

А.В. ТОРОПОВА
**ОБНАРУЖЕНИЕ ИРОНИИ
НА ОСНОВЕ СЕМАНТИЧЕСКОЙ БЛИЗОСТИ**

Торопова А.В. Обнаружение иронии на основе семантической близости.

Аннотация. Цель данной работы – предложить алгоритм, использующий свойства семантической близости и позволяющий автоматически определить, употреблено ли слово с иронией или в прямом смысле. Сделан обзор предыдущих подходов к данной проблеме. Сформулировано определение понятию «ирония». Собраны две коллекции, одна из которых содержит высказывания со словами, употребленными с иронией, а вторая – с теми же словами, употребленными в буквальном смысле. Исследованы и проведены эксперименты над разными способами измерения семантической близости. Предложен алгоритм для обнаружения иронии, позволяющий автоматически определить, употреблено ли слово с иронией или в прямом смысле.

Ключевые слова: семантическая близость, обнаружение иронии, разрешение лексической неоднозначности, алгоритм Леска, NGD.

Toropova A.V. Irony detection based on semantic similarity.

Abstract. The goal of this work is to propose an algorithm to automatically determine whether the word used ironically or literally using semantic similarity. Previous approaches to this problem are reviewed. The term “irony” is defined. Two sets are collected, the first of them contains statements with words used ironically, the second contains statements with the same words used literally. Different methods of semantic similarity measuring are studied. A semantic similarity based algorithm to automatically determine whether the word used ironically or literally is proposed.

Keywords: semantic similarity, irony detection, word sense disambiguation, Lesk algorithm, NGD.

1. Введение. С развитием интернета, а вместе с ним таких явлений, как форумы, блоги, социальные сети и другие, растет и роль анализа тональности текста и высказываний (Sentiment Analysis, Opinion Mining) [28]. Это область исследований, занимающаяся анализом мнений, оценок, чувств и эмоций, которые передаются посредством письменного языка. Приложения анализа тональности текста используются во многих областях бизнеса и социальных наук [14, 26, 43], так как мнения — это один из ключевых аспектов человеческой деятельности: действия и поведение людей в значительной мере обусловлены мнениями и представлениями других. Предполагается, что при выборе книг, фильмов, продуктов, услуг и пр. люди обычно учитывают отзывы, оставленные другими. Для решения задач этой области очень важна правильная оценка высказываний. И для того, чтобы ее получить, надо учитывать многие факторы. Так как ирония меняет смысл зачастую на практически противоположный, качественная автоматическая обра-

ботка отзывов должна учитывать наличие иронии в тексте. Однако на сегодняшний день автоматические системы оценки отзывов не учитывают иронию, в связи с тем, что современные системы обработки естественного языка не способны качественно её выявлять.

Также автоматическое обнаружение иронии может быть применимо в задачах автоматического определения авторства: наличие иронии может оказаться важной компонентой авторского идиостиля [5].

Задача, которая рассматривается в настоящей работе, заключается в том, чтобы автоматически определить по контексту слова, употреблено ли оно иронически. *Иронией* называется «Риторическая фигура, в которой слова употребляются в смысле, обратном буквальному» [1, 2, 35].

Например:

1) Хлеб был таким свежим, что плесени на нем почти не было видно.

2) Мне нравится запах свежего хлеба.

В первом случае слово «свежий» употреблено иронично, а во втором — в буквальном смысле.

Так как иронию можно рассматривать как дополнительное значение слова, имеющее смысл, противоположный буквальному [1, 2, 5], мы предлагаем решать задачу автоматического обнаружения иронии как подзадачу области разрешения лексической неоднозначности (Word Sense Disambiguation) [6]. На основе алгоритма Леска [22], одного из методов этой области, а также гипотезы о том, что слово в данном контексте употреблено иронически тогда и только тогда, когда смысл слова, противоположного ему по значению, является семантически ближе к смыслу данного контекста, чем смысл данного слова, мы разработали алгоритм для автоматического обнаружения иронии на основе семантической близости.

