



И. В. Трофимов, Ю. П. Сердюк, Е. А. Сулейманова,  
Н. А. Власова

## Разрешение событийно-несобытийной неоднозначности существительных: нейросетевой подход

**Аннотация.** В работе предложен метод автоматического определения значения существительных, неоднозначных с точки зрения событийно-несобытийной категоризации. Представлены результаты сопоставительного анализа нескольких нейросетевых архитектур, использованных для снятия неоднозначности. Обсуждается проблема обнаружения в тексте упоминаний о событиях.

**Ключевые слова и фразы:** обнаружение событий в тексте, разрешение лексической неоднозначности, нейронная сеть, обработка естественного языка.

### Введение

Рассматриваемая задача различения значений неоднозначных существительных возникает в контексте более широкой задачи обнаружения упоминаний событий в тексте (главным образом в постановке с нефиксированной предметной областью — т. наз. open-domain event extraction). Распознавание событий [1, 2] востребовано в различных прикладных задачах, связанных с информационным поиском и содержательной обработкой текстовой информации: создание вопросно-ответных систем, автоматическое реферирование [3], сюжетная и тематическая организация информации [4], построение темпорально аннотированных текстовых корпусов [5, 6].

Типичным языковым средством выражения событийного значения безусловно выступают глагольные формы. Однако высока и доля случаев, когда для референции к событиям используются существительные (35 % в корпусе ACE-2005 [7]). Существительным,

---

Работа выполнена при поддержке РФФИ (проект № 19-07-00991-а).

© И. В. Трофимов, Ю. П. Сердюк, Е. А. Сулейманова, Н. А. Власова, 2020

© Институт программных систем имени А. К. Айламазяна РАН, 2020

© Программные системы: теория и приложения (дизайн), 2020



способным выступать в роли имен ситуаций, свойственна регулярная многозначность, возникающая вследствие метонимических переносов вида «имя ситуации — участник ситуации (субъект, объект, инструмент, средство, результат, место)» (см. Таблицу 1 в [8]). Связанная с этим проблема неоднозначности в литературе по теме обнаружения событий упоминается, но отдельно не рассматривается [3, 7, 9–11]. Оценка методов в таких работах выполняется только в отношении качества обнаружения событий. В ряде источников приводятся оценки точности разметки событийных маркеров — имен существительных [3, 7, 10]. Оценки для подзадачи снятия событийно-несобытийной неоднозначности представлены лишь в работе [8], посвященной именно данной проблеме. Там же содержится ссылка на русскоязычное тестовое множество, которое может быть использовано для тестирования и сравнения методов.

Работа имеет следующую структуру. В разделе 1 дается общее описание предлагаемого нами подхода к разрешению событийно-несобытийной неоднозначности, рассматриваются архитектуры нейросетей, которые мы исследовали в роли классификатора. В разделе 2 детально описываются методология оценки, использованные инструменты, данные, значения параметров, характеризующих процесс обучения классификаторов. Наконец, в разделе 3 приводятся результаты оценки на валидационном и тестовом множествах, аналитические выкладки, результаты сопоставления с другими подходами.

## 1. Описание метода разрешения неоднозначности

Основным препятствием на пути практического применения методов разрешения лексической неоднозначности, опирающихся на машинное обучение, является отсутствие размеченных данных. Если для английского и ряда других языков существуют семантически аннотированные корпуса типа SemCor [12] с разметкой в терминах общепринятого инвентаря значений WordNet [13, 14], то для русского языка такого рода корпус в настоящее время недоступен. Семантическое аннотирование трудоемко. Тем не менее представляется возможным использовать для обучения косвенные и приближенные данные.

Применительно к задаче событийно-несобытийной категоризации текстовых единиц в качестве приближенных данных могут выступать контексты однозначных в интересующем нас аспекте слов. Успешный опыт обучения на таких контекстах представлен, в частности, в [8]. В работе предложено использовать *отдельный* классификатор

для каждого неоднозначного слова. В настоящем исследовании мы будем обучать *единый* классификатор, нацеленный на различение «событийных» и «несобытийных» контекстов вообще.

Предлагаемый нами подход заключается в следующем.

- (1) Экспертами выполняется отбор представительных множеств существительных, употребляемых либо только в событийном значении, либо только в несобытийном. Примерами первых могут послужить слова *милитаризация, искоренение, фестиваль*, вторых — *телефон, генерал, надежность, цена, Китай, Иван*. Чем более семантически разнородными будут эти множества, тем лучше.
- (2) Из большого текстового корпуса автоматически извлекаются примеры употребления этих слов в контексте. Формируются два множества текстовых примеров — для событийных и несобытийных слов.
- (3) На полученном множестве текстовых примеров выполняется обучение бинарного классификатора.

Подход имеет много общего с предложенным в [7], но отличается выбором признакового пространства и классификаторов. В качестве классификатора предлагается использовать нейронную сеть, на вход которой подается контекст классифицируемого слова в виде последовательности векторных представлений (эмбеддингов).

В рамках описанного общего подхода мы исследовали несколько нейросетевых архитектур:

- (1) многослойный персептрон,
- (2) два варианта одномерной сверточной сети,
- (3) два варианта рекуррентной двунаправленной сети (с LSTM и GRU ячейками),
- (4) модель из двух LSTM, независимо обрабатывающих левый и правый контексты,
- (5) модель на основе BERT-сети, классифицирующую контекстно-зависимые векторные представления,
- (6) модель на основе BERT-сети, специализированной для задачи языкового моделирования (with LM head).

Для моделей 1–4 контекст составлялся из  $\pm 5$  слов в окрестности классифицируемого. Модели 5–6 в качестве контекста использовали все предложение.

Модели 1–3 тривиальны, их листинги в терминах библиотеки Keras представлены в приложении 1. Особенностью модели 4 является то, что

входящие в ее состав LSTM-слои обрабатывают контекст в направлении от наиболее слабого (отдаленного от классифицируемого слова) к наиболее значимому (ближайшему). Схематическое представление модели 4 изображено на рисунке 1, подробности в листинге 5. LSTM, обрабатывающая левый контекст, рассматривает текст слева направо, а LSTM для правого контекста — справа налево.

