когда отобрать настолько обширную коллекцию текстов для текущей задачи невозможно. Такая проблема обнаруживается, например, при обработке малых языков, не имеющих широкого использования в Интернете и литературной традиции. Для решения задачи компьютерной обработки подобных малых языков следует разработать иные методы. В частности, научный интерес представляет эффективность, с которой ребенок изучает родной язык. Успешное моделирование онтогенеза языка могло бы обеспечить построение качественных моделей малых языков.

Заключение. Модель BERT опирается на использование механизма внимания и принцип переноса знаний. Работая совместно, эти две идеи позволяют эффективно обнаруживать закономерности между применением слов и токенов в огромной обучающей выборке неразмеченных текстов. Обнаруженных закономерностей оказывается достаточно, чтобы эффективно решать разнообразные задачи, связанные с автоматической обработкой естественного языка. Более того, качество работы можно увеличивать, усложняя архитектуру модели и расширяя обучающую выборку.

Главный недостаток BERT и подобных моделей заключается в том, что понимание имитируется за счет закономерностей, найденных в текстах, в том числе за счет ложных эвристик. Второй недостаток вытекает из первого: для качественной имитации понимания требуется длительное обучение на огромной выборке.

Отметим, что оба недостатка обусловлены архитектурой модели, поэтому BERT не может быть доработана с целью их исправления. Чтобы избежать упомянутых негативных моментов, следует рассматривать принципиально иные подходы к моделированию автоматических обработчиков естественного языка. Таким подходом, в частности, может быть моделирование онтогенеза языка – процесса обучения ребенка родному языку.

References

1. Cho K., Merriënboer B. van, Gulcehre C., Bahdanau D., Bougares F., Schwenk H., Bengio Y. Learning phrase representations using RNN encoder-decoderfor statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014*, pp. 1724–1734. https://doi.org/10.3115/v1/D14-1179

2. Sutskever I., Vinyals O., Le Q. V. Sequence to sequence learning with neural networks. *Neural Information Processing Systems*, 2014, pp. 3104–3112. Available at: https://arxiv.org/abs/1409.3215 (accessed 07.07.2020).

3. Serban I. V., Lowe R., Charlin L., Pineau J. Generative deep neural networks for dialogue: A short review. *Neural Information Processing Systems, Workshop on Learning Methods for Dialogue*, 2016. Available at: https://arxiv.org/abs/1611.06216 (accessed 07.07.2020).

4. Vinyals O., Toshev A., Bengio S., Erhan D. Show and tell: A neural image caption generator. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, Massachusetts, USA, 7–12 June 2015*, pp. 3156–3164. https://doi.org/10.1109/CVPR.2015.7298935

5. Loyola P., Marrese-Taylor E., Matsuo Y. A Neural architecture for generating natural language descriptions from source code changes. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, Canada, 30 July – 4 August 2017*, vol. 2, pp. 287–292. https://doi.org/ 10.18653/v1/P17-2045

6. Lebret R., Grangier D., Auli M. Neural text generation from structured data with application to the biography domain. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, USA, 1–5 November 2016*, pp. 1203–1213. https://doi.org/10.18653/v1/D16-1128

7. Nikolenko S., Kandurin A., Arhangelskaja E. Glubokoe obuchenie. *Deep Learning*. Saint Petersburg, Piter, 2020, 480 p. (in Russian).

8. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*, 2015. Available at: https://arxiv.org/abs/1409.0473 (accessed 07.07.2020).

9. Schuster M., Paliwal K. K. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions*, 1997, vol. 45(11), pp. 2673–2681. https://doi.org/10.1109/78.650093

10. Luong T., Pham H., Manning C. D. Effective approaches to attention-based neural machine translation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015*, pp. 1412–1421. https://doi.org/10.18653/v1/D15-1166

11. Chung J., Cho K., Bengio Y. A character-level decoder without explicit segmentation for neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016*, vol. 1, pp. 1693–1703. https://doi.org/10.18653/v1/P16-1160

12. Rush A., Chorpa S., Weston J. A neural attention model for abstractive sentence summarization. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015*, pp. 379–389. https://doi.org/10.18653/v1/D15-1044

13. Chorowski J., Bahdanau D., Serdyuk D., Cho K., Bengio Y. Attention-based models for speech recognition. *Proceedings of the 28th International Conference on Neural Information Processing Systems*, 2015, vol. 1, pp. 577–585. Available at: https://arxiv.org/abs/1506.07503 (accessed 07.07.2020).

