УДК 004.89 DOI 10.18522/2311-3103-2021-3-92-101

Ю.А. Кравченко, А.М. Мансур, Ж.Х. Мохаммад

МОДИФИЦИРОВАННЫЙ МЕТОД УСТРАНЕНИЯ НЕОДНОЗНАЧНОСТИ СМЫСЛА СЛОВ, ОСНОВАННЫЙ НА МЕТОДАХ РАСПРЕДЕЛЕННОГО ПРЕДСТАВЛЕНИЯ*

В задачах интеллектуального анализа текста текстовое представление должно быть не только эффективным, но и интерпретируемым, поскольку это позволяет понять операционную логику, лежащую в основе моделей интеллектуального анализа данных. В этой статье предлагается модифицированный метод устранения неоднозначности слов (WSD), который, по сути, имитирует хорошо известный вариант подхода Леска WSD. Для выбранного слова и его контекста алгоритм Леска проводит свои вычисления на основе проверки совпадений контекста слова и каждого определения его смыслов (глосс), для того чтобы выбрать правильное значение. Основным преимуществом данного метода является применение концепции сходства между определением и контекстом вместо «перекрытия», для каждого смысла целевого слова в дополнение к расширению определения примерами предоставленными WordNet. Предлагаемый метод также характеризуется использованием функций измерения схожести текстов, определенных в распределенном семантическом пространстве. Предлагаемый метод протестирован на пяти различных наборах эталонных данных для задачи устранения неоднозначности смысла слов и сравнивался с несколькими базовыми методами, включая Lesk, расширенный Lesk, WordNet 1st sense, Babelfy и UKB. Результаты показывают, что предлагаемый метод превосходит большинство известных аналогов, за исключением методов Babelfy и WN 1st sense.

Устранение неоднозначности слов; WSD; semEval; WordNet; сходство текста; интеллектуальный анализ текста.

Yu.A. Kravchenko, A.M. Mansour, J.H. Mohammad

MODIFIED WORD SENSE DISAMBIGUATION METHOD BASED ON DISTRIBUTED REPRESENTATION METHODS

In the text mining tasks, textual representation should be not only efficient but also interpretable, as this enables an understanding of the operational logic underlying the data mining models. This paper describes a modified Word Sense Disambiguation (WSD) method which extends two well-known variations of the Lesk WSD approach. Given a word and its context, Lesk bases its calculations on the overlap between the context of a word and each definition of its senses (gloss) in order to select the proper meaning. The main contribution of the proposed method is the adoption of the concept of "similarity" between definition and context instead of "overlap", in addition to expanding the definition with examples provided by WordNet for each sense of the target word. The proposed method is also characterized by the use of text similarity measurement functions defined in a distributed semantic space. The proposed method has been tested on five different benchmark datasets for words sense disambiguation tasks and compared with several basic methods, including simple Lesk, extended Lesk, WordNet 1st sense, Babelfy and UKB. The results show that proposed method outperforms most basic methods with the exception of Babelfy and the WN 1st sense methods.

Word sense disambiguation; WordNet; knowledge-based; WSD; semEval; text similarity; text mining.

Введение. Устранение неоднозначности слов WSD (англ. Word Sense Disambiguation) является открытой проблемой в компьютерной лингвистике, ее задача состоит в том, чтобы автоматически определить конкретное значение многозначного слова (смысл, чувства) в данном контексте [1, 2].

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 18-29-22019.

Устранение неоднозначности слов (WSD) используется практически во всех приложениях языковых технологий, как машинный перевод, поиск информации, интеллектуальный анализ текста и извлечение информации (IE). При этом применяется в качестве неотъемлемой части извлечения концептов, что считается базовым этапом в таких задачах, как поиск информации, построение профилей пользователей при персонализации веб-контента и рекомендации на основе контента, в дополнение к его приложениям в задачах классификации и кластеризации текстов.

В зависимости от источника знаний, используемого при устранении неоднозначности слов, подходы WSD подразделяются на подходы, основанные на знаниях, подходы с учителем и без учителя. Подходы WSD с учителем / без учителя применяются для получения публичных оценок [3], но для этого требуются большие объемы данных с ручными тегами, создание которых обычно является очень трудоемким процессом.