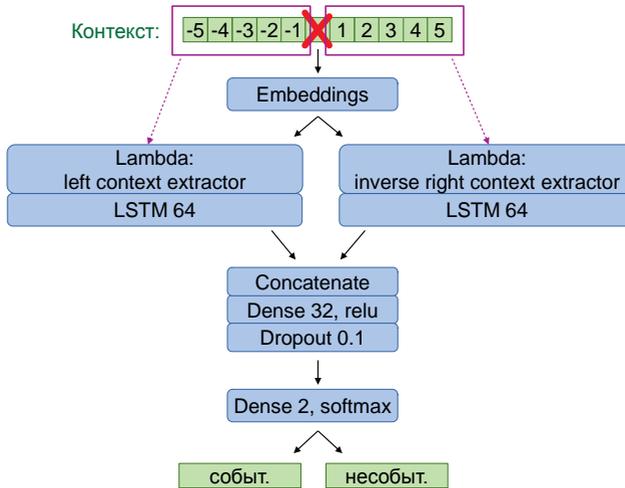


РИСУНОК 1. Модель с двумя LSTM-слоями, ответственными за обработку левого и правого контекстов

Наиболее перспективным для задачи разрешения событийно-несобытийной неоднозначности представлялось использование моделей на основе контекстно-зависимых векторных представлений, таких как ELMo[15] и BERT[16]. В частности, авторы ELMo отмечают, что такие представления «непосредственно могут использоваться в задаче снятия лексической неоднозначности<sup>1</sup>». Для исследования мы выбрали BERT-сеть, адаптированную к русскому языку [17].

Модель 5 представляет собой элементарный классификатор контекстно-зависимых векторных представлений. Мы извлекали векторное представление только из последнего слоя BERT. Так как BERT опирается на WordPiece-подход [18] к токенизации, векторное представление для фрагментированных слов находилось путем усреднения векторных

<sup>1</sup>[...] they can be used without modification to perform well on supervised word sense disambiguation task [...]

представлений фрагментов. Сильной стороной модели 5 является то, что она принимает во внимание само классифицируемое слово, а не классифицирует контекст, как модели 1–4.

Модель 6 на основе *BERT* с *LM-специализацией* использует способность последней предсказывать слово по его контексту. Обучение LM-надстройки выполняется в предположении, что в заданном контексте с какой-то вероятностью может появиться любая словарная единица. На выходе *BERT* с LM позволяет для произвольной текстовой позиции (замаскированного токена) получить оценку уместности появления в данной позиции того или иного слова (а именно, сеть выдает значения *logit*-функции для каждой словарной единицы) — см. рисунок 2.

В [MASK] говорится о важности применения методов ...



logit	Словарь
5.1	документе
4.9	ней
4.8	публикации
...	...
-4.9	медведю
-5	синяя
-5.3	бежит

Рисунок 2. Предсказание слова *BERT*-сетью с LM-надстройкой

В задаче разрешения событийно-несобытийной неоднозначности классифицируемое слово заменялось на специальный маскирующий символ, для которого *BERT* с LM порождала вектор (размером, равным размеру словаря), наполненный *logit*-значениями. В данном векторе сохранялись все положительные значения (в грубом приближении соответствующие возможным словам-кандидатам на подстановку вместо маскирующего символа), остальные значения замещались нулем. После этого получившийся вектор поступал на вход элементарного классификатора. Подробнее см. листинг 7 в приложении. Схематические изображения моделей 5 и 6 приведены соответственно на рисунках 3 и 4.

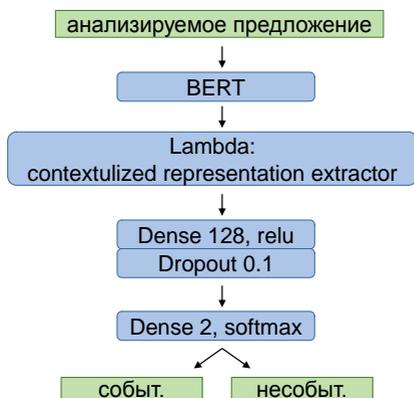


Рисунок 3. Модель BERT-simple — классификатор контекстуализованных векторных представлений

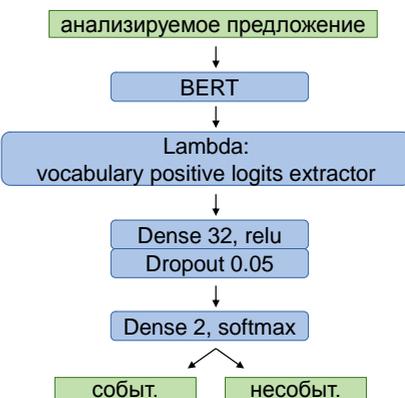


Рисунок 4. Модель BERT-LM — классификатор на основе LM-настройки для BERT

## 2. Методика оценки

Для оценки предложенного метода были вручную составлены списки существительных, однозначных с точки зрения событийно-несобытийной категоризации. В список однозначно событийных существительных вошло 610 слов, несобытийных — 820 слов.

Используя полученные списки, мы для каждого слова извлекли по 5000 примеров их употребления в тексте. Примерами послужили предложения из корпуса PaRuS [19]. На основе данного множества предложений были созданы обучающее и валидационное множества для алгоритма классификации. Разделение на подмножества осуществлялось следующим образом. Примеры употребления для 20 событийных и 20 несобытийных существительных были выделены в валидационное множество, остальные примеры вошли в состав обучающего. Таким образом, классификатор не будет обучаться на контекстах, специфичных для слов, вошедших в валидационное множество. Это позволит лучше оценить обобщающую способность классификатора по сравнению со схемой разделения, когда в валидационное множество выделяется случайное подмножество из обучающего. В общей сложности (с учетом фильтрации, связанной с ограничениями алгоритмов предобработки) размер обучающего множества превысил 6 млн. примеров, а валидационного — 160 тыс.

В качестве тестового множества мы использовали Russian Event Noun Disambiguation Test Set (6x200) [20] — 1200 примеров употребления неоднозначных слов. Данное тестовое множество состоит из примеров для шести слов: *организация*, *отделение*, *отопление*, *публикация*, *стройка*, *управление*. Для каждого слова даны по 100 употреблений в событийном и несобытийном значении.

Так как не-BERT-классификаторы требовали представления текста в векторном виде, в эксперименте нам необходимо было также определиться с выбором конкретного способа векторизации или готовой векторной модели. Для компактности и снижения вычислительной нагрузки мы решили использовать векторную модель для лемм, хотя модель для словоформ предположительно обладает лучшей классифицирующей силой (ср. *публикация автора* и *публикация автором*). Для создания эмбедингов мы воспользовались инструментарием `conll2vec`<sup>2</sup>. Особенностью данного инструмента является то, что создаваемые им векторные представления по существу являются конкатенацией двух векторных представлений меньшей размерности, одно из которых служит для моделирования категориального сходства, а другое — для оценки ассоциативной (тематической) близости. Кроме того, в `conll2vec` в отдельную процедуру выделено обучение векторным представлениям собственных имен; специальным алгоритмом создаются эмбединги для основных знаков препинания. В качестве обучающих данных для `conll2vec` снова использовался корпус PaRuS.

