Интеллектуальные системы управления, анализ данных

© 2020 г. А.В. ГЛАЗКОВА, канд. техн. наук (a.v.glazkova@utmn.ru)
(Тюменский государственный университет)

ТЕМАТИЧЕСКАЯ КЛАССИФИКАЦИЯ ТЕКСТОВЫХ ФРАГМЕНТОВ С УЧЕТОМ ИХ БЛИЖАЙШЕГО КОНТЕКСТА[1]

Описывается подход к проведению тематической классификации отрывков биографического текста, учитывающий ближайший контекст классифицируемых фрагментов, с помощью нейронной сети с несколькими входами. Выбор архитектуры модели обоснован предположением о том, что, поскольку тексты, написанные на естественном языке, отличаются логичностью и связностью, контекст отрывка может быть использован в качестве дополнительных входных данных. Модель обучена и протестирована на корпусе биографических текстов, составленном автором работы. Результаты, полученные с использованием предложенного подхода, превзошли результаты моделей, не учитывающих контекст отрывка.

Ключевые слова: классификация предложений, интеллектуальный анализ данных, рекуррентные нейронные сети, обработка естественного языка, биографический текст, контекст, корпус текстов, биографическое исследование, Word2Vec, BERT.

DOI: 10.31857/S0005231020120090

1. Введение

Интерпретация неструктурированной информации, представленной в виде текста на естественном языке, является одной из ключевых задач интеллектуального анализа данных и информационного поиска. Частной задачей информационного поиска является поиск биографической информации, актуальной при проведении биографических исследований, сборе историко-генеалогических данных и биографических фактов из жизни индивидуума. Спецификой данной задачи является, во-первых, жанровое многообразие источников биографической информации (автобиографии, заметки, очерки и т.д.), и, во-вторых, многоплановость биографической информации, включающей в себя разнообразные аспекты жизни человека: политический, личный, общественный, культурный.

[1] Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект № 18-37-00272).

Развитие методов поиска биографической информации осуществляется в основном в двух направлениях [1]:

1) предметный или «косвенный» поиск, когда пользователь поисковой системы формулирует запросы, основываясь на известных ему биографических фактах о некоторой персоне и пытаясь на их основе найти недостающую информацию;

2) свободный поиск, при котором пользователь не имеет начальных сведений об интересующей его персоне. Свободный поиск подразумевает просмотр биографических текстов, посвященных персоне, в целях обнаружения конкретной биографической информации, релевантной требованиям пользователя (например, информации о профессиональной деятельности или о личной жизни).

Во втором случае пользователь вынужден просматривать большие объемы текстов. Сократить затраты временных ресурсов при свободном поиске биографической информации мог бы помочь инструмент автоматической обработки биографических текстов, извлекающий из них фрагменты, связанные с тем или иным типом биографической информации. Такой инструмент может быть реализован на основе методов автоматической классификации текстов. В этом случае текст, предварительно разделенный на фрагменты, подается на вход классификатора, определяющего тематику каждого фрагмента.

Различные тексты в зависимости от жанра могут иметь стандартизированную структуру (к таким текстам относятся, например, документы официально-делового стиля) или обладать структурированностью, заложенной не в расположении структурных частей, а в логическом единстве [2]. Биографические тексты могут послужить примером второго типа текстов за счет того, что информация в них, как правило, изложена в хронологическом порядке, и знание тематики отрывка такого текста позволяет предположить, какой фрагмент ему предшествует и какой располагается после. Эта особенность позволяет предположить, что принятие во внимание логики изложения биографических текстов и учет ближайшего контекста фрагментов даст возможность улучшить качество тематической классификации отрывков.

В данной работе предлагается подход к тематической классификации фрагментов биографического текста на основе их ближайшего контекста. В качестве фрагмента рассматривается предложение, так как данная языковая единица представляет собой грамматически организованное соединение слов (или слово), обладающее смысловой законченностью [3]. В статье приводится сравнение нескольких моделей машинного обучения для классификации фрагментов биографических текстов с учетом ближайшего контекста и без него. Эксперименты проводятся на корпусе биографических текстов, собранном автором работы.

2. Работы по близкой тематике

Тематика работы в основном затрагивает две задачи обработки естественного языка:

1) извлечение биографической информации (биографических фактов);

2) тематическая классификация предложений.

Существующие работы, посвященные решению указанных задач, преследуют различные практические цели и используют разные подходы. Однако в целом в литературе по данной тематике извлечение информации и классификацию текстов определяют как слабоформализуемые задачи, а применяемые для их решения методы — как зависящие от специфики обрабатываемых текстов [4, 5]. Методы поиска и извлечения биографической информации развиваются преимущественно в трех направлениях: детерминированные подходы, основанные на применении шаблонов и правил; подходы, основанные на применении методов машинного обучения (в частности, нейронных сетей); гибридные подходы. Детерминированные подходы показывают достаточно высокую результативность во многих задачах, однако требуют разработки большого количества признаков, отражающих структурные, семантические и лексические особенности текстов. К преимуществам подходов, основанных на машинном обучении, можно отнести автоматическую настройку параметров моделей с помощью множества примеров, а также возможность не только соотносить результаты обработки текстов с их отдельными характеристиками, но и выявлять более сложные скрытые зависимости и закономерности [6]. Однако реализация подходов, использующих методы машинного обучения, требует построения обучающих выборок текстов, сопровождающихся качественной разметкой, что также бывает сложно осуществимо в реальных условиях. Одним из трендов обработки естественного языка являются предобученные модели на основе глубоких нейронных сетей (transfer learning) [7, 8], когда заранее обученная модель дообучается для решения специфических задач [9].

К детерминированным подходам к извлечению биографической информации можно отнести работу [1], в которой описана технология, представляющая биографический факт в виде древовидной структуры, корнем которой является тип факта (например, “рождение”), а листьями – связанные с фактом сущности. В [10] предлагается подход к извлечению биографических событий на основе трафика Википедии. В [11] описывается набор правил для извлечения биографической информации для текстов на русском языке. В [12] проводится сравнение нескольких подходов, основанных на правилах, а также предлагается таксономия биографических фактов, включающая в себя семь типов отношений. Существует достаточно много работ, авторы которых применяли различные методы машинного обучения для классификации фрагментов биографических текстов или извлечения биографических фактов. Так, в [13] используется наивный байесовский классификатор, в [14] – метод опорных векторов и деревья решений, в [15, 16] — нейронные сети. В [17] проводилось сравнение подходов, основанных на правилах, с методом опорных векторов на примере бинарной классификации фраз, содержащих и не содержащих биографическую информацию, в результате которого метод опорных векторов продемонстрировал значительно более высокое качество. В [18] сравнивались различные типы машинного обучения для извлечения отношений в биографических текстах (по сути извлечения фактов) на при-

мере португальского языка. Среди гибридных подходов могут быть названы [19, 20].

Во многих работах, связанных с поиском биографической информации, эксперименты проводились на текстах Википедии (в частности, [21–26]). Это связано с тем, что Википедия содержит в себе богатый и разнообразный материал для исследований, представленный тем не менее в стандартизованном виде.