В качестве альтернативы этим подходам подходы WSD, основанные на знаниях (англ. knowledge-based), используют информацию, содержащуюся в лексической базе знаний (LKB), для выполнения WSD без использования каких-либо дополнительных свидетельств из корпуса. Среди них популярными стали две категории алгоритмов, основанных на знаниях: методы на основе перекрытия и графов. Первый обязан своим успехом простой интуиции, лежащей в основе этого семейства алгоритмов, в то время как распространение второго начало расти после развития семантических сетей.

Алгоритмы на основе перекрытия происходят от алгоритма Леска [4], который вдохновил целое семейство методов, использующих количество общих слов в двух определениях чувств (глоссах) для выбора правильного значения в контексте. Глоссы играют ключевую роль в алгоритме Леска, который использует только два типа информации: 1) набор словарных статей (синсетов), по одной для каждого возможного значения слова, и 2) информацию о контексте, в котором встречается слово. Идея проста: для двух слов алгоритм выбирает те смыслы (синсеты), определения которых имеют максимальное перекрытие, то есть наибольшее количество общих слов в определении смыслов (синсетов).

Этот подход имеет два недостатка: 1) сложность, т.к. количество сравнений комбинаторно увеличивается с количеством слов в тексте; 2) выразительность определения, т.к. перекрытие основано только на совпадении слов в глоссах. Первый недостаток решается с помощью «упрощенной» версии алгоритма Леска [5], которая устраняет неоднозначность каждого слова отдельно. Для данного слова выбирается значение глосса, которое показывает максимальное совпадение с текущим контекстом, представленным окружающими словами. Упрощенный алгоритм Леска значительно превосходит исходный алгоритм Леска, как было доказано в работе [6]. Второй недостаток исключили Банерджи и Педерсен [7], предложившие «адаптированный» алгоритм Леска. Адаптированная вариация использует отношения между значениями: каждый глосс дополняется определениями семантически связанных значений. Кроме того, в [8] представлен метод *Lesk* с вложением, где используется функция сходства слов, определенная в распределенном семантическом пространстве, для вычисления перекрытия глосс-контекст, при этом создается вектор для представления каждого текста из группы составляющих его векторов слов, а затем применяются методы уменьшения размеров. Этот метод превосходит другие версии Lesk, но его недостатком является высокая вычислительная сложность процесса парного сопоставления слов, которая растет при увеличении количества используемых слов текста.

Несмотря на описанные улучшения, алгоритмы, основанные на перекрытии, не отставали от показателей, достигнутых с помощью подходов, основанных на графах. Эти подходы используют хорошо известные методы на основе графов для

поиска и использования структурных свойств графа, лежащего в основе конкретного LKB. Их способность устраняет неоднозначность сразу всех слов в последовательности, при этом используя существующие взаимосвязи (границы) между чувствами (узлами), сделала эти алгоритмы более принципиальными, чем методы, использующие перекрытия определения смысла. Однако, поскольку граф анализируется в целом, они требуют больших вычислительных затрат, используя такие большие графы как WordNet, в дополнение к тому факту, что полностью игнорируется роль определений в процессе устранения неоднозначности слов.

Недавно распределенные методы представления текстов, таких как встроенные слова, достигли современных результатов в задачах text mining, и особенно в задачах оценки сходства текстов. Используя эти методы, каждый текст представлен как низкомерный численный вектор, а затем сходство текстов можно просто измерять с использованием косинусной меры сходства. Соответственно, в данной работе предлагается разработать метод устранения неоднозначности слов, который имитирует метод *Lesk* (с вложением), с той разницей, что, с одной стороны, он применяет методы вложения на уровне текста при векторизации текста контекста и определений и представляет их как плотные низкоразмерные векторы напрямую, без необходимости уменьшения шага размерности, а с другой стороны, расширяет текст определения, чтобы включить примеры, предоставленные базой данных WordNet, что позволяет увеличить выразительность результирующих векторов.

Предложенный метод превосходит все версии *Lesk* и дает результаты, которые конкурируют с другими методами устранения неоднозначности слов.

1. Аналитический обзор методов устранение неоднозначности слов. *Лексическая база данных WordNet* используется как источник семантической информации. Для каждого понятия WordNet предоставляет список различных синсетов, каждое с определением (глоссом), объясняющим его значение. Каждый синсет содержит список синонимов или синонимичных словосочетаний и указатели, описывающие отношения между ним и другими синсетами. Определение правильного синонима для каждого термина требует шага обнаружения неоднозначности. Ниже приведены базовые методы WSD, основанные на знаниях.