В результате мы построили 100-мерные векторные представления для 114 тыс. лемм. Так как `conll2vec` инструмент новый и малоисследованный, необходимо было удостовериться, что порождаемые им векторные представления сопоставимы по качеству с известными решениями. Мы провели сравнение векторных моделей, построенных при помощи `conll2vec` и `word2vec` [21] на основе данных одного корпуса и со схожими параметрами обучения. Оценка моделей на стандартных тестах RUSSE [22] и RuSim [23] приведена в таблице 1. Мы исключили оценку на AE-тесте, т.к. он требует адаптации для моделей лемм. Оценки свидетельствуют о создании достаточно качественной векторной модели.

Опора на векторную модель лемм требует лемматизации текста перед его преобразованием в последовательность векторов. Для лемматизации тестового множества мы использовали `TreeTagger` [24]. Не будучи самым точным из имеющихся лемматизаторов для русского

---

<sup>2</sup><https://github.com/parus-proj/conll2vec>

Таблица 1. *Оценка качества векторной модели на стандартных тестах. Величина HJ — коэффициент ранговой корреляции Спирмена; для RT, AE2 и RuSim1000 приводится показатель средней точности (average precision). Для word2vec в скобках указан размер окна.*

Инструмент	HJ	RT	AE2	RuSim1000
conll2vec	0.66639	0.83264	0.91126	0.9001
word2vec(10)	0.66931	0.81458	0.93214	0.70673
word2vec(5)	0.67553	0.82222	0.93202	0.73749
word2vec(1)	0.68058	0.83606	0.92895	0.82199

языка [25], TreeTagger менее требователен к вычислительным ресурсам, обеспечивает приемлемое качество анализа и лучше согласуется с данными корпуса PaRuS в части токенизации. Леммы для обучающего и валидационного множества извлекались непосредственно из корпуса PaRuS.

Для классификаторов на основе BERT мы выбрали RuBERT [17]. Данный экземпляр BERT обучен на текстах русскоязычной википедии и новостных сообщениях. Векторные представления 768-мерные. Максимальный размер предложения на входе сети мы ограничили 128 токенами (включая фрагментированные и служебные токены).

Обучение сетей на базе BERT вычислительно затратно, даже если фиксировать веса самой BERT (их свыше 170 млн.). Поэтому для обучения BERT-классификаторов использовались сокращенные обучающее и валидационное множества. Сохранив лексическое разнообразие, мы включили в эти множества всего по 300 предложений для каждого однозначно событийного и однозначно несобытийного слова. В результате размер обучающего множества сократился до 415 тыс. примеров, а валидационного до 11 тыс.

Обучение всех классификаторов выполнялось по единой схеме. Максимальное количество эпох обучения равнялось 25. Использовался механизм ранней остановки в случае, если точность классификации на валидационном множестве не росла в течение 5 эпох. Был задействован механизм контрольных точек (checkpoint) для сохранения модели, продемонстрировавшей максимальную точность в ходе обучения. В дальнейшем эта модель использовалась для получения итоговых оценок на тестовом множестве. Размер пакета (batch): 512 для простых классификаторов, 128 для классификаторов на основе BERT. Начальная

скорость обучения (learning rate):  $1e-3$  для простых классификаторов,  $1e-5$  и  $1e-6$  для классификаторов на основе BERT.

Для детального ознакомления с параметрами экспериментов, воспроизведения и верификации результатов мы подготовили несколько `jupyter notebook` в `google colaboratory`<sup>3</sup>, все использованные данные опубликованы в репозитории `zenodo`<sup>4</sup>.

### 3. Результаты

Результаты оценки классификаторов на валидационном множестве представлены на рисунке 5. Оценки на тестовом множестве сведены в таблицу 2. Во всех случаях оцениваемым параметром была точность классификации (accuracy).

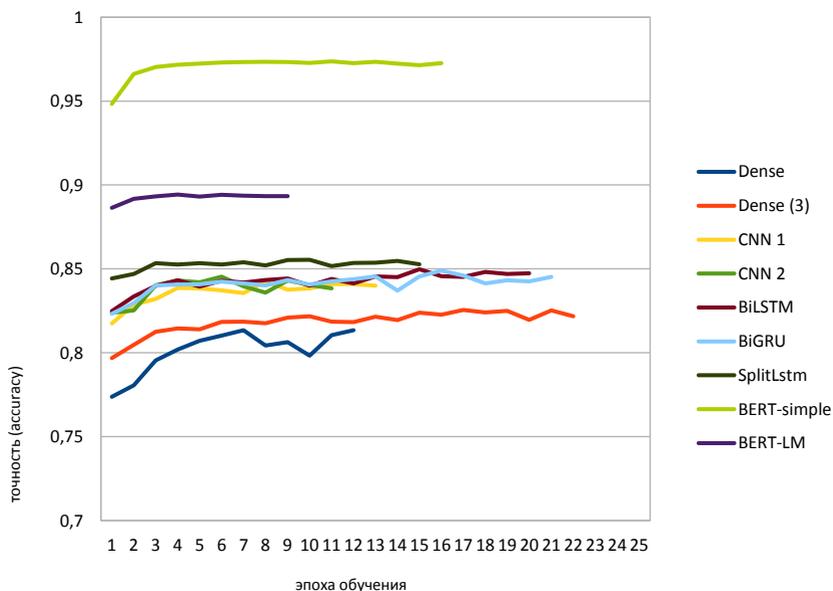


Рисунок 5. Оценка классификаторов на валидационном множестве, точность

Графики для валидационного множества приводят нас к следующим предположениям и выводам.