2. Обзор существующих подходов. Несмотря на то, что такое явление как ирония довольно хорошо исследовано в лингвистике и психологии, для автоматического определения иронии в тексте было сделано не так много. Одной из первых работ по этому вопросу была работа Утсуми [40], однако, предложенная им модель была слишком абстрактной для того, чтобы ее можно было использовать на практике. Он определил три свойства иронии: аллюзия на ожидание автора высказывания, прагматичная нечестность и эмоциональное отношение. Был также предпринят ряд попыток [7, 8, 10, 15, 20, 21, 23, 24, 33, 37] по решению связанных с этим проблем таких, как обнаружение юмора, враждебных посланий или, более общее, использование не буквально-

го смысла. Бирке и Саркар [7] представили метод создания корпуса, снабженного информацией об использовании глаголов в буквальном или небуквальном смысле. Работа основывалась на двух наборах, содержащих примеры для обеих ситуаций: набор с буквальным употреблением содержал данные из Wall Street Journal (WSJ) [42], во втором – были собраны идиоматические и метафорические выражения из словарей. Вычислив семантическую близость между предложением, содержащим тестируемый глагол, и этими двумя наборами, определяли его как буквальное или небуквальное, в зависимости от того, расстояние до какого из наборов оказывалось меньше.

Михалчеа и Страппарава [23] представили подход для автоматического определения юмористических высказываний (one-liners). Авторы начали с построения корпуса, содержащего 16 000 юмористических высказываний и такого же числа заголовков Reuters [32], пословицы из онлайн-коллекции, предложения из Британского Национального Корпуса (British National Corpus) [9] со словами, аналогичными словам высказываний и предложения из коллекции Open Mind Common Sense [27]. После этого были проведены несколько экспериментов по классификации, где были использованы: 1) специфические стилистические особенности юмора, такие, как наличие аллитерации, антонимии и сленг; 2) особенности, основанные на содержимом (униграммы); 3) сочетание этих особенностей. Результаты показали [23], что для дерева классификации на основе специфических особенностей юмора возможно дифференцировать юмористические высказывания от заголовков Reuters [32] и BNC [9] предложений, но не от пословиц и OMCS [27] предложений. Классификация на основе содержимого контента с использованием метода опорных векторов и наивных байесовских классификаторов показала [23], что возможно отличить высказывания от всех других типов предложений (кроме BNC [9], которые были выбраны за схожесть содержимого с высказываниями). Сочетание этих особенностей не дало никакого улучшения. Анализ производительности, основанный на особенностях, юмора показал точность от 61% до 65%.

Крейц и др. [21] изучали значение нескольких лексических факторов при определении иронических/саркастических высказываний. Они собрали, используя Google Book Search [17], набор из ста предложений, содержащих фразу «said sarcastically», затем убрали наречие «sarcastically», чтобы устранить явные указания на ироническое содержание предложения. Они вручную проанализировали каждое предложение, чтобы проверить, содержит ли оно какую-нибудь из следу-

ющих особенностей: 1) наличие наречий и прилагательных; 2) наличие междометий; 3) наличие таких знаков препинания, как восклицательный или вопросительный знаки. Затем 101 участника эксперимента попросили оценить, не имея никакого контекста, эти предложения вместе с набором контрольных предложений в зависимости от того, насколько они кажутся ироничными. Для рейтинга использовали восьмибалльную шкалу (0 — совсем не кажутся; 7 — кажутся). Результаты показали, что у саркастических предложений рейтинг оказался выше, чем у контрольных (4,85 против 2.89). С помощью регрессионного анализа было установлено [21], что только наличие междометий может быть использовано при прогнозировании такого рейтинга.

Карвальо и др. [11] исследовали особенности иронии в пользовательском информационном материале (User-generated content (UGC)) на португальском языке. Они выделили восемь таких особенностей: 1) уменьшительные формы; 2) указательные местоимения; 3) междометия; 4) морфология глаголов (второе лицо, единственное число); 5) кросс-конструкции (прилагательные, относящиеся к существительным с помощью предлога «de»); 6) пунктуация (восклицательные и вопросительные знаки); 7) кавычки; 8) выражения, обозначающие смех (акронимы (lol итд.), звукоподражательные выражения, смайлики). Для исследования они собрали коллекцию из 8 211 новостей и соответствующих комментариев онлайн-читателей с сайта популярной португальской газеты. Наиболее продуктивным оказалось использование выражений, обозначающие смех и пунктуации.