Первое чувство WordNet (WN 1st sense). Эта базовая линия просто выбирает кандидата, который считается первым смыслом (*чувством*) в WordNet 3.0. Несмотря на простоту этого метода, он дает хорошие результаты [2].

Леск (простой) [4], представляет собой простой алгоритм WSD, основанный на знаниях, который основывает свои вычисления на расчете перекрытия между контекстом целевого слова и его определениями, которые даны в реестре смыслов. Основываясь на том же принципе, в различных работах был адаптирован исходный алгоритм с учетом определений из связанных слов (*расширенный Леск)* [9] или путем вычисления распределительного сходства между определениями и контекстом целевого слова [8].

Леск (с вложением) [8]. В основе этого подхода лежит упрощенный алгоритм *Леска*, однако он использует функцию сходства слов, определенную в распределенном семантическом пространстве, для вычисления перекрытия глосс-контекст. Учитывая текст $w_1, w_2, ..., w_n$ из n слов, он устраняет неоднозначность по одному, принимая во внимание сходство между глоссом, связанным с каждым смыслом целевого слова w_i, и контекстом. Выбирается значение, глосса которое имеет наибольшее сходство. Контекст может быть представлен подмножеством окружающих слов или всего текста, в котором это слово встречается. Более того, принимая во внимание идею адаптации Банерджи [9], каждый глосс дополняется соответствующими значениями. *BabelNet* используется как реестр чувств, очень большая многоязычная семантическая сеть, построенная на основе WordNet и Wikipedia.

В BabelNet лингвистические знания обогащены энциклопедическими концепциями, взятыми из Википедии. Синсеты WordNet и концепции (страницы) Википедии связаны автоматически.

Graph-based WSD. В дополнение к этим подходам, важная ветвь систем, основанных на знаниях, нашла свои методы на структурных свойствах семантических графов из лексических ресурсов [10–12]. Как правило, эти системы WSD на основе графов сначала создают графическое представление входного текста, а затем используют различные алгоритмы на основе графов для данного представления (например, PageRank) и выполнения WSD.

UKB – это система WSD на основе графов, которая использует случайные блуждания по семантической сети (в данном случае граф WordNet). UKB применяет алгоритм Personalized Page Rank [13], инициализированный с использованием контекста целевого слова. В отличие от большинства систем WSD, UKB не отступает от эвристики первого смысла WordNet и является самодостаточной (то есть не использует никаких внешних ресурсов / корпусов).

Babelfy – это основанный на графах подход к устранению неоднозначности, который использует случайные блуждания для определения связей между синсетами [14]. В частности, *Babelfy* использует случайные обходы с перезапуском по *BabelNet* [15], большой семантической сети, объединяющей WordNet среди других ресурсов, таких как Wikipedia или Wiktionary. Алгоритм метода *Babelfy* основан на эвристике сверхплотного подграфа для выбора семантических интерпретаций с высокой когерентностью вводимого текста. Лучшая конфигурация *Babelfy* учитывает не только целевое предложение, в котором встречается целевое слово, но и весь документ.

2. Постановка задачи и разработка метода устранения неоднозначности слов. Пусть имеется контекст $\boldsymbol{C}$, фрагментированный из текста, а $\boldsymbol{w}$ – это слово в этом контексте. Также, пусть $\boldsymbol{KB}$ – это лексическая база знаний (например, WordNet), которая предоставляет для каждого слова список синсетов, аннотаций, объясняющих множественные значения слова, причём каждый синсет имеет определение (глосс) и примеры того, как он используется в языке.

Задача состоит в том, чтобы определить правильное значение слова $\boldsymbol{w}$ в данном контексте $\boldsymbol{C}$, используя информацию лексической базы данных $\boldsymbol{KB}$. Задача решается путём измерения сходства между текстом контекста и текстами синсетов целевого слова, а затем выбирается синсет, который достигает наивысшей степени сходства с контекстом. Алгоритм предлагаемого метода описывается следующими шагами.