Особенностями задачи классификации предложений являются, во-первых, сравнительно небольшая длина классифицируемых текстов и, во-вторых, наличие контекста у предложений, который также может приниматься во внимание алгоритмами классификации. Будет ли во время классификации учитываться контекст, зависит от специфики решаемой задачи и данных, имеющихся для проведения исследования. Многие существующие системы для классификации коротких текстов используют алгоритмы, построенные на использовании вероятностных и статистических методов: байесовского классификатора [27], условных случайных полей [28], скрытых марковских моделей [29], логистической регрессии [30]. Для решения задач классификации текстов широко применяются рекуррентные нейронные сети, обученные с помощью векторных представлений символов и слов. В частности, подходы к обработке естественного языка, основанные на применении рекуррентных нейронных сетей, представлены в [31–35]. В последние годы высокие результаты в классификации коротких текстов демонстрируют инструменты, использующие модели ELMo и BERT (в частности, [36–38]).

Среди исследований, связанных с использованием контекста, можно назвать работу [39], где была предложена архитектура нейронной сети для классификации реплик в диалоге. Описанная в указанной работе модель имела несколько входов, один из которых принимал текущую реплику, а другие – ее контекст, т.е. предшествующие фразы. Модель, построенная таким образом, продемонстрировала более высокое качество классификации в сравнении с обычной рекуррентной нейронной сетью на англоязычных диалоговых текстовых корпусах. В [40, 41] тем же коллективом автором были предложены нейросетевая модель для разбиения фрагментов аннотаций медицинских статей по пяти имеющимся классам: введение, обзор существующих работ, методология, результаты и выводы. В [42] описывается подход к классификации предложений по тональности с использованием ряда дискурсивных признаков.

3. Методы

В данной работе предлагается нейросетевая архитектура для классификации фрагментов биографических текстов, основанная на архитектуре для классификации реплик в диалоге, описанной в [39]. Предлагаемая модель включает в себя несколько входов, раздельно обрабатывающих текущий текстовый фрагмент, а также предыдущие и последующие фрагменты. Векторы, являющиеся результатами обработки фрагментов во входных блоках, объединяются в общий слой нейронной сети. Для оценки качества классификации используется корпус биографических текстов, собранный и размеченный ав-

тором работы в полуавтоматическом режиме [43]. В работе рассматриваются два варианта нейросетевой архитектуры, использующие разные типы представления предложений:

1) рекуррентная нейронная сеть. Текст представляется в виде последовательностей слов. В качестве матрицы векторных представлений слов (для слоя Embedding) используются предобученные вектора модели Word2Vec [44];

2) сеть прямого распространения, обученная на векторных представлениях предложений, полученных с помощью модели BERT [7].

3.1. Архитектура

Рекуррентная модель основана на использовании рекуррентных слоев долгой краткосрочной памяти (long short-term memory, LSTM), в которых в отличие от классических рекуррентных архитектур предусмотрен механизм хранения долгосрочных зависимостей, позволяющий избежать проблемы затухания градиента [45]. Структура ячейки LSTM-сети представлена на рис. 1 [46].

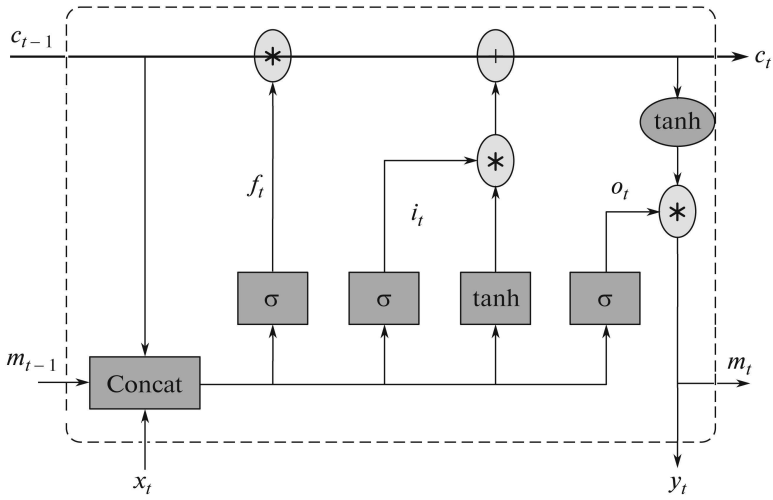

Рис. 1. Структура ячейки LSTM.

Пусть x_t и y_t – входной и выходной сигналы соответственно в момент времени t, а c_t и m_t – состояние ячейки и выхода в момент t. Преобразование входного сигнала в выходной при этом происходит следующим образом:

$$
\begin{gathered}
i_t = \sigma\big(W_{ix}x_t + W_{im}m_{t-1} + W_{ic}c_{i-1} + b_i\big),\\
f_t = \sigma\big(W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{i-1} + b_f\big),\\
o_t = \sigma\big(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_{i-1} + b_o\big),\\
m_t = o_t \odot h(c_t),\\
y_t = \varphi\big(W_{ym}m_t + b_y\big),\\
c_t = f_t \odot c_{t-1} + i_t \odot g\big(W_{cx}x_t + W_{cm}m_{t-1} + b_c\big),
\end{gathered}
\tag{1}
$$

где W_{cx}, W_{ix}, W_{fx}, W_{ox} – веса входов, W_{cm}, W_{im}, W_{fm}, W_{om} – веса состояний ячеек, b_o, b_i, b_f – смещения, W_{ic}, W_{fc}, W_{oc} – веса связей между ячейками и слоем выходного фильтра, W_{ym} и b_y – вес и смещение для выхода, σ, g, h представляют собой некоторые нелинейные функции.

В сети прямого распространения рекуррентные слои заменены слоями прямого распространения, т.е. слоями без рекуррентных связей, с функцией активации "гиперболический тангенс".

Входными данными моделей являются текущее предложение и его контекст, т.е. n предшествующих и n последующих предложений. Пусть s_j – предложение с порядковым номером j. Тогда входом служит множество предложений S:

$$S = \{s_{j-n}, \ldots, s_{j-1}, s_j, s_{j+1}, \ldots, s_{j+n}\}, \quad j \in [n+1, J-n], \tag{2}$$

J – количество предложений в тексте.

В том случае, когда $j < n+1$ или $j > J-n$, предложение контекста s_k, $k \in \{j-n, \ldots, j-1, j+1, \ldots, j+n\}$ подается на вход сети, если $1 \leqslant k \leqslant J$. В противном случае в качестве входных данных для соответствующей позиции подаются метки начала (для $k < 1$) или конца текста ($k > J$).

Каждому предложению из множества S соответствует отдельный вход сети. Таким образом, входными данными сети являются $2n+1$ предложений, а выходными данными входных блоков являются векторы, соответствующие входным предложениям.