14. Chan W., Jaitly N., Le Q. V., Vinyals O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. *2016 IEEE International Conference on Acoustics, Speech and Signal Processing, Shanghai, China, 20–25 March 2016*, pp. 4960–4964. https://doi.org/10.1109/ICASSP.2016.7472621

15. Hermann K. M., Kočiský T., Grefenstette E., Espeholt L., Kay W., Suleyman M., Blunsom P. Teaching machines to read and comprehend. *Neural Information Processing Systems 28: 29th Annual Conference on Neural Information Processing Systems*, 2015, pp. 1693–1701. Available at: https://arxiv.org/abs/1506.03340 (accessed 07.07.2020).

16. Wu Y., Schuster M., Chen Z., Le Q. V., Norouzi M., …, Dean J. *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*, 2016. Available at: https://arxiv.org/abs/1609.08144 (accessed 07.07.2020).

17. Hochreiter S., Schmidhuber, J. Long short-term memory. *Neural Computation*, 1997, vol. 9(8), pp. 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

18. Cho K., Merrienboer B. van, Bahdanau D., Bengio Y. On the properties of neural machinetranslation: Encoder-decoder approaches. *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, 25 October 2014*, pp. 103–111. https://doi.org/10.3115/v1/ W14-4012

19. Martin E., Cundy C. Parallelizing linear recurrent neural nets over sequence length. *International Conference on Learning Representations*, 2018. Available at: https://arxiv.org/abs/1709.04057 (accessed 07.07.2020).

20. Kalchbrenner N., Espeholt L., Simonyan K., Oord van den A., Graves A., Kavukcuoglu K. *Neural Machine Translation in Linear Time*, 2016. Available at: https://arxiv.org/abs/1610.10099 (accessed 07.07.2020).

21. Gehring J., Auli M., Grangier D., Yarats D., Dauphin Y. N. Convolutional sequence to sequence learning. *Proceedings of the 34th International Conference on Machine Learning*, 2017, vol. 70, pp. 1243–1252. Available at: https://arxiv.org/abs/1705.03122 (accessed 07.07.2020).

22. LeCun Y., Bottou L., Bengio Y., Haffner P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998, vol. 86(11), pp. 2278–2324. https://doi.org/10.1109/5.726791

23. Parikh A. P., Täckström O., Das D., Uszkoreit J. A decomposable attention model for natural language inference. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, USA, 1–5 November 2016*, pp. 2249–2255. https://doi.org/10.18653/v1/D16-1244

24. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., …, Polosukhin I. Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, California, USA, 4–9 December 2017*, pp. 6000–6010. Available at: https://arxiv.org/abs/ 1706.03762 (accessed 07.07.2020).

25. Mitkov R. *Anaphora Resolution: The State of the Art*. Paper based on the COLING'98/ACL'98 tutorial on anaphora resolution. Wolverhampton, University of Wolverhampton, 1999, 34 p.

26. Ba J. L., Kiros J. R., Hinton G. E. *Layer Normalization*, 2016. Available at: https://arxiv.org/abs/1607.06450 (accessed 07.07.2020).

27. Li N., Liu S., Liu Y., Zhao S., Liu M., Zhou M. Neural speech synthesis with transformer network. *The AAAI Conference on Artificial Intelligence*, 2019. Available at: https://arxiv.org/abs/1809.08895 (accessed 07.07.2020).

28. Khandelwal U., Clark K., Jurafsky D., Kaiser Ł. *Sample Efficient Text Summarization using a Single Pre-Trained Transformer*, 2019. Available at: https://arxiv.org/abs/1905.08836 (accessed 07.07.2020).

29. Vlasov V., Mosig J. E. M., Nicho A. *Dialogue Transformers*, 2019. Available at: https://arxiv.org/abs/1910.00486 (accessed 07.07.2020).

30. Griffith K., Kalita J. Solving arithmetic word problems automatically using transformer and unambiguous representations. *International Conference on Computational Science and Computational Intelligence, Las Vegas, USA, 5–7 December 2019*, pp. 526–532. https://doi.org/10.1109/CSCI49370.2019.00101

31. Kang W.-C., McAuley J. Self-attentive sequential recommendation. *IEEE International Conference on Data Mining, Singapore, 17–20 November 2018*, pp. 197–206. https://doi.org/10.1109/ ICDM.2018.00035

32. Huang C.-Z. A., Vaswani A., Uszkoreit J., Shazeer N., Simon I., …, Eck D. *Music Transformer*, 2018. Available at: https://arxiv.org/abs/1809.04281 (accessed 07.07.2020).

33. Dehghani M., Gouws S., Vinyals O., Uszkoreit J., Kaiser Ł. Universal transformers. *7th International Conference on Learning Representations*, 2019. Available at: https://arxiv.org/abs/1807.03819 (accessed 07.07.2020).