Шаг1. Для контекста $\boldsymbol{C}$ и целевого слова *w* (неоднозначное слово) извлекается набор синсетов $\boldsymbol{S} = (\boldsymbol{S_0}, \boldsymbol{S_1} \dots, \boldsymbol{S_N})$ с помощью WordNet. Затем для каждого синсета извлекается определение $\mathrm{O} = (\mathrm{o_1}, \mathrm{o_2}, \dots, \mathrm{o_N})$ и набор примеров $\mathrm{П} = (\mathrm{п_1}, \mathrm{п_2}, \dots, \mathrm{п_N})$. Предлагается объединить определение с примерами как единый текст в так называемой аннотации синсета:

$$\boldsymbol{T}_{\text{аннотации}} = \boldsymbol{concatenate}(\mathrm{П}, \mathrm{O}) = (\boldsymbol{t_1}, \boldsymbol{t_2}, \dots, \boldsymbol{t_N}).$$

Шаг 2. Тексты контекста и аннотация для каждого синсета представляются в виде числовых векторов. Существует много известных методов векторизации текста, однако в данной работе авторы предлагают использовать один из следующих методов, поскольку они позволили достичь наилучших результатов в области схожести текстов [16, 17].

1. Среднее значение вложений слов (на уровне слов), когда каждый текст представлен средним вектором встраивания всех слов, которые появляются в тексте. Можно использовать любой из методов генерации вложений таких как, *Word2vec* и *GloVe* [18, 19].

2. Вложения *Sentence-BERT (SBERT)* (на уровне предложения) – это модификация предварительно обученной сети *BERT*, в которой используются сиамские и триплетные сетевые структуры для получения семантически значимых вложений предложений, которые можно сравнивать с использованием косинусного сходства:

$$\boldsymbol{t_i^B} = \text{Вложения } (\boldsymbol{t_i}), \boldsymbol{t_i^G} = \text{Вложения } (\boldsymbol{t_i}), \quad (1)$$

где, $\boldsymbol{t_i^B}$ – это вектор вложения аннотации $\boldsymbol{i}$-го синсета, а $\boldsymbol{B,G}$ указывают тип используемого вложения (*B – BERT, G – GloVe*)

Шаг 3. Измерение сходства между контекстом $\boldsymbol{K}$ и аннотацией $\boldsymbol{t_i^B}$ с помощью меры сходства косинуса, задаваемой следующей формулой:

$$\boldsymbol{sim}\left(\boldsymbol{t_j}, \boldsymbol{K}\right) = \boldsymbol{cos}(\boldsymbol{\theta}) = \frac{\boldsymbol{X.Y}}{\|\boldsymbol{X}\|.\|\boldsymbol{Y}\|} = \frac{\sum_{i=1}^{n} \boldsymbol{X_i Y_i}}{\sqrt{\sum_{i=1}^{n} \boldsymbol{X_i^2}} \sqrt{\sum_{i=1}^{n} \boldsymbol{Y_i^2}}}, \quad (2)$$

где $\boldsymbol{X_i}$ и $\boldsymbol{Y_i}$ – компоненты вектора вложения контекста и аннотация $\boldsymbol{K}$ и $\boldsymbol{t_i^B}$, соответственно.

Степень сходства можно рассчитать с использованием одного или обоих методов векторизации, а затем принять среднее значение в качестве окончательного значения сходства:

$$\boldsymbol{sim} = \boldsymbol{f}(\boldsymbol{Sim_{GloVe}}, \boldsymbol{Sim_{Bert}}). \quad (3)$$

Синсеты ранжируются по степени сходства, затем выбирается синсет с наивысшим показателем схожести. Логику работы предложенного метода устранения неоднозначности слов показывает представленный ниже алгоритм.

Алгоритм устранения неоднозначности слов

Ввод Контекст **K**, многозначное целевое слово **w**, лексическая база данных WordNet

Вывод $\boldsymbol{S_{best}}$ Конкретное значение слова **w** в данном контексте

$S = (S_0, S_1 \ldots, S_N)$ // Список возможных синсетов (чувств) слова извлекается из WordNet

$\text{О} = (\text{о}_\mathbf{1}, \text{о}_\mathbf{2}, \ldots, \text{о}_\boldsymbol{N})$ //извлекается определение (глосс) каждого синсета

$\text{П} = (\text{п}_\mathbf{1}, \text{п}_\mathbf{2}, \ldots, \text{п}_\boldsymbol{N})$ //извлекается набор примеров (examples) каждого синсета

$T_{\text{пр}} = \text{П} + \text{О} = (\boldsymbol{t_1}, \boldsymbol{t_2}, \ldots, \boldsymbol{t_N})$// Текст определения и примеры объединены