3.2. Варианты учета контекста

Далее рассматривались три варианта архитектуры каждой модели:

1) в первом случае результат конкатенации выходных векторов входных блоков подается на слой прямого распространения. Результирующие величины поступают в выходной слой модели, также представляющий собой слой прямого распространения, имеющий размерность, равную количеству классов, и функцию активации softmax. Выходной слой сети возвращает распределение вероятностей между тематическими классами для предложения (рис. 2,*а*);

2) во втором варианте к каждому входному блоку добавляется по одному слою прямого распространения, после чего осуществляется конкатенация, результат которой подается на выходной слой. Таким образом, учет влияния контекста происходит на уровне последнего слоя модели (рис. 2,*б*);

3) третий вариант представляет собой комбинацию первых двух. Выполняется конкатенация выходных векторов входных блоков и обработка результата слоем прямого распространения. Одновременно с этим выходные векторы входных блоков, соответствующих предложениям контекста, подаются на вход слоев прямого распространения. Осуществляется конкатенация всех результирующих векторов, результат подается на выходной слой. Так, влияние контекста учитывается как на уровне выходных векторов рекуррентных блоков, так и на уровне последнего слоя модели (рис. 2,*в*).

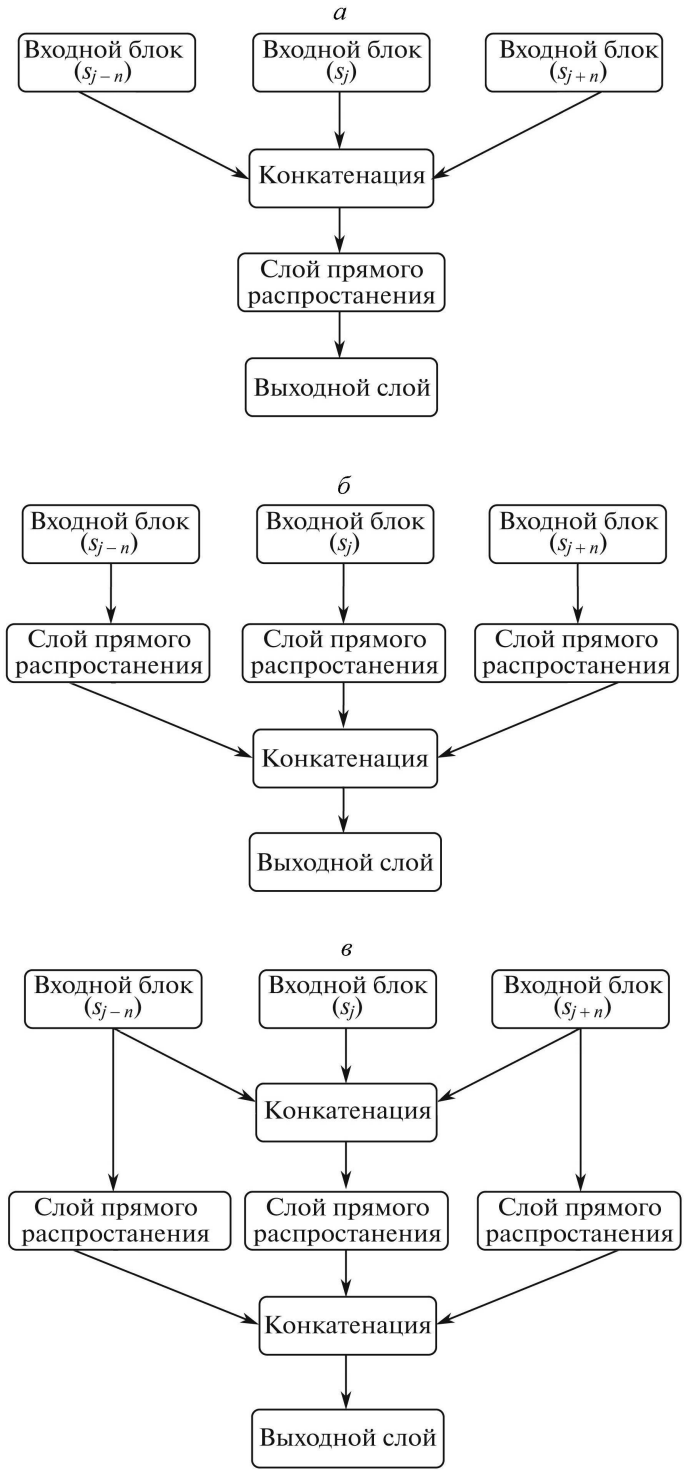

Рис. 2. Варианты архитектуры модели.

4. Эксперименты

4.1. Данные

Для обучения и тестирования моделей был составлен корпус биографических текстов. Он представляет собой коллекцию, содержащую биографические тексты из онлайн-энциклопедии Википедия, разбитые на предложения и снабженные тематической разметкой. В версии корпуса, использованной для экспериментов, содержатся 200 текстов, описывающих биографии людей, живших или живущих в XX–XXI вв. Корпус находится в свободном доступе на сайте [47].

Каждому предложению в корпусе биографических текстов сопоставлена метка класса, наиболее полно соответствующего его тематике: рождение, информация о родительской семье, место жительства, род занятий, место работы, семья, образование, личные события, профессиональные события, смерть. Некоторым предложениям в корпусе соответствуют два класса – основной и дополнительный. В данной работе при классификации таких предложений использовалась метка основного класса. Таким образом, каждый фрагмент соответствует одному из 10 классов. Характеристики корпуса представлены в табл. 1.

Для выравнивания количества примеров в классах был проведен простой оверсэмплинг, т.е. дублирование случайных элементов миноритарных классов. Общее количество элементов для обучения и валидации моделей после проведения оверсэмплинга – 8251, объем тестовой выборки, на которой оценивалось финальное качество моделей, – 177 предложений. Предварительная обработка данных включала в себя приведение текста к нижнему регистру, удаление специальных символов, стоп-слов и знаков препинания, а также приведение слов к начальной форме.

Таблица 1. Характеристики корпуса

Класс	Средняя длина предложения (в токенах)	Средний размер контекста для $n = 1$ (в токенах)	Количество примеров
Рождение	13,9	13,25	134
Информация о родительской семье	13,05	28,9	86
Место жительства	13,17	31,23	94
Род занятий	16,93	32,4	943
Место работы	14,4	32,27	113
Семья	11,83	24,25	48
Образование	15,73	30,04	374
Личные события	20,35	38,07	105
Профессиональные события	21,36	37,72	490
Смерть	10,56	22,15	111
Все предложения	**16,83**	**31,52**	**2498**

В случае отсутствия необходимого числа соседних предложений в тексте на вход моделей подавались специальные метки “begin” и “end” – для предшествующих $(s_{j-1}, s_{j-2}, \ldots, s_{j-n})$ и последующих $(s_{j+1}, s_{j+2}, \ldots, s_{j+n})$ предложений соответственно. Так, в случае $j = 1$ (порядковый номер предложения в тексте), $J = 3$ (количество предложений в тексте) и $n = 3$ (размер контекста) входные данные будут иметь следующий вид:

$$
\begin{aligned}
s_{j-3} &= \text{“}\textit{begin}\text{”},\\
s_{j-2} &= \text{“}\textit{begin}\text{”},\\
s_{j-1} &= \text{“}\textit{begin}\text{”},\\
s_{j} &= \text{“Текст предложения } s_j\text{”},\\
s_{j+1} &= \text{“Текст предложения } s_{j+1}\text{”},\\
s_{j+2} &= \text{“Текст предложения } s_{j+2}\text{”},\\
s_{j+3} &= \text{“}\textit{end}\text{”}.
\end{aligned}
$$

4.2. Реализация и обучение моделей

В ходе экспериментов было проведено сравнение моделей, учитывающих контекст, с нейронными сетями, основанными на transformer-архитектуре, а также с методом опорных векторов, испытанным на представлениях предложений в виде Bag-of-Words TF-IDF. В данной работе использовались две модели BERT:

1) mBERT (multilingual BERT), поддерживающая 104 языка [7];
2) RuBERT, модель BERT для русского языка, обученная на русскоязычной Википедии и текстах новостных порталов [48]. Для русскоязычных текстов данная модель на ряде задач показала качество, значительно превосходящее качество многоязычной модели BERT.