<sup>3</sup><https://drive.google.com/drive/folders/1CLlvp7Y4usUJjyS2XIirVzblWw50DNui>

<sup>4</sup><https://zenodo.org/record/4030241>

ТАБЛИЦА 2. Оценка точности классификаторов

Модель	Оценки для неоднозначных слов тестового множества						macro- avg
	органи- зация	отде- ление	отоп- ление	публи- кация	строй- ка	управ- ление	
Dense	.755	.79	.761	.758	.745	.72	.755
Dense (3)	.77	.785	.772	.803	.78	.71	.77
CNN 1	.795	.84	.804	.747	.76	.77	.786
CNN 2	.815	.8	.783	.788	.775	.78	.79
BiLSTM	.78	.815	.793	.747	.785	.79	.786
BiGRU	.77	.815	.804	.773	.85	.78	.799
SplitLstm	.805	.86	.821	.773	.81	.775	.807
BERT-simple	.895	.854	.514	.773	.739	.909	.781
BERT-LM	.895	.889	.728	.828	.859	.889	.848

- (1) Многослойный перцептрон (Dense) уступает в точности всем другим классификаторам. Попытка обучения перцептрона с меньшим контекстным окном — модель Dense (3) — более успешна, чем с базовым окном 5, что свидетельствует в пользу недоученности модели Dense.
- (2) Сверточные сети (CNN) достигли практически того же уровня качества, что и рекуррентные (BiLSTM, BiGRU). Это говорит о том, что порядок слов в контексте не является критически значимым признаком в рамках данной задачи классификации.
- (3) Модель из двух LSTM, независимо обрабатывающих левый и правый контексты (SplitLstm), учится быстрее простых двунаправленных рекуррентных сетей, но не имеет над ними весомого преимущества в точности.
- (4) BERT-simple при классификации учитывала не только контекст, но и само классифицируемое слово. Высокие результаты на валидационном множестве могут указывать на то, что контекстуализованные векторные представления классифицируемых слов из валидационного множества имеют сходство с аналогичными представлениями в обучающем множестве (по крайней мере, в некоторых измерениях вектора).
- (5) На валидационном множестве BERT-LM лучшая среди классификаторов, оперирующих только контекстом.

На тестовом множестве (собственно в задаче снятия событийно-несобытийной неоднозначности) наиболее точна оказалась BERT-LM. Обе BERT-модели испытывают затруднения с классификацией значений слова *отопление*. Хотя SplitLstm и уступает по среднему показателю точности, разброс значений от слова к слову у нее меньше, что позволяет предполагать, что эксперименты с другими неоднозначными словами дадут схожие результаты.

В таблицах 3–4 приводятся данные о точности классификации других методов в сравнении с предлагаемым нами.

Таблица 3. *Сопоставление с классификатором [8], точность*

Модель	Оценки для неоднозначных слов тестового множества						macro-avg
	органи-зация	отде-ление	отоп-ление	публи-кация	строй-ка	управ-ление	
BERT-LM	.895	.889	.728	.828	.859	.889	.848
SplitLstm	.805	.86	.821	.773	.81	.775	.807
[8], синтакс.	.677	.8033	.6814	.7957	.7719	.9137	.774

Таблица 4. *Результаты классификатора из системы Evita, точность*

Модель	Оценки для неоднозначных слов тестового множества		
	organization	publication	construction
Evita [11]	.603	.5	.521

Классификатор из [8] оперирует контекстом из слов, синтаксически связанных с классифицируемым. Кроме того, для каждого неоднозначного слова обучался отдельный классификатор. Тем не менее в среднем предложенный нами подход более эффективен.

Система Evita [11] является составной частью пакета TARSQI Toolkit<sup>5</sup>, нацеленного на задачи, связанные с извлечением информации о событиях и темпоральных рассуждениями. Evita предназначена для обнаружения в тексте событийных маркеров, выраженных разными частями речи. Для существительных Evita распознает событийное значение на основе двушаговой процедуры. Сначала слово-кандидат проходит проверку в системе WordNet. Если все значения данного слова относятся к событийной категории (в WordNet авторы Evita выделили группу синсетов верхнего уровня, указывающих на принадлежность к данной категории), то слово помечается как событие. В противном случае ответ дает наивный байесовский классификатор, обученный на корпусах SemCor и TimeBank 1.2. Приводимая разработчиками системы точность обнаружения событийных маркеров (включая глагольные) составляет 74 %.

Мы же провели оценку возможностей Evita в задаче различения событийного и несобытийного значения неоднозначных существительных. Для этого по аналогии с Russian Event Noun Disambiguation Test

<sup>5</sup><https://github.com/tarsqi/ttk>

Set (6x200) было создано тестовое множество для трех неоднозначных английских слов: *construction*, *organization*, *publication*<sup>6</sup>. Вычисленные на этом множестве показатели точности для *Evita* оказались в диапазоне 0.5–0.6.

Полученные нами результаты возможно сравнить также с результатами из [7]. В частности, авторы [7] приводят данные о точности классификации существительных на основе признаков, извлеченных только из контекста классифицируемого слова. В работе речь идет о классификации всех существительных, а не только неоднозначных. Точность классификации не превысила 0.73, причём более чем для 30 % существительных классификация была невозможна.

Заметим, что стопроцентной точности на рассматриваемой задаче достичь, по-видимому, в принципе невозможно. Это связано, в частности, с тем, что отглагольные существительные определенного типа допускают в речи одновременную реализацию своих значений событийного и несобытийного ряда. Вследствие этого в некоторых случаях имеет место не столько неоднозначность (*ambiguity*), требующая непременно разрешения, сколько неопределенность (*vagueness*) [26], при которой выбор между не противоречащими друг другу интерпретациями затруднен или невозможен.

Примеры.

- *Эта публикация вызвала большой резонанс* ('факт публикации' или 'опубликованный материал').
- *В комнате плохое освещение* ('плохие осветительные приборы' или 'плохо освещается').
- *Картина А. А. Дейнеки «На стройке новых цехов»* ('строительство' или 'стройплощадка').

#### 4. Заключение

Представленный в данной работе метод продемонстрировал возможность создания классификатора для различения событийного и несобытийного значений существительных. Метод не требует значительных усилий на разметку данных. Экспериментальная оценка показала, что достижимая методом точность выше, чем у известных аналогичных решений. Классификатор может применяться для автоматической разметки событийных маркеров в тексте, а также снятия лексической неоднозначности в рассматриваемом аспекте.

Перспективным представляется использование метода для разметки текстов с целью получения лексически разнообразного обучающего множества, которое, в свою очередь, позволит достичь еще более

<sup>6</sup><https://zenodo.org/record/4073109>

высокой точности классификации. Кроме того, к повышению точности может привести замена векторных представлений для нормальных форм на представления для текстовых форм. Целесообразно также оценить переносимость метода на другие задачи соотношения слов в тексте с крупными семантическими категориям, такими как абстрактное–конкретное, предметное–ментальное и т.п.

Авторы статьи выражают благодарность ЦКП «Центр данных ДВО РАН» [27] за предоставление вычислительных ресурсов для экспериментов с *BERT*.