$\text{К}^B = Bert(\text{К})$ // Представление текста контекста в виде вектора с помощью вложений Берта

$\text{К}^g = avgGloVe(\text{К})$ // Представление текста контекста в виде вектора с помощью вложений GloVe

$\boldsymbol{N} = \boldsymbol{len}(S)$ //количество чувств

$Similarity\_scores = [\,]$

$\boldsymbol{for}\ i = 0\ to\ \boldsymbol{N}\ \boldsymbol{do}$

  $t_i^g = avgGloVe(\boldsymbol{t_i})$ // Представление текста синсета $\boldsymbol{S[i]}$ в виде вектора с помощью вложений avgGloVe

  $t_i^B = Bert(\boldsymbol{t_i})$ // Представление текста синсета $\boldsymbol{S[i]}$ в виде вектора с помощью вложений Берта

  $\boldsymbol{Sim_{GloVe}} = \boldsymbol{cos}(\overrightarrow{t_i^g}, \overrightarrow{\text{К}^g})$

  $\boldsymbol{Sim_{Bert}} = \boldsymbol{cos}(\overrightarrow{t_i^B}, \overrightarrow{\text{К}^B})$

  $\boldsymbol{sim} = \boldsymbol{f}(\boldsymbol{Sim_{GloVe}}, \boldsymbol{Sim_{Bert}})$ // общее сходство

  $\boldsymbol{Similarity\_scores}\,[\boldsymbol{i}] \leftarrow \boldsymbol{sim}$

$\boldsymbol{end}$

$\boldsymbol{S_{best}} = \boldsymbol{max}(\boldsymbol{Sort}(\boldsymbol{S}, \boldsymbol{Similarity\_scores}))$ // Ранжирование синсетов по степени сходства с использованием одного или комбинации предыдущих мер и выбрать первый

3. Вычислительный эксперимент и анализ полученных результатов. Эффективность метода оценивалась путем сравнения его с несколькими другими методами на стандартном наборе данных, предназначенных для решения задачи устранения неоднозначности слов. Приведем описание использованных наборов данных и метрик оценки.

Для оценки предложенного метода использовались пять наборов данных, представленных в табл. 1.

♦ Senseval-2 (Se2) [20]. Этот набор данных изначально был аннотирован WordNet 1.7. После стандартизации он состоит из 2282 смысловых аннотаций, включая существительные, глаголы, наречия и прилагательные.

♦ Senseval-3 (Se3) [21]. Этот набор данных состоит из трех документов, относящихся к трем различным областям (редакционная, новостная и художественная), всего 1850 смысловых аннотаций. Версия WordNet этого набора данных была 1.7.1.

♦ SemEval-07 (Se7) [3]. Этот набор данных содержит 455 смысловых аннотаций только для существительных и глаголов. Изначально он был аннотирован с помощью инвентаризации WordNet 2.1.

♦ SemEval-13 (Se13) [22]. Этот набор данных включает тринадцать документов из разных доменов. Исходной инвентаризацией смысла был WordNet 3.0. Количество смысловых аннотаций – 1644, хотя учитываются только существительные.

♦ SemEval-15 (Se15) [14]. Это самый последний доступный на сегодняшний день набор данных WSD, аннотированный с помощью WordNet 3.0. Он состоит из 1022 смысловых аннотаций в четырех документах из трех разнородных областей: биомедицины; математики вычислений и социальных вопросов.

Согласно [2], уровень неоднозначности каждого набора данных вычисляется как общее количество возможных смыслов разделенное на количество смысловых аннотаций. Значение неоднозначности может указывать на то, насколько сложным может быть данный набор данных. В этом случае SemEval-07 отображает самый высокий уровень неоднозначности среди всех наборов оценочных данных.

Таблица 2

Наборы использованных данных

Набор данных	Кол-во документов	Кол-во предложения	Кол-во аннотаций	Неоднозначность
SensEval-2	3	242	2282	5.4
SensEval-3	3	352	1850	6.8
SemEval-7	3	135	455	8.5
SemEval-13	13	306	1644	4.9
SemEval-15	4	138	1022	5.5

В задачах WSD для измерения соответствия между предсказанным сходством и принятым стандартом, т.е. среднего сходства по оценке экспертов-людей, используется известная *F*-мера.