Таблица 2. Оптимизация параметров моделей

Параметр	Множество параметров	Выбранное значение
Функция активации на внутренних слоях (рекуррентная сеть)	гиперболический тангенс, relu	гиперболический тангенс
Функция активации на внутренних слоях (сеть прямого распространения)	гиперболический тангенс, relu	гиперболический тангенс
Размерность слоя LSTM (рекуррентная сеть)	степени 2 из диапазона [32;256]	64
Размерность слоя прямого распространения во входном блоке (сеть прямого распространения)	степени 2 из диапазона [64;512]	256
Размерность общего слоя прямого распространения (рекуррентная сеть)	степени 2 из диапазона [32;128]	32
Размерность общего слоя прямого распространения (сеть прямого распространения)	степени 2 из диапазона [32;128]	32

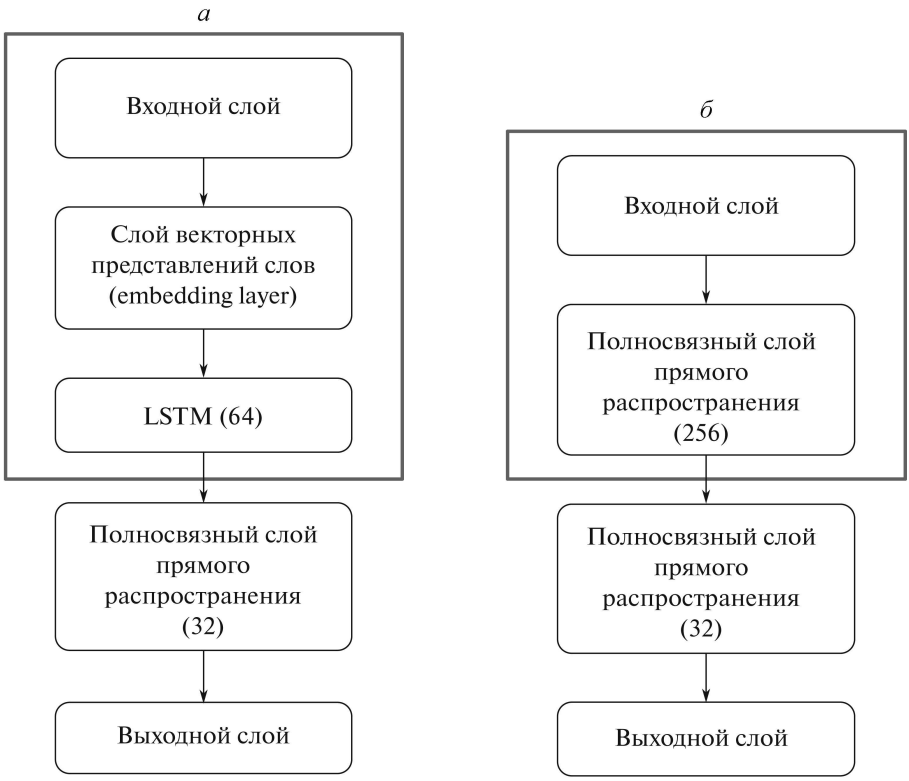

Рис. 3. Модели без учета контекста: *а* — рекуррентная сеть; *б* — сеть прямого распространения.

Реализация моделей, основанных на transformer-архитектуре, выполнена с помощью библиотек Transformers [49] и PyTorch [50] и языка программирования Python 3.6. В качестве элемента входных данных для модели BERT выступает предложение, заключенное в токены [CLS] и [SEP]. Предложение обрабатывается токенизатором, преобразующим токены в последовательности индексов в соответствии со словарем модели. Размерность элемента входной последовательности для BERT ограничена 512 токенами, размер батча – 8, количество эпох обучения – 3.

Метод опорных векторов реализован с помощью библиотеки Scikit-Learn (LinearSVC) [51]. В качестве входных данных использованы представления предложений по модели Bag-of-Words TF-IDF (матрица Bag-of-Words, где на пересечении строки и столбца располагается значение меры TF-IDF для данного слова в заданном документе). Размерность векторных представлений – 5000 признаков.

Реализация моделей, использующих контекст, выполнена с помощью средств библиотеки Keras [52] и языка программирования Python 3.6. Входные блоки рекуррентных сетей состоят из входного слоя, слоя матрицы весовых коэффициентов (embedding layer) и рекуррентного слоя LSTM. Размерность слоя LSTM составляет 64 нейрона, слоев прямого распространения –

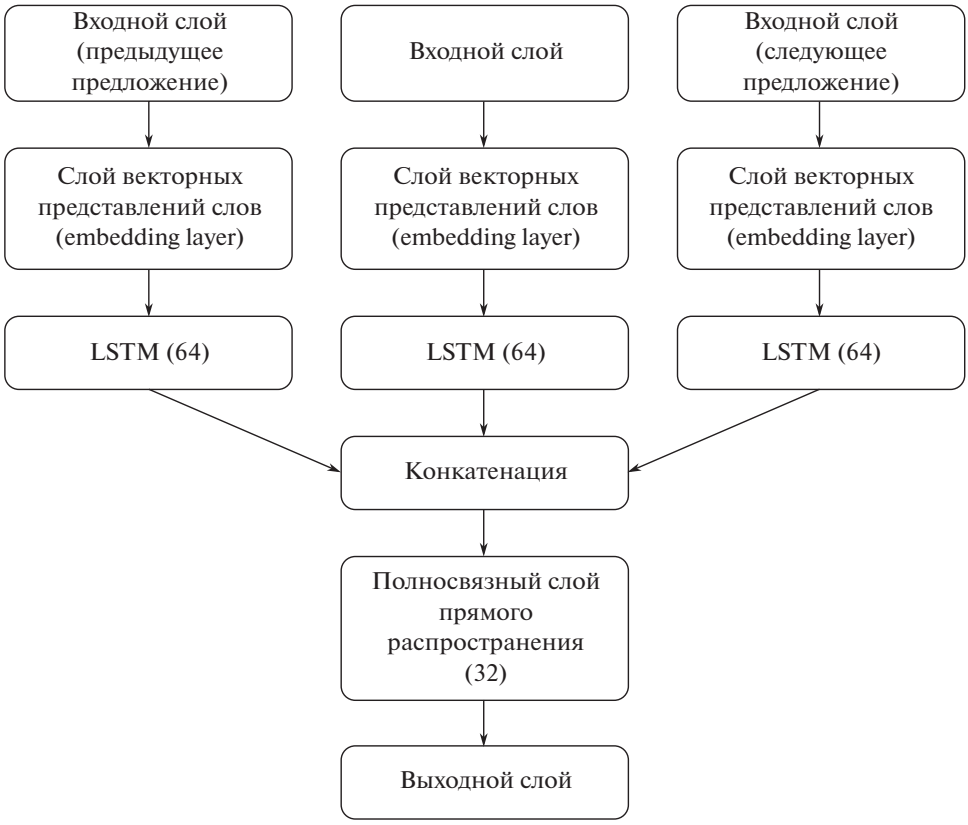

Рис. 4,*а*. Визуализация рекуррентных моделей при $n = 1$.