### Список литературы

- [1] F. Hogenboom, F. Frasincar, U. Kaymak, F. D. Jong. “An overview of event extraction from text”, Workshop on Detection, Representation and Exploitation of Events in the Semantic Web (DeRiVE) (Bonn, Germany, October 23, 2011), CEUR Workshop Proceedings, vol. **779**, 2011, pp. 48–57.  [↑](#)<sub>31</sub>
- [2] W. Xiang, B. Wang. “A survey of event extraction from text”, *IEEE Access*, **7** (2019), pp. 173111–173137.  [↑](#)<sub>31</sub>
- [3] Z. Subecz. “Automatic Detection of nominal events in Hungarian texts with dependency parsing and WordNet”, ICIST 2016: Information and Software Technologies, Communications in Computer and Information Science, vol. **639**, eds. G. Dregvaite, R. Damasevicius, Springer, Cham, 2016, ISBN 978-3-319-46253-0, pp. 580–592.  [↑](#)<sub>31,32</sub>
- [4] J. Allan, *Topic Detection and Tracking: Event-Based Information Organization*, The Information Retrieval Series, vol. **12**, Springer Science & Business Media, 2012, ISBN 978-1-4613-5311-9.  [↑](#)<sub>31</sub>
- [5] A. Lefevre-Halftermeyer, J.-Y. Antoine, A. Couillault, E. Schang, L. Abouda, A. Savary, D. Maurel, I. Eshkol, D. Battistelli. “Covering various needs in temporal annotation: a proposal of extension of ISO TimeML that preserves upward compatibility”, *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, LREC’16, European Language Resources Association (ELRA), 2016, pp. 3802–3806.  [↑](#)<sub>31</sub>
- [6] N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen, J. Pustejovsky. “SemEval-2013 Task 1: TempEval-3: Evaluating Time expressions, events, and temporal relations”, *Second Joint Conference on Lexical and Computational Semantics (\*SEM)*. V. 2: *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Association for Computational Linguistics, 2013, pp. 1–9.  [↑](#)<sub>31</sub>
- [7] C. Creswell, M. J. Beal, J. Chen, Th. L. Cornell, L. Nilsson, R. K. Srihari. “Automatically Extracting Nominal Mentions of Events with a Bootstrapped Probabilistic Classifier”, *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions* (Sydney, Australia, July, 2016), Association for Computational Linguistics, 2006, pp. 168–175.  [↑](#)<sub>31,32,33,42</sub>
- [8] И. В. Трофимов, Е. А. Сулейманова, Н. А. Власова, А. В. Подобреев.

- «Разрешение событийно-несобытийной неоднозначности существительных», *Программные системы: теория и приложения*, **9:4(39)** (2018), с. 3–33 (in Russian).  [↑<sub>32,41</sub>](#)
- [9] E. Tadesse, R. Tsegaye, K. Qaqqabaa. “Event extraction from unstructured amharic text”, *Proceedings of The 12th Language Resources and Evaluation Conference* (Marseille, France, May, 2020), European Language Resources Association, 2020, pp. 2103–2109.  [↑<sub>32</sub>](#)
- [10] B. Arnulphy, V. Claveau, X. Tannier, A. Vilnat. “Supervised machine learning techniques to detect TimeML events in French and English”, *NLDB 2015: Natural Language Processing and Information Systems* (Passau, Germany, June 17–19, 2015), *Lecture Notes in Computer Science*, vol. **9103**, eds. Ch. Biemann, S. Handschuh, A. Freitas, F. Meziane, E. Métais, Springer, Cham, 2015, ISBN 978-3-319-19580-3, pp. 19–32.  [↑<sub>32</sub>](#)
- [11] R. Sauri, R. Knippen, M. Verhagen, J. Pustejovsky. “Evita: a RobustEvent recognizer for QA Systems”, *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, HLT/EMNLP 2005* (Vancouver, British Columbia, Canada, October, 2005), pp. 700–707.  [↑<sub>32,41</sub>](#)
- [12] T. Petrolito, F. Bond. “A survey of WordNet annotated corpora”, *Proceedings of the Seventh Global Wordnet Conference* (Tartu, Estonia, January, 2014), University of Tartu Press, 2014, pp. 236–245.  [↑<sub>32</sub>](#)
- [13] G. A. Miller. “WordNet: a lexical database for English”, *Communications of the ACM*, **38:11** (1995), pp. 39–41.  [↑<sub>32</sub>](#)
- [14] *WordNet: An Electronic Lexical Database*, ed. Ch. Fellbaum, MIT Press, 1998, ISBN 9780262061971, 449 pp. [↑<sub>32</sub>](#)
- [15] M. Peters, M. Neumann, M. Iyyer, M. Gardner, Ch. Clark, K. Lee, L. Zettlemoyer. “Deep contextualized word representations”, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. V. 1* (New Orleans, Louisiana, June, 2018), Association for Computational Linguistics, 2018, pp. 2227–2237.   [↑<sub>34</sub>](#)
- [16] J. Devlin, M-W. Chang, K. Lee, K. Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for language understanding*, 2018, 16 pp. arXiv:[1810.04805](#)[cs.CL] [↑<sub>34</sub>](#)
- [17] Yu. Kuratov, M. Arkhipov. *Adaptation of Deep Bidirectional Multilingual Transformers for Russian language*, 2019, 8 pp. arXiv:[1905.07213](#)[cs.CL] [↑<sub>34,38</sub>](#)
- [18] Y. Wu, M. Schuster, Zh. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, J. Dean. “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation”, *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018. [↑<sub>34</sub>](#)

- [19] Н. А. Власова, И. В. Трофимов, Ю. П. Сердюк, Е. А. Сулейманова, И. Н. Воздвиженский. «PaRuS — синтаксически аннотированный корпус русского языка», *Программные системы: теория и приложения*, 10:4(43) (2019), с. 181–199.    <sup>36</sup>
- [20] I. V. Trofimov, N. A. Vlasova, E. A. Suleymanova, A. V. Podobryaev. *Russian event noun disambiguation test set (6x200)*, Zenodo, 2018.   <sup>37</sup>
- [21] T. Mikolov, K. Chen, G. Corrado, J. Dean. *Efficient estimation of word representations in vector space*, 2013, 12 pp. arXiv:1301.3781[cs.CL]  <sup>37</sup>
- [22] А. Панченко, Н. В. Лукашевич, Д. Усталов, Д. Паперно, К. М. Мейер, Н. Константинова. «RUSSE: семинар по оценке семантической близости для русского языка», *Annual International Conference «Dialogue»*, RGGU, 2015, с. 89–105 (in English).   <sup>37</sup>
- [23] И. В. Трофимов, Е. А. Сулейманова. «Дистрибутивная модель для различения семантического сходства и ассоциации», *International Conference «Dialogue», Computational Linguistics and Intellectual Technologies*, т. 1(16), 2017, с. 349–359 (in English).    <sup>37</sup>
- [24] H. Schmid. “Probabilistic part-of-speech tagging using decision trees”, *Proceedings of the Conference on New Methods in Language Processing* (Manchester, UK, 1994), 1994, pp. 44–49.   <sup>37</sup>
- [25] О. Н. Ляшевская, Т. О. Шаврина, И. В. Трофимов, Н. А. Власова. «GramEval 2020: дорожка по автоматическому морфологическому и синтаксическому анализу русских текстов», *Annual International Conference «Dialogue»*, 19, 2020, с. 553–569 (in English).    <sup>38</sup>
- [26] Анна А. Зализняк. «Феномен многозначности и способы его описания», *Вопросы языкознания*, 2004, №2, с. 20–45.   <sup>42</sup>
- [27] А. А. Сорокин, С. В. Макогонов, С. П. Королев. «Информационная инфраструктура для коллективной работы ученых Дальнего Востока России», *Научно-техническая информация. Сер. 1: Организация и методика информационной работы*, 2017, №12, с. 14–16.   <sup>43</sup>