F-мера – это мера точности теста. Она представляет собой гармоническое среднее между точностью и полнотой, где точность – это количество истинно положительных результатов, деленное на количество всех положительных результатов, включая те, которые не были идентифицированы правильно, а полнота – это количество истинно положительных результатов, разделенное на количество всех выборок, которые должны были быть идентифицированы как положительные.

$$\boldsymbol{F} - \text{мера} = \frac{2 \times \text{точность} \times \text{Полнота}}{\text{точность} + \text{Полнота}}. \quad (4)$$

где

$$\text{Точность} = \frac{\text{сумма долей правильных прогнозов}}{\text{сумма долей правильных прогнозов+умма долей неверных прогнозов}}. \quad (5)$$

$$\text{Полнота} = \frac{\text{сумма долей правильных прогнозов}}{\text{количество экземпляров истинности}}. \quad (6)$$

В частности, доля неверных прогнозов вычисляется из ложноположительных прогнозов системы.

В проведенном вычислительном эксперименте сначала несколько реализаций предложенного метода были протестированы с разными аннотациями, только с определением, только с примерами, а затем путем объединения определения с примерами. Затем предложенный метод оценивался по результатам решения задачи устранения неоднозначности слов путем сравнения с семью проанализированными ранее методами, а именно: *WN 1st sense; Lesk (простой); Lesk (расширенный); Lesk (с вложением); UKB; UKB* (глосс) и *Babelfy*.

Кроме того, был проведен дополнительный эксперимент, идея которого заключается в объединении предложенного метода с одним из эффективных и проверенных канонических алгоритмов решения задач устранения неоднозначности, таких как *Babelfy* и *WordNet 1'st sen*se. Это необходимо для оценки того, дает ли предлагаемый метод дополнительное качество решений, которое указанные методы не достигают. Результаты эксперимента показаны в табл. 2.

Таблица 3

Результаты WSD по F-мере

		F-Мера на наборе данных				
Методы		**Se2**	**se3**	**se7**	**se13**	**se15**
WN 1st sense		66.8	**66.2**	55.2	62.95	67.8
Lesk (простой)		40.29	35.94	22.63	38.27	41.21
Lesk (расширенный)		50.6	44.5	32.0	53.6	51.0
Lesk (с вложением)		63.0	63.7	**56.7**	66.2	64.6
UKB$_{глосс}$		60.6	54.1	42.0	59.0	61.2
UKB		56	51.7	39.0	53.6	55.2
Babelfy		**67.0**	63.5	51.6	**66.4**	70.3
Предлагаемый метод						
Avg-GloVe	***Примеры + Глосс***	**56.96**	51.78	38.46	57.11	**64.38**
	Глосс	53.06	48.48	34.28	56.2	59.98
	Примеры	51.84	48.81	35.6	46.3	54.4
STS BERT	***Примеры + Глосс***	55.43	**52.27**	**42.63**	**58.94**	62.91
	Глосс	51,44	48,91	38,24	54,13	58,31
	Примеры	48.11	44.86	35.38	44,82	51,07
WN 1st + glove 0.14		**68.14***	64.43	51.43	**66.18**	**71.74***

При реализации предложенного метода, для каждого синсета извлекается определение и набор доступных примеров. Чтобы проанализировать влияние каждого из них в отдельности, эксперименты были проведены с использованием определения и примеров как независимых текстов, так и дополнения в виде их объединения в один текст.

Для векторизации текстов на уровне слов с помощью среднего значения вложений слов используется предварительно обученная модель *GloVe* [18], которая была обучена на наборе данных *Wikipedia 2014+360 Gigaword5*. *SBERT SentenceTransformers* и использовалась как метод векторизации предложений, а также является фреймворком *Python* для современных вложений предложений, текста и изображений [16].

Для метода *Леска* (расширенного и с вложением) использовалась та же реализация, предложенная в [2], а для реализация простого *Леска* использовалась функция, предоставляемая библиотекой *NLTK* в *Python*. Для *UKB* использовались обе конфигурации по умолчанию: с использованием графа *WordNet* (*UKB*) и полного графа, включая устраненные неоднозначности и глоссы в качестве соединений (глосс *UKB*).