32 нейрона. Входные блоки сетей прямого распространения включают в себя входной слой и один слой прямого распространения. Размерность слоев прямого распространения во входных блоках составляет 256 нейронов, в общем слое – 32 нейрона. Функция активации для внутренних слоев – гиперболический тангенс, для выходного слоя – softmax. Оптимизация гиперпараметров моделей проводилась на примере моделей без учета контекста с помощью простого поиска по решетке (grid search). Список оптимизируемых параметров и их диапазонов приводится в табл. 2. В качестве оптимизационного алгоритма для всех моделей использован adaptive moment estimation (adam optimizer), в качестве функции ошибки – категориальная кросс-энтропия.

На рис. 3 изображены схемы рекуррентной модели и сети прямого распространения без учета контекста. Наклонным шрифтом выделены входные блоки. В моделях с учетом контекста (когда $n > 0$) каждому входному предложению из множества S соответствует отдельный входной блок. На рис. 4 в качестве примера представлена визуализация трех вариантов рекуррентной модели для $n = 1$. Модели для $n > 1$ имеют аналогичный вид при большем количестве входов.

Данные для обучения нейросетевых моделей были разделены на обучающую и валидационную выборки. Обучение проводилось с использованием

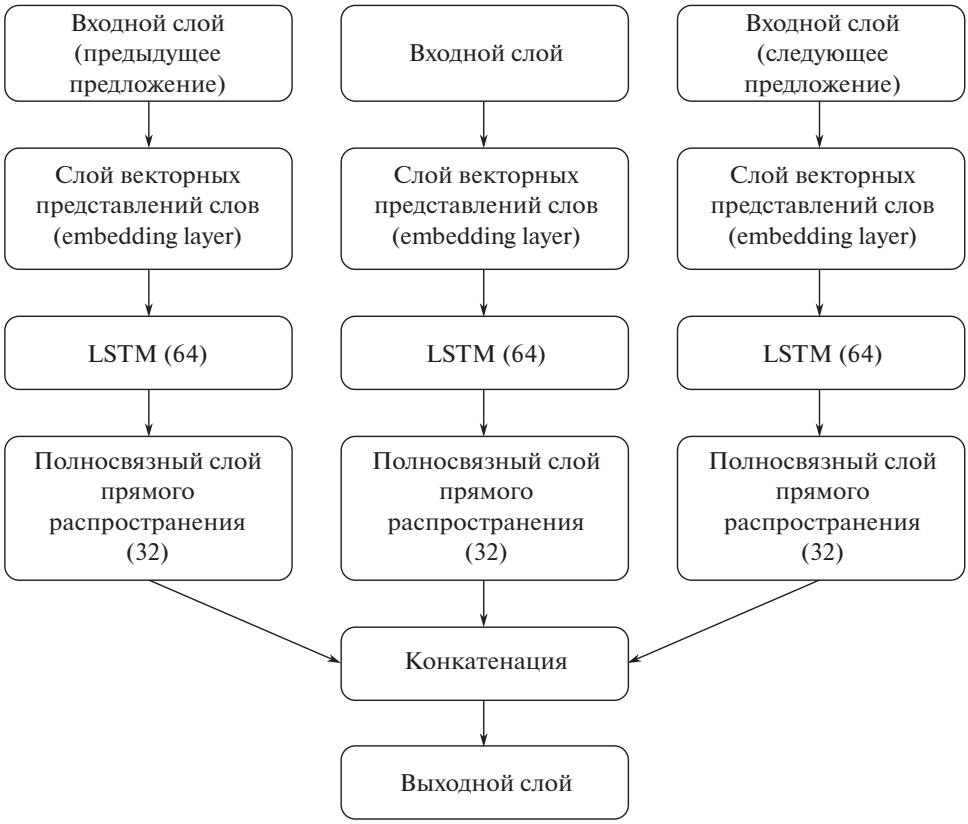

Рис. 4,*б*. Визуализация рекуррентных моделей при $n = 1$.

обучающей выборки, остановка обучения выполнялась согласно показателям модели на валидационной выборке. Финальное тестирование модели осуществлялось на независимой тестовой выборке, не участвовавшей в процессе обучения. Для рекуррентных сетей входные данные подавались в модели в виде последовательностей слов (sequences) на основе матрицы векторных представлений слов, составленной из векторов модели Word2Vec и множества лексем, представленных в обучающей выборке. В качестве модели Word2Vec использовалась модель, обученная на текстах русскоязычной Википедии и Национального корпуса русского языка за 2018 г. с использованием алгоритма обучения Skip-gram [53]. Размерность векторного представления слова в модели равна 300. Входные данные сетей прямого распространения выглядят как одномерный вектор размерностью 768, полученный для текущего фрагмента текста из модели RuBERT с помощью библиотеки DeepPavlov [54].

Исходный код всех моделей доступен по ссылке [55].

4.3. Результаты

Для оценки результатов использовалась F-мера (macro-averaging), которая определялась как средняя величина значений F-меры, рассчитанных для