## Приложение 1

### Листинг 1. Модель многослойного перцептрона

```

model = Sequential([
Embedding( vocab_size, embedding_size, input_length=seq_len,
weights=[embedding_weights], trainable=False ),
Reshape((-1,)),
Dense(128, activation="relu"),
Dropout(0.1),
Dense(64, activation="relu"),
Dropout(0.1),
Dense(2, activation="softmax")
])

```

**Листинг 2. Модель сверточной сети: вариант 1**

```

model = Sequential([
    Embedding( vocab_size, embedding_size, input_length=seq_len,
               weights=[embedding_weights], trainable=False ),
    Conv1D(filters=128, kernel_size=3, activation="relu"),
    MaxPooling1D(pool_size=2),
    Conv1D(filters=1024, kernel_size=3, activation="relu"),
    GlobalMaxPooling1D(),
    Dense(64, activation="relu"),
    Dropout(0.1),
    Dense(2, activation="softmax")
])

```

**Листинг 3. Модель сверточной сети: вариант 2**

```

model = Sequential([
    Embedding( vocab_size, embedding_size, input_length=seq_len,
               weights=[embedding_weights], trainable=False ),
    Conv1D(filters=128, kernel_size=3, activation="relu"),
    Conv1D(filters=256, kernel_size=3, activation="relu"),
    Conv1D(filters=512, kernel_size=3, activation="relu"),
    GlobalMaxPooling1D(),
    Dense(64, activation="relu"),
    Dropout(0.1),
    Dense(2, activation="softmax")
])

```

**Листинг 4. Модель рекуррентной LSTM-сети. Модель с GRU аналогична и отличается лишь типом ячейки**

```

model = Sequential([
    Embedding( vocab_size, embedding_size, input_length=seq_len,
               weights=[embedding_weights], trainable=False ),
    Bidirectional(LSTM( 64, dropout=0.1 )),
    Dense(2, activation="softmax")
])

```

### Листинг 5. Модель из двух LSTM, независимо обрабатывающих левый и правый контексты

```

common_input = Input(shape=(2*window_size,))
emb_layer = Embedding(vocab_size, embedding_size, input_length=2*window_size,
                      weights=[embedding_weights], trainable=False)(common_input)
left_ctx = Lambda(lambda x : x[:, :window_size, :]) (emb_layer)
left_lstm = LSTM(64) (left_ctx)
right_ctx = Lambda(lambda x : x[:, :-window_size-1:-1, :]) (emb_layer)
right_lstm = LSTM(64) (right_ctx)
merge_layer = Concatenate() ([left_lstm, right_lstm])
dense = Dense(32, activation="relu") (merge_layer)
drop = Dropout(0.1) (dense)
softmax_layer = Dense(2, activation="softmax") (drop)
model = Model( inputs=common_input, outputs=softmax_layer )

```

### Листинг 6. Модель на основе BERT-сети, классифицирующая контекстно-зависимые векторные представления

```

@tf.function
def range2indices(r):
    return tf.range(r[0], r[1])

def extr_target_embedding(x):
    last_hidden_state_seq = x[0]
    ranges = x[1]
    result = tf.map_fn( lambda t: tf.math.reduce_mean(
        tf.gather(t[0], range2indices(t[1]), axis=0), axis=0),
        (last_hidden_state_seq, ranges),
        fn_output_signature=tf.float32
    )
    return result

def create_model_emb(model_name):
    encoder = TFBertModel.from_pretrained(model_name, from_pt=True, trainable=False)
    input_ids = Input(shape=(MAX_BERT_INPUT_LEN,), dtype=tf.int32)
    tti = Input(shape=(MAX_BERT_INPUT_LEN,), dtype=tf.int8)
    atm = Input(shape=(MAX_BERT_INPUT_LEN,), dtype=tf.int8)
    cwi = Input(shape=(2,), dtype=tf.int32, name="classified_word_indices")
    embedding = encoder(input_ids, token_type_ids=tti, attention_mask=atm)
    extr = Lambda(extr_target_embedding) ([embedding[0], cwi])
    dense = Dense(128, activation="relu") (extr)
    drop = Dropout(0.1) (dense)
    softmax_layer = Dense(2, activation="softmax") (drop)

```

```

model = Model( inputs=[input_ids, tti, atm, cwi],
               outputs=softmax_layer )
return model

```

### Листинг 7. Модель на основе BERT-сети с LM-надстройкой

```

extr_relu = lambda x: tf.nn.relu( tf.squeeze(
    tf.gather(x[0][0], x[1], axis=1, batch_dims=1), [1] ) )

def create_model_voc(model_name):
    encoder = TFAutoModelForMaskedLM.from_pretrained( model_name,
                                                       from_pt=True,
                                                       trainable=False )
    input_ids = Input(shape=(MAX_BERT_INPUT_LEN,), dtype=tf.int32)
    tti = Input(shape=(MAX_BERT_INPUT_LEN,), dtype=tf.int8)
    atm = Input(shape=(MAX_BERT_INPUT_LEN,), dtype=tf.int8)
    cwi = Input(shape=(1,), dtype=tf.int32, name="classified_word_index")
    embedding = encoder(input_ids, token_type_ids=tti, attention_mask=atm)
    extr = Lambda(extr_relu, name="words_distrib_extractor") ( [embedding, cwi] )
    dense_ab = Dense(64, activation="relu") (extr)
    drop_ab = Dropout(0.05) (dense_ab)
    softmax_layer = Dense(2, activation="softmax") (drop_ab)
    model = Model( inputs=[input_ids, tti, atm, cwi],
                  outputs=softmax_layer )
    return model