34. Dai Z., Yang Z., Yang Y., Carbonell J., Le Q. V., Salakhutdinov R. Transformer-XL: attentive language models beyond a fixed-length context. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July – 2 August 2019*, pp. 2978–2988. https://doi.org/10.18653/v1/P19-1285

35. So D. R., Liang C., Le Q. V. The evolved transformer. *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 5877–5886. Available at: https://arxiv.org/abs/1901.11117 (accessed 07.07.2020).

36. Zhao C., Xiong C., Rosset C., Song X., Bennett P., Tiwary S. Transformer-XH: multi-evidence reasoning with eXtra hop attention. *8th International Conference on Learning Representations*, 2020. Available at: https://openreview.net/forum?id=r1eIiCNYwS (accessed 07.07.2020).

37. Mikolov T., Chen K., Corrado G., Dean J. Distributed representations of words and phrases and their compositionality. *Proceedings of the 26th International Conference on Neural Information Processing Systems*, 2013, vol. 2, pp. 3111–3119. Available at: https://arxiv.org/abs/1310.4546 (accessed 07.07.2020).

38. Pennington J., Socher R., Manning C. D. Glove: global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 25–29 October 2014*, pp. 1532–1543. https://doi.org/10.3115/v1/D14-1162

39. Sahlgren M. The distributional hypothesis. From context to meaning. Distributional models of the lexicon in linguistics and cognitive science (special issue of the Italian Journal of Linguistics). *Rivista di Linguistica*, 2008, vol. 20(1), pp. 33–53.

40. McCann, B., Bradbury J., Xiong C., Socher R. Learned in translation: contextualized word vectors. *31st Conference on Neural Information Processing Systems*, 2017, pp. 6297–6308. Available at: https://arxiv.org/abs/1708.00107 (accessed 07.07.2020).

41. Hedderich M. A., Yates A., Klakow D., Melo G. de. Using Multi-Sense Vector embeddings for reverse dictionaries. *Proceedings of the 13th International Conference on Computational Semantics – Long Papers, Gothenburg, Sweden, 23–27 May 2019*, pp. 247–258. https://doi.org/10.18653/v1/W19-0421

42. Ruder S. *Neural Transfer Learning for Natural Language Processing. Ph. D. Thesis*. Galway, National University of Ireland, 2019, 329 p.

43. Deng J., Dong W., Socher R., Li L.-J., Li K., Fei-Fei L. ImageNet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009*, pp. 248–255. https://doi.org/10.1109/CVPR.2009.5206848

44. Papandreou G., Zhu T., Kanazawa N., Toshev A., Tompson J., Bregler C., Murphy K. Towards accurate multi-person pose estimation in the wild. *IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017*, pp. 3711–3719. https://doi.org/10.1109/CVPR.2017.395

45. He K., Gkioxari G., Dollár P., Girshick R. Mask R-CNN. *IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017*, pp. 2980–2988. https://doi.org/10.1109/ICCV.2017.322

46. Mahajan D., Girshick R., Ramanathan V., He K., Paluri M., …, Maaten L. van der. Exploring the limits of weakly supervised pretraining. *European Conference on Computer Vision, Munich, Germany, 8–14 September 2018*, pp. 181–196. https://doi.org/10.1007/978-3-030-01216-8_12

47. Dai A. M., Le Q. V. Semi-supervised sequence learning. *Neural Information Processing Systems 28: 29th Annual Conference on Neural Information Processing Systems 2015, Montreal, Canada, 7–12 December 2015*, vol. 2, pp. 3079–3087. https://doi.org/10.18653/v1/P17-1161

48. Peters M. E, Ammar W., Bhagavatula C., Power R. Semi-supervised sequence tagging with bidirectional language models. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 2017, vol. 1, pp. 1756–1765. Available at: https://arxiv.org/abs/1705.00108 (accessed 07.07.2020).

49. Howard J., Ruder S. Universal language model fine-tuning for text classification. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018*, vol. 1, pp. 328–339. https://doi.org/10.18653/v1/P18-1031

50. Peters M. E., Neumann M., Iyyer M., Gardner M., Clark C., Lee K., Zettlemoyer L. Deep contextualized word representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, Louisiana, USA, 1–6 June 2018*, vol. 1, pp. 2227–2237. https://doi.org/10.18653/v1/N18-1202

51. Merity S., Xiong C., Bradbury J., Socher R. Pointer sentinel mixture models. *5th International Conference on Learning Representations*, 2017. Available at: https://arxiv.org/abs/1609.07843 (accessed 07.07.2020).

52. Radford A., Narasimhan K., Salimans T., Sutskever I. Improving language understanding with unsupervised learning. *Technical report*, 2018. Available at: https://openai.com/blog/language-unsupervised/ (accessed 07.07.2020).