Рейес и Росо [31] занимались задачей распознавания иронии в отзывах покупателей. Для экспериментов была собрана коллекция всего из 8 861 документа, состоящая из отзывов на некоторые продукты на Amazon.com [4] и Slashdot.com [34]. Авторы определили модель для обнаружения иронии из шести категорий: 1) n-граммы (n-grams); 2) n-граммы частей речи (POS n-grams); 3) Профилирование по словам, относящимся к юмору (Funny profiling, слова помеченные тегом «sexuality» в WordNet Domains, слова, относящиеся к множествам «relation», «relationship», «relative» в WordNet); 4) Профилирование по позитивным/негативным элементам (Positive/Negative Profiling); 5) Профилирование по элементам, указывающим на эмоции (Affective Profiling); 6) Профилирование по словам, относящимся к приятным эмоциям (Pleasantness Profiling). Были использованы три вида классификаторов, это наивный байесовский классификатор, метод опорных векторов и деревья решений. Наиболее подходящими категориями для дифференцирования оказались для данных с Amazon.com [4]: POS 3-

grams, профилирование по словам, относящимся к юмору и приятным эмоциям; а для данных с Slashdot.com [34]: профилирование по словам, относящимся к приятным эмоциям, 5-grams и профилирование по словам, относящимся к юмору.

Гонсалес-Ибанес и др. [15] рассмотрели проблему автоматического обнаружения сарказма в Twitter-сообщениях. Они использовали корпус, аннотированный самими пользователями Twitter [38], и обнаружили, что основными особенностями саркастических сообщений являются обращение к пользователю и смайлики.

Также можно упомянуть работы Вила и Хао [41], которые сфокусировались на задаче определения иронии в таких конструкциях английского языка, как «as ... as ...», и работу Бурфута и Болдуина [10], которые занимались автоматическим определением сатиры в статьях, оценивая особенности, связанные с заголовками, ненормативной лексикой и сленгом.

3. Постановка задачи. Стоит отметить, что предыдущие исследования, посвященные автоматическому обнаружению иронии [11, 31, 39–41] рассматривали иронию на уровне предложений, выясняя, является ли ироническим контекст, не уделяя вниманию тому, какие именно слова в данном контексте употреблены фигурально. Такая постановка задачи подходит для анализа отзывов, поскольку в этом случае важно оценить, изменилась ли тональность отзыва на противоположную. Однако в некоторых случаях необходим более тщательный семантический анализ, например, при решении задач, связанных с автоматическим пониманием текста.

В данной работе мы решаем следующую задачу: Определить по данному контексту слова, является ли оно употребленным иронически.

Под контекстом понимается фрагмент текста, включающий избранный для анализа единицу, необходимый и достаточный для определения значения этой единицы, являющегося непротиворечивым по отношению к общему смыслу данного текста [3].

Мы рассматриваем иронию на уровне слов, то есть выраженную одним словом, другие случаи в работе не рассматриваются.

4. Подход к решению задачи. Наш подход к решению задачи основан на следующей гипотезе: слово в данном контексте употреблено иронически тогда и только тогда, когда смысл слова, противоположного ему по значению, является семантически ближе к смыслу данного контекста, чем смысл данного слова.

Семантическая близость [19, 29] — это способ измерения, насколько сильны семантические связи между единицами текста, будь

то слова, предложения или документы. В нашем случае мы рассматриваем семантическую близость слов.

Некоторые пары слов встречаются в тексте вместе чаще, чем этого можно было ожидать. Это может быть показателем семантической близости между ними. Например, слова «кошка» и «собака» семантически ближе друг к другу, чем слова «кошка» и «автомобиль».

Мы рассматриваем нашу задачу как подзадачу *разрешения лексической неоднозначности* (word sense disambiguation, WSD) [6, 29, 33]. Разрешение лексической неоднозначности – это область исследований, занимающаяся решением задачи выбора значения (или смысла) слова в зависимости от контекста, в котором оно находится.

Подходы к решению задач области разрешения лексической неоднозначности обычно классифицируются по основному источнику знаний, использованному для дифференциации слов: это методы, основанные на словарях или на базах знаний (dictionary-based или knowledge-based methods), методы, которые практически полностью обходятся без каких-либо внешних данных, они называются неконтролируемыми (unsupervised methods), и контролируемые и полуконтролируемые методы (supervised and semi-supervised methods), использующие аннотированный корпус для тренировки или начальной загрузки.