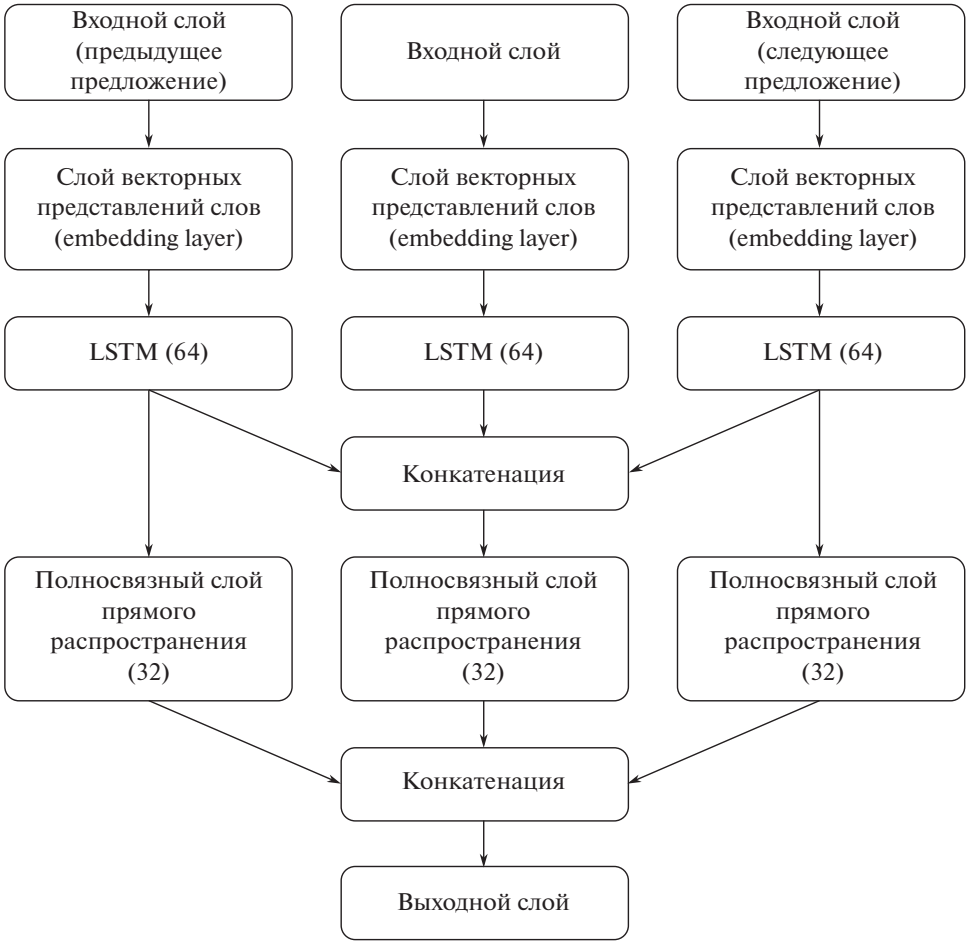

Рис. 4,*в*. Визуализация рекуррентных моделей при $n = 1$.

каждого класса по показателям точности (precision) и полноты (recall). Значения точности и полноты приведены в скобках после значения F-меры (первый показатель – precision, второй – recall).

В табл. 3 представлены показатели качества классификации. Поскольку ввиду случайной инициализации начальных параметров результаты классификации могут варьироваться при разных запусках моделей, каждая нейросетевая модель была запущена m раз, в таблице указаны средние значения. В данной работе $m = 5$. В экспериментах рассматривались значения для моделей, учитывающих контекст фрагмента в диапазоне $0 \leqslant n \leqslant 3$, так как дальнейшее увеличение величины n не давало роста качества классификации и отрицательно сказывалось на временной сложности модели.

Полужирным шрифтом в таблице выделены наиболее высокие значения F-меры среди всех рассмотренных моделей (рекуррентная – вариант 1, 94,77%) и среди моделей, не учитывающих контекст (RuBERT, 93,16%). Как показывают данные таблицы, в большинстве случаев добавление пред-

Таблица 3. Качество моделей (F-мера (Precision / Recall), значения указаны в %)

Архитектура	Значение n			
	0	1	2	3
Рекуррентная (вариант 1)	–	91,25 (90,31 / 92,15)	93,13 (91,23 / 94,13)	**94,77 (91,33 / 95,77)**
Рекуррентная (вариант 2)	–	92,54 (91,45 / 92,56)	93 (91,98 / 94,03)	92,9 (91,56 / 92,35)
Рекуррентная (вариант 3)	–	92,07 (92,01 / 91,87)	92,3 (92,12 / 91,96)	94,07 (91,4 / 95,88)
Прямого распространения (вариант 1)	–	86,23 (85,89 / 88,42)	87,14 (84,78 / 89,2)	87,45 (85,91 / 88,45)
Прямого распространения (вариант 2)	–	87,18 (86,56 / 89,12)	87,03 (86,22 / 88,14)	87,5 (87,02 / 88,49)
Прямого распространения (вариант 3)	–	87,35 (87,53 / 89,36)	87,56 (87,34 / 89,01)	87,14 (87,02 / 88,32)
Рекуррентная (без учета контекста)	89,46 (90,13 / 88,3)	–	–	–
Прямого распространения (без учета контекста)	86,83 (89,3 / 88,26)	–	–	–
LinearSVC	66,37 (64,39 / 77,44)			
mBERT	89,01 (92,11 / 88,12)	–	–	–
RuBERT	**93,16 (90,72 / 97,05)**	–	–	–

ложений контекста позволило улучшить качество классификации фрагментов. Причем для рекуррентных моделей наилучший результат был достигнут при $n = 3$, а для сетей прямого распространения – при $n = 2$. Наибольшие абсолютные показатели улучшения заметны для рекуррентных моделей (+5,31%).

В табл. 4 приводятся примеры предложений, ошибочно классифицированных рекуррентной моделью с использованием контекста и моделью RuBERT. В большинстве случаев ошибки связаны с фрагментами, тематически связанными более чем с одним классом. Многие из этих фрагментов имели в оригинальном корпусе метку дополнительного класса. Так, предложению из первого примера (фрагмент биографии художника Б.В. Эндера) разметчики корпуса сопоставили класс “Информация о родительской семье” в качестве основного и класс “Рождение” в качестве дополнительного. Выбор метки основного класса связан с тем, что предложение описывает происхождение персоны, а не конкретизирует факт рождения (дата, место). Обе сети отнесли данный отрывок к классу “Рождение”. Второе предложение является примером, характеризующим сильные стороны модели с использованием контекста. Вероятно, фрагмент, описывающий профессио-

Таблица 4. Примеры ошибок моделей

Фрагмент	Разметка	Результат классификации (RuBERT)	Результат классификации (рекуррентная модель с использованием контекста)
1	2	3	4
s_{j-3}: “begin” s_{j-2}: “begin” s_{j-1}: “begin” s_j: “Родился в семье агронома, происходящего из рода обрусевших немцев.” s_{j+1}: “Две его младшие сестры – Ксения (1894–1955) и Мария (1897–1942) – также стали художницами.” s_{j+2}: “В 1905–1907 брал частные уроки рисования у И.Я. Билибина.” s_{j+3}: “В 1911 г. сблизился с М.В. Матюшиным и Е.Г. Гуро, часто бывал в их квартире в доме на Песочной улице.”	Информация о родительской семье	Рождение	Рождение
s_{j-3}: “begin” s_{j-2}: “begin” s_{j-1}: “Училась в школе №1, индустриальном техникуме.” s_j: “1935 – аэроклуб, после гражданская авиация в Грузии вместе с мужем.” s_{j+1}: “1941 – инструктор (200 курсантов).” s_{j+2}: “Апрель 1944 – Саранск, 3-е Военно-Морское летное училище (летчики-штурмовики).” s_{j+3}: “По окончании – назначение в 7 ГвПШАП ВВС КБФ (командир полка дважды Герой Советского Союза А.Е. Мазуренко).”	Род занятий	Семья	Род занятий

Таблица 4 (окончание)

1	2	3	4
s_{j-3}: “В 1918 г., после демобилизации, поступил в Петроградские Государственные свободные художественные мастерские, занимался у К.С. Петрова-Водкина, затем у Матюшина.”	Личные события	Личные события	Профессиональные события
s_{j-2}: “Завершив обучения в 1923 г., продолжил работать под началом Матюшина в Отделе органической культуры Инхука, вошел в созданную им группу «Зорвед».”			
s_{j-1}: “В 1920-е принимал активное участие в выставках «мастерской пространственного реализма».”			
s_j: “Познакомился с К.С. Малевичем, Н.М. Суетиным, И.Н. Харджиевым, И.Г. Эренбургом, поддерживал с ними постоянную переписку.”			
s_{j+1}: “В 1927 г. переехал в Москву.”			
s_{j+2}: “В 1930-х гг. много работал в области монументального искусства.”			
s_{j+3}: “Принимал участие в оформлении павильона СССР на Международной выставке в Париже (1937).”			