Табл. 2 показывает результаты экспериментов по задаче устранения неоднозначности слов, причём, имеется несоответствие производительности систем по наборам данных, поскольку во всех случаях существует большой разрыв в производительности между лучшим и худшим набором данных. Уровень неоднозначности может указывать на то, насколько сложным может быть соответствующий набор данных. Фактически, системы WSD получают относительно низкие результаты в наборе *SemEval*-07, который является наиболее неоднозначным набором данных (см. табл. 1).

Для различных реализаций предложенного метода (жирные и подчеркнутые значения) наблюдается, что *BERT* достигает значений *F*-меры выше, чем *avg-GloVe* на наборах данных *semEval*-13, *semEval*-7 и *sensEval*-3, тогда как *avg-GloVe* дает лучшие результаты в *semEval*-15 и наборах данных *sensEval*-2, которые согласно табл. 1 менее неоднозначны. Это можно логически объяснить способностью *BERT* улавливать семантические отношения на уровне предложения лучше, чем *avg-GloVe*.

Также, отмечается, что производительность метода в случае объединения примеров с определениями намного лучше, чем если бы каждый из них использовался независимо. В целом производительность метода при использовании определений (будь то с *BERT* или *avg-GloVe*) лучше, чем при использовании примеров, и это связано с тем, что не во всех синтаксисах есть примеры, и поэтому разработанная система будет рассматривать их как ложноположительные прогнозы.

Что касается сравнения производительности разработанного метода с другими методами, следует отметить, что предложенный метод превзошел *Lesk* (простой) и *Lesk* (расширенный) по всем наборам данных, но он не смог превзойти расширенную версию *Lesk* (с вложением), что имеет смысл, поскольку *Lesk* (с вложением) опирается на несколько внешних источников для "определений", такие как WordNet и Wikipedia, а разработанный метод использует только информацию, предоставленную WordNet.

Также предложенный метод (с векторизации *avg-GloVe*) превзошел *UKB* по всем наборам данных, в то время как он превзошел расширенную версию $UKB_{глосс}$ только в наборе данных *semEval-15* .Предложенный метод не смог превзойти ни *Babelfy*, ни алгоритм WN 1st sense. Фактически, несмотря на простоту *WN 1st sense*, автоматические системы WSD не могут превзойти этот базовый уровень [11, 23].

На основе результатов, представленных в таблице 2, был выбран метод «*WN 1st sens*» для второго эксперимента по объединению с предлагаемым методом, учитывая его эффективность, простоту и легкость реализации. Что касается порога сравнения, то он был установлен экспериментально, где порог 0,14 дал наилучшие результаты по всем наборам данных. Результаты экспериментов в таблице 2, отмеченные звездочкой, показывают, что процесс слияния привел к значительному улучшению производительности обоих методов и превысил все базовые показатели на наборах данных *SensEval*-2 и *SemEval*-15, при этом достигнув значений, очень близких к максимальным значениям на остальных наборах данных.

Заключение. В данной работе представлена разработка модифицированного метода устранения неоднозначности слов WSD (англ. *Word Sense Disambiguation*), который по своему механизму имитирует хорошо известный вариант метода Леска «с вложением», характеризующийся применением концепции сходства между определением и контекстом.

Предлагаемый метод отличается простотой по сравнению с другими методами и использованием новейших доступных методов представления текста. Кроме того, метод использует преимущества примеров, предоставленных WordNet, чтобы получить достаточное объяснение каждого понятия в дополнение к определению, что позволяет уточнить векторы, используемые в аннотации, описывающей каждый синсет, это приводит к повышению точности определения правильного значения неоднозначного слова.

Поскольку определение значения слова связано с его контекстом и предметом документа, в котором оно встречается, вполне вероятно, что предложенный метод будет лучше работать в рамках других практических задач, в которых WSD является подзадачей.