В данной работе мы использовали методы разрешения лексической неоднозначности, основанные на метриках близости NGD [12, 13] и алгоритма Леска [22].

Normalized Google distance (NGD) [12, 13], нормированное Google-расстояние — это мера семантической близости слов, выводимая из количества результатов, выдаваемых поиском Google [16], для заданных наборов слов:

$$NGD(x, y) = \frac{G(x,y) - \min(G(x), G(y))}{\max(G(x), G(y))} = \frac{\max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log N - \min\{\log f(x), \log f(y)\}},$$

где $f(x)$ — это количество страниц, которые содержат x , а $f(x, y)$ — количество страниц, на которых одновременно встречаются x и y , возвращаемых поиском Google [16]. N — общее количество страниц, находящихся в индексе Google [16].

1) Область изменения NGD от 0 до ∞ :

а) Если $x = y$ или если $x \neq y$, но частота $f(x) = f(y) = f(x, y) > 0$, then $NGD(x, y) = 0$. Это значит, что для поиска Google смыслы x и y одинаковы.

b) Если частота $f(x) = 0$, тогда для каждого запроса y мы будем иметь $f(x, y) = 0$, и $NGD(x, y) = \infty/\infty$, мы считаем его равным 1 по определению.

2) NGD всегда неотрицательно и $NGD(x, x) = 0$ для каждого x . Для каждой пары x, y имеем свойство симметричности: $NGD(x, y) = NGD(y, x)$. Но несмотря на это NGD все-таки не является метрикой. Оно не удовлетворяет соотношению $NGD(x, y) > 0$ для каждого $x \neq y$. Выберем $x \neq y$ и $f(x) = f(y)$, тогда $f(x) = f(y) = f(x, y)$ и $NGD(x, y) = 0$. Также оно не удовлетворяет неравенству треугольника $NGD(x, y) \leq NGD(x, z) + NGD(z, y)$ для любых x, y, z . Например выберем $z = x \cup y$, $x \cap y = \emptyset$, $x = x \cap z$, $y = y \cap z$ и $|x| = |y| = \sqrt{N}$, тогда $f(x) = f(y) = f(x, z) = f(y, z) = \sqrt{N}$, $f(z) = 2\sqrt{N}$ и $f(x, y) = 0$. Из этого получается, что $NGD(x, y) = \infty$ и $NGD(x, z) = NGD(z, y) = 2/\log N$, что не удовлетворяет неравенству треугольника при любом N .

3) NGD независимо от масштаба в следующем смысле: Предположим, что когда N , число индексируемых Google страниц растет, число страниц, содержащих данный поисковый запрос, становится равным определенной части N , то же самое происходит с количеством страниц, содержащих объединение поисковых запросов. Чтобы NGD дало нам объективное значение семантической близости между искомыми словами, нужно чтобы оно оставалось неизменным, в то время как число N неограниченно растет.

Например, для слова поиск Google [16] «horse» выдает 46 700 000 результатов, а для слова «rider» — 12 000 000. Поиск страниц, на которых встречаются оба эти слова выдает 2 630 000 результатов, а количество индексируемых Google страниц равно 8 058 044 651. Используя эти данные и подставив их в формулу (1), выводим, что Google расстояние между словами «horse» и «rider» равно $NGD(\text{horse}, \text{rider}) \approx 0.443$.

Смысл слова во многих задачах области разрешения лексической неоднозначности предлагается рассматривать как совокупность его употреблений (sense is a group of contextually similar occurrences of a word) [33]. Мы представляем смыслы слова как наборы документов, связанных со словом и окружающим это слово текстом. Такие представления можно получить с помощью поисковых систем, где в качестве поискового запроса для слова используется оно само.

Алгоритм Леска [22] — это алгоритм разрешения многозначности, основанный на знаниях. Оригинальный Алгоритм Леска [22] снимает неоднозначность целевого слова, сравнивая его определения с определениями окружающих слов. Целевому слову присваивается то

определение, которое имеет наибольшее словарное перекрытие со словами контекста. Мы воспользуемся версией этого алгоритма [20], в которой неоднозначность целевого слова снимается после сопоставления его определений со словами контекста.