нальную деятельность советской летчицы Л.И. Шулайкиной и упоминающий ее супруга, верно отнесен к классу “Род занятий” за счет тематики контекста, в то время как модель RuBERT классифицировала этот фрагмент как элемент класса “Семья”. Противоположный пример представляет собой третье предложение (фрагмент биографии Б.В. Эндера), которое отнесено разметчиками корпуса к классу “Личные события”, так как описывает встречи и личные знакомства художника. Модель с учетом контекста классифицировала данный фрагмент как “Профессиональные события” (в корпусе данному классу соответствуют упоминания официальных встреч и наград). Возможно, полученный результат обусловлен “профессиональной” тематикой контекста.

5. Заключение

В работе представлен подход к выполнению тематической классификации отрывков текста, учитывающий их ближайший контекст. Модель апробирована на примере корпуса биографических текстов. Поскольку биографический текст отличается хронологической последовательностью изложения, все модели, принимающие в качестве входных данных контекст отрывка, показали лучшие результаты в сравнении с моделью без учета контекста.

Архитектура, предложенная в данной статье, может быть применена при решении сходных задач тематической классификации отрывков текстов, обладающих явной логической структурой и последовательностью изложения.

СПИСОК ЛИТЕРАТУРЫ

1. *Адамович И.М., Волков О.И.* Система извлечения биографических фактов из текстов исторической направленности // Системы и средства информатики. 2015. № 3. С. 235–250. https://doi.org/10.14357/08696527150315.
2. *Голуб И.Б.* Стилистика русского языка: Учеб. пособие. М.: Рольф; Айрис-пресс, 1997.
3. *Валгина Н.С., Розенталь Д.Э., Фомина М.И.* Современный русский язык. Учебник. 6-е изд., перераб. и доп. М.: Логос, 2002.
4. *Manning C., Raghavan P., Schütze H.* Introduction to Information Retrieval. Cambridge University Press, 2008.
5. *Большакова Е.И., Воронцов К.В., Ефремова Н.Э. и др.* Автоматическая обработка текстов на естественном языке и анализ данных: учеб. пособие. М.: Изд-во НИУ ВШЭ, 2017.
6. *Захарова И.Г.* Big Data и управление образовательным процессом // Вестн. Тюмен. гос. ун-та. Гуманитарные исследования. Humanitates. 2017. Т. 3. № 1. С. 210–219. https://doi.org/10.21684/2411-197X-2017-3-1-210-219.
7. *Devlin J., Chang M.W., Lee K., et al.* Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. 2018.
8. *Peters M.E., Neumann M., Iyyer M., et al.* Deep contextualized word representations // Proc. NAACL-HLT. V. 1. 2018. P. 2227–2237.
9. *Барахнин В.Б., Кожемякина О.Ю., Мухамедиев Р.И. и др.* Проектирование структуры программной системы обработки корпусов текстовых документов // Бизнес-информатика. 2019. Т. 13. № 4. С. 60–72.

   https://doi.org/10.17323/1998-0663.2019.4.60.72.
10. *Hogue A., Nothman J., Curran J.R.* Unsupervised biographical event extraction using wikipedia traffic // Proc. Australasian Language Technology Association Workshop. 2014. P. 41–49.
11. *Bonch-Osmolovskaya A., Kolbasov M.* Tolstoy Digital: Mining Biographical Data in Literary Heritage Editions // CEUR Workshop Proc. 1. BD 2015 – Proc. 1st Conf. on Biographical Data in a Digital World 2015. 2015. P. 48–52.
12. *Garera N., Yarowsky D.* Structural, transitive and latent models for biographic fact extraction // Proc. 12th Conf. of the Eur. Chapter of the ACL (EACL 2009). 2009. P. 300–308. https://doi.org/10.3115/1609067.1609100.
13. *Conway M.* Mining a corpus of biographical texts using keywords // Liter. Lingist. Comput. 2010. V. 25. No. 1. P. 23–35. https://doi.org/10.1093/llc/fqp035.

14. *Zhou L., Ticrea M., Hovy E.* Multi-document biography summarization // Proc. 2004 Conf. on Empirical Methods in Natural Language Processing. 2004. P. 434–441.
15. *Vempala A., Blanco E.* Extracting Biographical Spatial Timelines: Corpus and Experiments // IEEE/ACM Transactions on Audio, Speech, and Language Processing. 2020. https://doi.org/10.1109/taslp.2020.2988418/
16. *Chisholm A., Radford W., Hachey B.* Learning to generate one-sentence biographies from wikidata // Proc. 15th Conf. of the Eur. Chapter of the Association for Computational Linguistics: V. 1, Long Papers. 2017. P. 633–642.
https://doi.org/10.18653/v1/e17-1060.
17. *Yu D., Ji H., Li S., et al.* Why read if you can scan? Trigger scoping strategy for biographical fact extraction // Proc. 2015 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015. P. 1203–1208. https://doi.org/10.3115/v1/n15-1126.
18. *Garcia M., Gamallo P.* Exploring the effectiveness of linguistic knowledge for biographical relation extraction // Natural Language Engineering. 2015. V. 21. No. 4. P. 519–551. https://doi.org/10.1017/s1351324913000314.
19. *Jing H., Kambhatla N., Roukos S.* Extracting social networks and biographical facts from conversational speech transcripts // Proc. 45th Annual Meeting of the Association of Computational Linguistics. 2007. P. 1040–1047.
20. *Biadsy F., Hirschberg J., Filatova E.* An unsupervised approach to biography production using Wikipedia // Proc. ACL-08: HLT. 2008. P. 807–815.
21. *Gotti F., Langlais P.* From French Wikipedia to Erudit: A test case for cross-domain open information extraction // Computational Intelligence. 2018. V. 34. No. 2. P. 420–439. https://doi.org/10.1111/coin.12120.
22. *Menini S., Sprugnoli R., Moretti G. et al.* Ramble on: tracing movements of popular historical figures // Proc. Software Demonstrations of the 15th Conf. of the Eur. Chapter of the Association for Computational Linguistics. 2017. P. 77–80. https://doi.org/10.18653/v1/e17-3020/
23. *Russo I., Caselli T., Monachini M.* Extracting and Visualising Biographical Events from Wikipedia // BD. 2015. P. 111–115.
24. *Plum A., Zampieri M., Orasan C. et al.* Large-scale data harvesting for biographical data // Biographical Data in a Digital World. At: Varna, Bulgaria. 2019.
25. *Flekova L., Ferschke O., Gurevych I.* What makes a good biography? Multidimensional quality analysis based on Wikipedia article feedback data // Proc. 23rd Int. Conf. on World wide web. 2014. P. 855–866.
https://doi.org/10.1145/2566486.2567972.
26. *Petrasova S., Khairova N., Lewoniewski W. et al.* Similar text fragments extraction for identifying common Wikipedia communities // Data. 2018. V. 3. No. 4. P. 66. https://doi.org/10.3390/data3040066.
27. *Huang K.C., Chiang I.J., Xiao F., et al.* PICO element detection in medical text without metadata: Are first sentences enough? // J. Biomed. Inform. 2013. No. 5. P. 940–946. https://doi.org/10.1016/j.jbi.2013.07.009.
28. *Yamamoto Y., Takagi T.* A sentence classification system for multi biomedical literature summarization // 21st Int. Conf. on Data Engineering Workshops (ICDEW'05). 2005. P. 1163–1163. https://doi.org/10.1109/icde.2005.170.
29. *Xu R., Supekar K., Huang Y. et al.* Combining text classification and Hidden Markov Modeling techniques for categorizing sentences in randomized clinical trial abstracts // Annual Symposium proceedings. AMIA Symposium. American Medical Informatics Association. 2006. P. 824–828.