Таким образом, для проверки эффективности предложенного метода проводилось его тестирование при решении других практических задач, таких как оценка сходства между текстами, классификация текстов, кластеризация, извлечение понятий, рекомендация, основанная на содержании, поиск информации, а также изучалось его влияние на выполнение этих задач по сравнению с другими методами.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Pal A.R., Saha D.J. a. p. a. Word sense disambiguation: A survey, 2015.
2. *Raganato A., Camacho-Collados J., Navigli R.* Word sense disambiguation: A unified evaluation framework and empirical comparison, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*: Vol. 1. Long Papers, 2017, pp. 99-110.
3. *Pradhan S., Loper E., Dligach D., Palmer M.* Semeval-2007 task-17: English lexical sample, srl and all words, *Proceedings of the fourth international workshop on semantic evaluations (SemEval-2007)*, 2007, pp. 87-92.
4. *Lesk M.* Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone, *Proceedings of the 5th annual international conference on Systems documentation*, 1986, pp. 24-26.
5. Kilgarriff A., Rosenzweig J. Framework and results for English SENSEVAL, *Computers the Humanities*, 2000, Vol. 34, No. 1, pp. 15-48.
6. *Vasilescu F., Langlais P., Lapalme G*. Evaluating Variants of the Lesk Approach for Disambiguating Words, *Lrec*., 2004.
7. *Banerjee S., Pedersen T*. An adapted Lesk algorithm for word sense disambiguation using WordNet, *International conference on intelligent text processing and computational linguistics*. Springer, 2002, pp. 136-145.
8. *Basile P., Caputo A., Semeraro G.* An enhanced lesk word sense disambiguation algorithm through a distributional semantic model, *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 1591-1600.
9. Banerjee S., Pedersen T. Extended gloss overlaps as a measure of semantic relatedness, *Ijcai*., Vol. 3. Citeseer, 2003, pp. 805-810.
10. *Agirre E., Soroa A.* Personalizing PageRank for Word Sense Disambiguation, EACL, 2009.
11. *Agirre E., Lopez de Lacalle O., Soroa A.* Random walks for knowledge-based word sense disambiguation, *Computational Linguistics*, 2014, Vol. 40, No. 1, pp. 57-84.
12. *Tripodi R., Pelillo M*. A game-theoretic approach to word sense disambiguation, *Computational Linguistics*, 2017, Vol. 43, No. 1, pp. 31-70.
13. *Haveliwala T.H.* Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search, *IEEE transactions on knowledge data engineering*, 2003, Vol. 15, No. 4, pp. 784-796.

14. *Moro A., Navigli R.* Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking, *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, 2015, pp. 288-297.
15. *Navigli R., Ponzetto S.P.* BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network, *Artificial intelligence*, 2012, Vol. 193, pp. 217-250.
16. *Reimers N., Gurevych I.* Sentence-bert: Sentence embeddings using siamese bert-networks, *arXiv preprint arXiv: 10084*, 2019.
17. *Iacobacci I., Pilehvar M.T., Navigli R.* Embeddings for word sense disambiguation: An evaluation study, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Vol. 1: Long Papers), 2016, pp. 897-907.
18. *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 (EMNLP)*, 2014, pp. 1532-1543.
19. *Kenter T., De Rijke M.* Short text similarity with word embeddings, *Proceedings of the 24th ACM international on conference on information and knowledge management*, 2015, pp. 1411-1420.
20. *Edmonds P., Cotton S.* Senseval-2: overview, *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, 2001, pp. 1-5.
21. *Snyder B., Palmer M.* The English all-words task, *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, 2004, pp. 41-43.
22. *Navigli R., Jurgens D., Vannella D.* Semeval-2013 task 12: Multilingual word sense disambiguation, *Second Joint Conference on Lexical and Computational Semantics (* SEM). Vol. 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, 2013, pp. 222-231.
23. *Navigli R.J.A. c. s.* Word sense disambiguation: A survey, 2009, Vol. 41, No. 2, pp. 1-69.

Статью рекомендовал к опубликованию к.т.н., доцент С.Г. Буланов.

Кравченко Юрий Алексеевич – Южный федеральный университет; e-mail: yakravchenko@sfedu.ru; г. Таганрог, Россия; тел.: +79289080151; кафедра систем автоматизированного проектирования; доцент.

Мансур Али Махмуд – e-mail: mansur@sfedu.com; тел.: 88634371651; кафедра систем автоматизированного проектирования; аспирант.

Мохаммад Жуман Хуссайн – e-mail: zmohammad@sfedu.ru; кафедра систем автоматизированного проектирования; аспирант.

Kravchenko Yury Alekseevich – Southern Federal University; e-mail: yakravchenko@sfedu.ru; Taganrog, Russia; phone: +79289080151; the department of computer aided design; associate professor.

Mansour Ali Mahmoud – e-mail: mansur@sfedu.com; phone: +78634371651; the department of computer aided design; graduate student.

Mohammad Juman Hussain – e-mail: zmohammad@sfedu.ru; the department of computer aided design; graduate student.