Этот алгоритм основан на двух гипотезах. Первая гипотеза заключается в том, что со слов, встречающихся в предложении, можно снять неоднозначность, присвоив им смысл, который наиболее близок к соседним словам. Это следует из интуитивного понятия того, что слова, встречающиеся вместе в одном предложении, обязательно каким-то образом связаны между собой, так как, согласуясь друг с другом, они передают некоторый смысл. Вторая гипотеза состоит в том, что связанные между собой смыслы можно идентифицировать, находя в их определениях слова, которые встречаются в толкованиях обоих слов. Эту гипотезу также можно обосновать с помощью интуитивного понятия, а именно того, что слова, которые связаны по смыслу, могут быть определены с одними и теми же терминами, а также могут ссылаться друг на друга в своих определениях.

В качестве примера можно рассмотреть два значения слова «калейдоскоп»:

- 1) оптический прибор — трубка с зеркальными пластинками и цветными стеклышками, при поворачивании складывающимися в разнообразные узоры;
- 2) быстрая смена разнообразных явлений.

И текст «Красивые цветные узоры калейдоскопа завораживают и взрослых, и детей». С первым определением у этого текста два общих слова, а со вторым — ни одного. Следовательно, алгоритм Леска берет первое значение.

5. Данные. Для проведения экспериментов в рамках данной работы были собраны две коллекции на английском языке.

Мы решили работать с английским языком, так как для английского языка создано большое количество различных библиотек и других ресурсов, в то время как для русского их не так много.

Одна из коллекций — «положительная», в ней собраны 60 высказываний, в каждом из которых одно из слов употреблено в противоположном обычному употреблению смысле, то есть можно назвать это слово словом-иронией. Для этой коллекции небольшая часть данных была взята из коллекции, собранной в [31], но так как подходящих нам высказываний в ней оказалось не так много, основная часть была найдена в Twitter [38] социальной сети, представляющей собой систе-

му микроблогов, позволяющую пользователям создавать короткие текстовые заметки.

Вторая коллекция — «отрицательная». В ней содержится 60 высказываний, в каждом из которых то слово, которое в первой коллекции было употреблено иронически, здесь употребляется в прямом смысле.

6. Эксперименты. Так как высказывания в коллекциях достаточно короткие, в качестве контекста определяемого слова мы берем высказывание целиком, исключая само определяемое слово.

Для подтверждения гипотезы о том, что слова, которые употреблены в ироническом смысле, имеют меньшую семантическую близость к контексту, чем слова, которые являются их антонимами, мы решили воспользоваться NGD [12, 13].

Для начала мы вычислили расстояния между словами-ирониями и контекстами, антонимами к словам-ирониям и теми же контекстами, исключая из поиска само высказывание, в «положительной» коллекции. Антонимы к словам-ирониям мы находили с помощью [25, 36]. Так как количество вводимых в поисковый запрос слов в Google [16] ограничено, некоторые контексты были сокращены до приемлемого размера.

Были получены следующие результаты: из 60 высказываний только у 24 расстояние между словом-иронией и контекстом оказалось больше, чем между антонимом и контекстом.

В «отрицательной» же коллекции только в 25 случаях расстояние между словом-иронией (здесь использованном уже в прямом смысле) и контекстом оказалось меньше, чем между антонимом и контекстом.

Таблица 1. Результаты эксперимента, основанного на NGD

	Верно определены	Неверно определены
Положительные случаи (слово употреблено иронически)	24	36
Отрицательные случаи (слово употреблено в прямом смысле)	25	35

Точность (precision) и Полнота (recall), определяемые по следующим формулам:

$$Pr = \frac{|{\text{true positive}}|}{|{\text{true positive}}|+|{\text{false positive}}|} \text{ и } Re = \frac{|{\text{true positive}}|}{|{\text{true positive}}|+|{\text{false negative}}|}$$

соответственно равны $Pr = 41,38\%$ и $Re = 40,68\%$.

F-мера F_1 (F-measure), вычисляемая как гармоническое среднее значений Pr и Re :

$$F_1 = \frac{2 \times Pr \times Re}{Pr + Re}, \text{ равна } F_1 = 41,03\%.$$

Таким образом, этим экспериментом наша гипотеза не подтвердилась.