53. Liu P. J., Saleh M., Pot E., Goodrich B., Sepassi R., Kaiser L., Shazeer N. Generating wikipedia by summarizing long sequences. *6th International Conference on Learning Representations*, 2018. Available at: https://arxiv.org/abs/1801.10198 (accessed 07.07.2020).

54. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019*, vol. 1, pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423

55. Taylor W. L. Cloze procedure: A new tool for measuring readability. *Journalism Bulletin*, 1953, vol. 30(4), pp. 415–433.

56. Zhu Y., Kiros R., Zemel R., Salakhutdinov R., Urtasun R., Torralba A., Fidler S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015*, pp. 19–27. https://doi.org/10.1109/ICCV.2015.11

57. Wang A., Singh A., Michael J., Hill F., Levy O., Bowman S. R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *Proceedings of the 2018 EMNLP Workshop Blackbox NLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, 1 November 2018*, pp. 353–355. https://doi.org/10.18653/v1/W18-5446

58. Liu Y., Ott M., Goyal N., Du J., Joshi M., …, Stoyanov V. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*, 2019. Available at: https://arxiv.org/abs/ 1907.11692 (accessed 07.07.2020).

59. Lan Z., Chen M., Goodman S., Gimpel K., Sharma P., Soricut R. ALBERT: a lite BERT for self-supervised learning of language representations. *8th International Conference on Learning Representations*, 2020. Available at: https://openreview.net/forum?id=H1eA7AEtvS (accessed 07.07.2020).

60. Sanh V., Debut L., Chaumond J., Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Conference on Neural Information Processing Systems*, 2019. Available at: https://arxiv.org/abs/1910.01108 (accessed 07.07.2020).

61. Hinton G., Vinyals O., Dean J. Distilling the knowledge in a neural network. *Neural Information Processing Systems. Deep Learning and Representation Learning Workshop*, 2015. Available at: https://arxiv.org/abs/1503.02531 (accessed 07.07.2020).

62. Jiao X., Yin Y., Shang L., Jiang X., Chen X., …, Liu Q. *TinyBERT: Distilling BERT for Natural Language Understanding*, 2019. Available at: https://arxiv.org/abs/1909.10351 (accessed 07.07.2020).

63. Liu X., He P., Chen W., Gao J. Multi-task deep neural networks for natural language understanding. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July – 2 August 2019*, pp. 4487–4496. https://doi.org/10.18653/v1/P19-1441

64. Liu X., Gao J., He X., Deng L., Duh K., Wang Y.-Y. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, 31 May– 5 June 2015*, pp. 912–921. https://doi.org/10.3115/v1/N15-1092

65. Wang W., Bi B., Yan M., Wu C., Xia J., …, Si L. StructBERT: incorporating language structures into pre-training for deep language understanding. *8th International Conference on Learning Representations*, 2020. Available at: https://openreview.net/forum?id=BJgQ4lSFPH (accessed 07.07.2020).

66. Elman J. L. Finding structure in time. *Cognitive Science*, 1990, vol. 14(2), pp. 179–211.

67. Lee J., Yoon W., Kim S., Kim D., Kim S., …, Kang J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 2020, vol. 36(4), pp. 1234–1240. https://doi.org/10.1093/bioinformatics/btz682

68. Lu J., Batra D., Parikh D., Lee S. *ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks*, 2019. Available at: https://arxiv.org/abs/1908.02265 (accessed 07.07.2020).

69. Niven T., Kao H.-Y. Probing neural network comprehension of natural language arguments. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July – 2 August 2019*, pp. 4658–4664. https://doi.org/10.18653/v1/P19-1459

70. Zellers R., Holtzman A., Bisk Y., Farhadi A., Choi Y. HellaSwag: Can a machine really finish your sentence? *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July – 2 August 2019*, pp. 4791–4800. https://doi.org/10.18653/v1/P19-1472

71. McCoy T., Pavlick E., Linzen T. Right for the wrong reasons: diagnosing syntactic heuristics in natural language inference. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July – 2 August 2019*, pp. 3428–3448. https://doi.org/10.18653/v1/P19-1334

Информация об авторе

Качков Дмитрий Ильич, аспирант кафедры многопроцессорных систем и сетей факультета прикладной математики и информатики, Белорусский государственный университет, Минск, Беларусь.
E-mail: dmitriydikanskiy@gmail.com

Information about the author

Dzmitry I. Kachkou, Postgraduate Student of Department of Multiprocessor Systems and Networks of the Faculty of Applied Mathematics and Informatics, Belarusian State University, Minsk, Belarus.
E-mail: dmitriydikanskiy@gmail.com