30. *Mikhalkova E.V., Ganzherli N.V., Karyakin Y.E., et al.* Machine learning classification of user interests across languages and social networks // Komp. Lingvistika i Intel. Tehn. 2018. P. 501–511.
31. *Chen T., Xu R., He Y., et al.* Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN // Expert Systems with Applications. 2017. V. 72. P. 221–230. https://doi.org/10.1016/j.eswa.2016.10.065.
32. *Kim Y.* Convolutional Neural Networks for Sentence Classification // Proc. 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP). 2014. P. 1746–1751. https://doi.org/10.3115/v1/d14-1181.
33. *Wang J., Yu L.C., Lai K.R., et al.* Dimensional sentiment analysis using a regional CNN-LSTM model // Proc. 54th Annual Meeting of the Association for Computational Linguistics (V. 2: Short Papers). 2016. P. 225–230.
https://doi.org/10.18653/v1/p16-2037.
34. *Trofimovich J.* Comparison of neural network architectures for sentiment analysis of russian tweets // Computational Linguistics and Intellectual Technologies: Proc. Int. Conf. Dialogue. 2016. P. 50–59.
35. *Gordeev D.* Detecting state of aggression in sentences using CNN // Int. Conf. on Speech and Computer. Springer, Cham. 2016. P. 240–245.
https://doi.org/10.1007/978-3-319-43958-7_28.
36. *Miftahutdinov Z., Alimova I., Tutubalina E.* KFU NLP Team at SMM4H 2019 Tasks: Want to Extract Adverse Drugs Reactions from Tweets? BERT to The Rescue // Proc. Fourth Social Media Mining for Health Applications (# SMM4H) Workshop & Shared Task. 2019. P. 52–57. https://doi.org/10.18653/v1/w19-3207.
37. *Mapes N., White A., Medury R., et al.* Divisive Language and Propaganda Detection using Multi-head Attention Transformers with Deep Learning BERT-based Language Models for Binary Classification // Proc. Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda. 2019. P. 103–106. https://doi.org/10.18653/v1/d19-5014.
38. *Peng Y., Yan S., Lu Z.* Transfer Learning in Biomedical Natural Language Processing: An Evaluation of BERT and ELMo on Ten Benchmarking Datasets // Proc. 18th BioNLP Workshop and Shared Task. 2019. P. 58–65.
https://doi.org/10.18653/v1/w19-5006.
39. *Lee J.Y., Dernoncourt F.* Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks // Proc. NAACL-HLT, 2016. P. 515–520. https://doi.org/10.18653/v1/n16-1062.
40. *Dernoncourt F., Lee J.Y., Szolovits P.* Neural Networks for Joint Sentence Classification in Medical Paper Abstracts // Proc. 15th Conf. of the Eur. Chapter of the Association for Computational Linguistics: V. 2, Short Papers. 2017. P. 694–700. https://doi.org/10.18653/v1/e17-2110.
41. *Jin D., Szolovits P.* Hierarchical Neural Networks for Sequential Sentence Classification in Medical Scientific Abstracts // Proc. 2018 Conf. on Empirical Methods in Natural Language Processing. 2018. P. 3100–3109.
https://doi.org/10.18653/v1/d18-1349.
42. *Yang B., Cardie C.* Context-aware learning for sentence-level sentiment analysis with posterior regularization // Proc. 52nd Annual Meeting of the Association for Computational Linguistics (V. 1: Long Papers). 2014. P. 325–335.
https://doi.org/10.3115/v1/p14-1031.
43. *Глазкова А.В.* Автоматический поиск фрагментов, содержащих биографическую информацию, в тексте на естественном языке // Тр. ин-та сист. прогр. РАН. 2018. № 6. С. 221–236. https://doi.org/10.15514/ISPRAS-2018-30(6)-12.

44. *Mikolov T., Chen K., Corrado G., et al.* Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781. 2013.

45. *Hochreiter S., Schmidhuber J.* Long Short-term Memory // Neural. Comput. 1997. No. 8. P. 1735–1780.

46. *Bai T., Dou H.J., Zhao W.X., et al.* An Experimental Study of Text Representation Methods for Cross-Site Purchase Preference Prediction Using the Social Text Data // J. Comput. Sci. Technol. 2017. No. 4. P. 828–842.
https://doi.org/10.1007/s11390-017-1763-6.

47. Корпус биографических текстов. URL: https://sites.google.com/site/utcorpus. Дата доступа: 06.10.19.

48. *Kuratov Y., Arkhipov M.* Adaptation of deep bidirectional multilingual transformers for Russian language. arXiv preprint arXiv:1905.07213. 2019.

49. Transformers. URL: https://huggingface.co/transformers/. Дата доступа: 27.05.20.

50. PyTorch. URL: https://pytorch.org/. Дата доступа: 27.05.20.

51. Scikit-Learn. Machine Learning in Python.
URL: https://scikit-learn.org/stable/index.html. Дата доступа: 29.05.20.

52. Keras: The Python Deep Learning library. URL: https://keras.io/. Дата доступа: 17.09.19.

53. *Kutuzov A., Kuzmenko E.* WebVectors: A Toolkit for Building Web Interfaces for Vector Semantic Models // Communicat. Comput. Inform. Sci. V. 661. P. 155–161. https://doi.org/10.1007/978-3-319-52920-2_15.

54. DeepPavlov: an open source conversational AI framework.
URL: http://deeppavlov.ai/. Дата доступа: 27.05.20.

55. Тематическая классификация фрагментов биографии с учетом их ближайшего контекста. URL: https://github.com/oldaandozerskaya/ait. Дата доступа: 27.05.20.

Статья представлена к публикации членом редколлегии О.П. Кузнецовым.

Поступила в редакцию 08.10.2019

После доработки 30.05.2020

Принята к публикации 09.07.2020