Скорее всего, это произошло из-за того, что во многих случаях антонимы к словам-ирониям встречаются достаточно редко, так как для того, чтобы выразить нужный смысл чаще используются не антонимы, а отрицание. Например, возьмем глагол *like* (любить, нравиться), очевидно, что его антоним *dislike* (испытывать неприязнь, не любить) будет употребляться гораздо реже, чем отрицательная форма того же слова *like*.

Но для того, чтобы учитывать отрицания поисковая система Google [16] не подходит (отрицания, например *no* и *not*, входят в список стоп-слов Google).

При проведении эксперимента с использованием алгоритма Леска для представления смысла слов мы воспользовались поисковой системой Британского Национального Корпуса (BNC) [9].

Британский национальный корпус [9] — это 100 миллионов словарных коллекций образцов письменной и устной речи, собранных из широкого спектра источников таких, как выдержки из региональных и национальных газет, специализированных изданий, журналов для людей разных возрастов и интересов, научных книг и популярной фантастики, писем, школьных и университетских эссе, образцов живой речи, он создан для представления среза британского английского, как устного, так и письменного, языка конца XX века. Кроме того он оснащен удобной системой поиска, позволяющей найти предложения, в которых обязательно присутствуют отрицания.

Для начала были собраны представления для слов из списка слов-ироний и их отрицаний. Эти представления были составлены из около тридцати примеров употребления слова в Британском Национальном Корпусе каждое.

После этого представления и исходные контексты прошли процедуру стемминга [30].

Далее были посчитаны словарные перекрытия исходной фразы, из которой исключили слово-иронию, и представлений слова-иронии и его отрицания, то есть было подсчитано количество слов, встречающихся и в контексте и в представлении слова. Мы получили следующие результаты:

Для «положительной» коллекции в 37 случаях из 60 словарные перекрытия отрицаний и контекста оказались больше, чем словарные перекрытия слов-ироний и контекста.

Для «отрицательной» коллекции в 45 случаях словарные перекрытия слов-ироний, употребленных в обычном смысле, и контекста оказались больше, чем словарные перекрытия отрицаний и контекста.

Таблица 2. Результаты эксперимента, основанного на алгоритме Леска

	Верно определены	Неверно определены
Положительные случаи (слово употреблено иронически)	37	23
Отрицательные случаи (слово употреблено в прямом смысле)	45	15

Точность и Полнота для этого эксперимента оказались равными соответственно $P_r = 71,15\%$ и $R_e = 61,67\%$. F-мера равна $F_1 = 66,07\%$.

7. Формулировка алгоритма. Таким образом, алгоритм определения, употреблено ли целевое слово с иронией, может быть представлен в следующем виде.

Дано: целевое слово в контексте

Требуется: определить в буквальном смысле употребляется слово или с иронией.

- 1) собрать представление для целевого слова
- 2) собрать представление для отрицания целевого слова
- 3) для полученных текстов провести процедуру стемминга
- 4) вычислить по алгоритму Леска меры близости
 - a) для контекста и целевого слова
 - b) для контекста и отрицания целевого слова
- 5) в зависимости от результата определить целевое слово
 - a) как употребленное в буквальном смысле, если мера близости для контекста и целевого слова больше, чем для контекста и отрицания целевого слова
 - b) как употребленное с иронией, в обратном случае
 - c) если меры близости для контекста и целевого слова и для контекста и отрицания целевого слова равны нулю или равны между собой, то признать невозможность классификации слова с помощью описанного метода.

8. Заключение. В данной работе был предложен метод обнаружения слов, употребленных иронически. В основе метода лежит следующая идея: слова со значением противоположным буквальному значению слова, употребленного с иронией, будут семантически ближе к контексту, чем само это слово. Были исследованы разные способы измерения семантической близости. Также были проведены два вида экспериментов на основе методов области разрешения лексической неоднозначности, NGD и алгоритма Леска. Удовлетворительные результаты были получены при использовании алгоритма Леска. Предложен алгоритм для автоматического определения иронии в тексте.