```

Поступила в редакцию 14.10.2020

Переработана 12.11.2020

Опубликована 28.11.2020

Рекомендовал к публикации

*к.т.н. С. А. Амелкин*

*Пример ссылки на эту публикацию:*

И. В. Трофимов, Ю. П. Сердюк, Е. А. Сулейманова, Н. А. Власова.  
 «Разрешение событийно-несобытийной неоднозначности существительных:  
 нейросетевой подход». *Программные системы: теория и приложения*, 2020,  
 11:4(47), с. 31–53.  10.25209/2079-3316-2020-11-4-31-53

 [http://psta.pspiras.ru/read/psta2020\\_4\\_31-53.pdf](http://psta.pspiras.ru/read/psta2020_4_31-53.pdf)

*Об авторах:*



### **Игорь Владимирович Трофимов**

старший научный сотрудник Исследовательского центра искусственного интеллекта ИПС им. А. К. Айламазяна, специалист по технологиям автоматической обработки естественного языка, извлечения информации, автоматического планирования.

 0000-0002-6903-4730

e-mail: itrofimov@gmail.com



### **Юрий Петрович Сердюк**

старший научный сотрудник Исследовательского центра искусственного интеллекта ИПС им. А. К. Айламазяна, научные интересы: параллельное программирование, формальные исчисления процессов, системы типов.

 0000-0003-2916-2102

e-mail: Yuri@serdyuk.botik.ru



### **Елена Анатольевна Сулейманова**

научный сотрудник Исследовательского центра искусственного интеллекта ИПС им. А. К. Айламазяна, одна из разработчиков технологии построения систем извлечения информации.

 0000-0002-0792-9651

e-mail: yes@helen.botik.ru



### **Наталья Александровна Власова**

младший научный сотрудник Исследовательского центра искусственного интеллекта ИПС им. А. К. Айламазяна РАН, одна из разработчиков технологии построения систем извлечения информации.

 0000-0002-7843-6870

e-mail: nathalie.vlassova@gmail.com

CSCSTI 16.31.21  
UDC 004.89:81'322.2

Igor Trofimov, Yuri Serdyuk, Elena Suleymanova, Natalia Vlasova. *Eventive vs. non-eventive sense of nouns: disambiguation using neural network approach.*

ABSTRACT. The paper addresses the issue of automatic disambiguation of event nominals. Such nouns account for a large proportion of event mentions in text and therefore are, together with verbs, of relevance to the task of event extraction. Since event-denoting nouns are often polysemous between ‘eventuality’ and ‘non-eventuality’ senses, disambiguation is a critical step in event recognition. We expect that the suggested disambiguation method will contribute to the accuracy of event extraction from text.

Lack of labelled data is a well-known impediment to machine-learning word sense disambiguation. To handle this problem, we used a semi-supervised technique. Two sets of unambiguous event- and entity-denoting nouns were created by hand (610 and 820 nouns resp.). From a large text corpus (PaRuS, 2.6 B tokens), we extracted 5 000 sentences per noun and split this set of contexts into two disjoint subsets: the validation set (contexts for 20 event-denoting nouns and 20 non-event nouns) and the training set (contexts for the rest of the nouns). We used the training set to train eight neural network classifiers of different architecture (MLP, CNN, RNN, BERT+MLP). For evaluation of this method, we tested the trained classifiers on the Russian Event Noun Disambiguation Test Set. The BERT-based model achieved the highest average accuracy of 84.8%.

*Key words and phrases:* detection of events in the text, resolution of lexical ambiguity, neural network, natural language processing.

2010 *Mathematics Subject Classification:* 68T50; 68T05, 91F20

## References

- [1] F. Hogenboom, F. Frasinca, U. Kaymak, F. D. Jong. “An overview of event extraction from text”, Workshop on Detection, Representation and Exploitation of Events in the Semantic Web (DeRiVE) (Bonn, Germany, October 23, 2011), CEUR Workshop Proceedings, vol. **779**, 2011, pp. 48–57. [URL](#)<sup>↑</sup><sub>31</sub>
- [2] W. Xiang, B. Wang. “A survey of event extraction from text”, *IEEE Access*, **7** (2019), pp. 173111–173137. [DOI](#)<sup>↑</sup><sub>31</sub>
- [3] Z. Subecz. “Automatic Detection of nominal events in Hungarian texts with dependency parsing and WordNet”, ICIST 2016: Information and Software Technologies, Communications in Computer and Information Science, vol. **639**, eds. G. Dregvaite, R. Damasevicius, Springer, Cham, 2016, ISBN 978-3-319-46253-0, pp. 580–592. [DOI](#)<sup>↑</sup><sub>31, 32</sub>
- [4] J. Allan, *Topic Detection and Tracking: Event-Based Information Organization*, The Information Retrieval Series, vol. **12**, Springer Science & Business Media, 2012, ISBN 978-1-4613-5311-9. [DOI](#)<sup>↑</sup><sub>31</sub>
- [5] A. Lefevre-Halftermeyer, J.-Y. Antoine, A. Couillault, E. Schang, L. Abouda, A. Savary, D. Maurel, I. Eshkol, D. Battistelli. “Covering various needs in temporal annotation: a proposal of extension of ISO TimeML that preserves upward compatibility”, *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, LREC’16, European Language Resources Association (ELRA), 2016, pp. 3802–3806. [URL](#)<sup>↑</sup><sub>31</sub>
- [6] N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen, J. Pustejovsky. “SemEval-2013 Task 1: TempEval-3: Evaluating Time expressions, events, and temporal relations”, *Second Joint Conference on Lexical and Computational Semantics (\*SEM)*. V. 2: *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Association for Computational Linguistics, 2013, pp. 1–9. [URL](#)<sup>↑</sup><sub>31</sub>
- [7] C. Creswell, M. J. Beal, J. Chen, Th. L. Cornell, L. Nilsson, R. K. Srihari. “Automatically Extracting Nominal Mentions of Events with a Bootstrapped Probabilistic Classifier”, *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions* (Sydney, Australia, July, 2016), Association for Computational Linguistics, 2006, pp. 168–175. [URL](#)<sup>↑</sup><sub>31, 32, 33, 42</sub>
- [8] I. V. Trofimov, Ye. A. Suleymanova, N. A. Vlasova, A. V. Podobryayev. “Disambiguation between eventive and non-eventive meaning of nouns”, *Program Systems: Theory and Applications*, **9:4**(39) (2018), pp. 3–33 (in Russian). [DOI](#)<sup>↑</sup><sub>32, 41</sub>
- [9] E. Tadesse, R. Tsegaye, K. Qaqqabaa. “Event extraction from unstructured amharic text”, *Proceedings of The 12th Language Resources and Evaluation Conference* (Marseille, France, May, 2020), European Language Resources Association, 2020, pp. 2103–2109. [URL](#)<sup>↑</sup><sub>32</sub>
- [10] B. Arnulphy, V. Claveau, X. Tannier, A. Vilnat. “Supervised machine learning techniques to detect TimeML events in French and English”, NLDB 2015: Natural Language Processing and Information Systems (Passau, Germany, June 17–19, 2015), Lecture Notes in Computer Science, vol. **9103**, eds. Ch. Biemann, S. Handschuh, A. Freitas, F. Mezziane, E. Métais, Springer, Cham, 2015, ISBN 978-3-319-19580-3, pp. 19–32. [DOI](#)<sup>↑</sup><sub>32</sub>
- [11] R. Saufi, R. Knippen, M. Verhagen, J. Pustejovsky. “Evita: a RobustEvent recognizer for QA Systems”, *Proceedings of Human Language Technology Conference and*

- Conference on Empirical Methods in Natural Language Processing, HLT/EMNLP 2005* (Vancouver, British Columbia, Canada, October, 2005), pp. 700–707. [URL](#)<sup>↑</sup><sub>32,41</sub>
- [12] T. Petrolito, F. Bond. “A survey of WordNet annotated corpora”, *Proceedings of the Seventh Global Wordnet Conference* (Tartu, Estonia, January, 2014), University of Tartu Press, 2014, pp. 236–245. [DOI](#)<sup>↑</sup><sub>32</sub>
- [13] G. A. Miller. “WordNet: a lexical database for English”, *Communications of the ACM*, **38**:11 (1995), pp. 39–41. [DOI](#)<sup>↑</sup><sub>32</sub>
- [14] *WordNet: An Electronic Lexical Database*, ed. Ch. Fellbaum, MIT Press, 1998, ISBN 9780262061971, 449 pp.<sup>↑</sup><sub>32</sub>
- [15] M. Peters, M. Neumann, M. Iyyer, M. Gardner, Ch. Clark, K. Lee, L. Zettlemoyer. “Deep contextualized word representations”, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. V. 1* (New Orleans, Louisiana, June, 2018), Association for Computational Linguistics, 2018, pp. 2227–2237. [DOI](#)<sup>↑</sup><sub>34</sub>
- [16] J. Devlin, M-W. Chang, K. Lee, K. Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for language understanding*, 2018, 16 pp. arXiv:[1810.04805](#)<sup>~</sup>[cs.CL]<sup>↑</sup><sub>34</sub>
- [17] Yu. Kuratov, M. Arkhipov. *Adaptation of Deep Bidirectional Multilingual Transformers for Russian language*, 2019, 8 pp. arXiv:[1905.07213](#)<sup>~</sup>[cs.CL]<sup>↑</sup><sub>34,38</sub>
- [18] Y. Wu, M. Schuster, Zh. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X Liu, Ł. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, J. Dean. “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation”, *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.<sup>↑</sup><sub>34</sub>
- [19] N. A. Vlasova, I. V. Trofimov, Yu. P. Serdyuk, Ye. A. Suleymanova, I. N. Vozdvizhenskiy. “PaRuS — syntax annotated Russian corpus”, *Program Systems: Theory and Applications*, **10**:4(43) (2019), pp. 181–199 (in Russian). [DOI](#)<sup>↑</sup><sub>36</sub>
- [20] I. V. Trofimov, N. A. Vlasova, E. A. Suleymanova, A. V. Podobryaev. *Russian event noun disambiguation test set (6x200)*, Zenodo, 2018. [DOI](#)<sup>↑</sup><sub>37</sub>
- [21] T. Mikolov, K. Chen, G. Corrado, J. Dean. *Efficient estimation of word representations in vector space*, 2013, 12 pp. arXiv:[1301.3781](#)<sup>~</sup>[cs.CL]<sup>↑</sup><sub>37</sub>
- [22] A. Panchenko, N. V. Lukashevich, D. Ustalov, D. Paperno, K. M. Meyyer, N. Konstantinova. “RUSSE: The First Workshop on Russian semantic similarity, computational linguistics and intellectual technologies”, *Annual International Conference “Dialogue”*, RGGU, 2015, pp. 89–105. [URL](#)<sup>↑</sup><sub>37</sub>
- [23] I. V. Trofimov, Ye. A. Suleymanova. “A syntax-based distributional model for discriminating between semantic similarity and association”, *International Conference “Dialogue”, Computational Linguistics and Intellectual Technologies*, vol. **1**(16), 2017, pp. 349–359 (in English). [URL](#)<sup>↑</sup><sub>37</sub>
- [24] H. Schmid. “Probabilistic part-of-speech tagging using decision trees”, *Proceedings of the Conference on New Methods in Language Processing* (Manchester, UK, 1994), 1994, pp. 44–49. [URL](#)<sup>↑</sup><sub>37</sub>
- [25] O. N. Lyashevskaya, T. O. Shavrina, I. V. Trofimov, N. A. Vlasova. “GramEval 2020 shared task: Russian full morphology and universal dependencies parsing”, *Annual International Conference “Dialogue”*, 19, 2020, pp. 553–569. [DOI](#)<sup>↑</sup><sub>38</sub>

- [26] Anna A. Zaliznyak. “The phenomenon of polysemy and ways of their description”, *Voprosy yazykoznaniya*, 2004, no. 2, pp. 20–45 (in Russian).<sup>↑</sup><sub>42</sub>
- [27] A. A. Sorokin, S. V. Makogonov, S. P. Korolev. “Informatsionnaya infrastruktura dlya kollektivnoy raboty uchenykh Dal’nego Vostoka Rossii”, *Nauchno-tekhnicheskaya informatsiya. Ser. 1: Organizatsiya i metodika informatsionnoy raboty*, 2017, no. 12, pp. 14–16 (in Russian).<sup>↑</sup><sub>43</sub>

*Sample citation of this publication:*

Igor Trofimov, Yuri Serdyuk, Elena Suleymanova, Natalia Vlasova. “Eventive vs. non-eventive sense of nouns: disambiguation using neural network approach”. *Program Systems: Theory and Applications*, 2020, **11**:4(47), pp. 31–53. (In Russian).

 10.25209/2079-3316-2020-11-4-31-53

 [http://psta.psir.ru/read/psta2020\\_4\\_31-53.pdf](http://psta.psir.ru/read/psta2020_4_31-53.pdf)