Литература

1. *Винокур Г.О., Ларин Б.А., Ожегов С.И., Томашевский Б.В., Ушаков Д.Н.* Толковый словарь русского языка: В 4 т. М.
2. *Дынский В.* Ирония // Литературная энциклопедия: В 11 т. Т. 4. М.: Изд-во Ком. Акад., 1930. С. 571–580.
3. *Торсуева И.Г.* // Лингвистический энциклопедический словарь. М.: СЭ, 1990. С. 238–239.
4. Amazon.com – Электронный ресурс: <http://www.amazon.com/> (доступ 27.01.2013).
5. *Argamon S., Koppel M., Penebaker J. W., Schler J.* Automatically profiling the author of an anonymous text. Commun. ACM, 52. 2009. P. 119–123.
6. *Agirre E., Edmonds Ph.* Word Sense Disambiguation. Algorithms and Applications // Text, Speech and Language Technology. V. 33, 2006.
7. *Birke J., Sarkar An.* A Clustering Approach for the Nearly Unsupervised Recognition of Nonlinear Language // Proceedings of EAACL-06. 2006.
8. *Bogdanova D.* A Framework for Figurative Language Detection Based on Sense Differentiation // Association for Computational Linguistics. V. 7. 2010. P. 67–72.
9. British National Corpus – Электронный ресурс: <http://corpus.byu.edu/bnc/> (доступ 27.01.2013).
10. *Burfoot C., Baldwin T.* Automatic satire detection: Are you having a laugh? // ACL-IJCNLP'09: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. 2009. P. 161–164.
11. *Carvalho P., Sarmiento L., Silva M., de Oliveira E.* Clues for detecting irony in user-generated contents: oh...!! it's "so easy" ;-) // TSA '09: Proceeding of the 1st international CIKM workshop on Topicsentiment analysis for mass opinion. 2009. P. 53–56.
12. *Cilibrasi R., Vitanyi M.B.* Automatic Meaning Discovery Using Google // Kolmogorov Complexity and Applications Proceedings. 2006.
13. *Cilibrasi R., Vitanyi M.B.* Automatic The Google Similarity // IEEE Transactions on Knowledge and Data Engineering. Vol. 19, No 3. 2007. P. 370–383.
14. *Dave K., Lawrence S., Pennock D.M.* Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews // Proc. 12th Int'l World Wide Web Conf. 2003.
15. *González-Ibáñez R., Meresan S., Wacholder N.* Identifying Sarcasm in Twitter: A Closer Look // Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: shortpapers. 2011. P. 581–586.
16. Google – Электронный ресурс: <http://www.google.ru/> (доступ 27.01.2013).
17. Google Books – Электронный ресурс: <http://books.google.com/> (доступ 27.01.2013).

18. *Hao Y., Veale T.* An Ironic Fist in a Velvet Glove: Creative Mis-Representation in the Construction of Ironic Similes // *Journal of Minds and Machines*, V. 20 (4). 2010. P. 635–650.
19. *Hassan S., Mihalcea R.* Semantic Relatedness Using Salient Semantic Analysis // *Proceedings of the American Association for Artificial Intelligence (AAAI 2011)*. San Francisco. 2011.
20. *Kilgarriff A., Rosenzweig J.* English SENSEVAL: Report and Results // *Proceedings of the 2nd International Conference on Language Resources and Evaluation*. 2000. P. 1239–1244.
21. *Kreuz R., Caucci G.* Lexical Influences on the Perception of Sarcasm // *Proceedings of the Workshop on Computational Approaches to Figurative Language*. 2007. P. 1–4.
22. *Lesk M.* Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone // *SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation*. 1986. P. 24–26.
23. *Mihalcea R., Strapparava C.* Learning to Laugh (Automatically): Computational Models for Humor Recognition // *Journal of Computational Intelligence*. V. 22(2). P. 126–142.
24. *Mihalcea R., Pulman St.* Characterizing Humour: An Exploration of Features in Humorous Texts // *Proceedings of the Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*. Springer. Mexico City. 2007.
25. *Mohammad S., Dorr B., Hirst Gr.* Computing Word-Pair Antonymy // *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-2008)*. 2008.
26. *Morinaga S., Yamanishi K., Tateishi K., Fukushima T.* Mining Product Reputations on the Web // *Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining*, 2002. P. 341–349.
27. Open Mind Common Sense – Электронный ресурс: <http://openmind.media.mit.edu/> (доступ 27.01.2013).
28. *Pang B., Lee L.* Opinion Mining and Sentiment Analysis // *Foundations and Trends in Information Retrieval*. Vol. 2, No 1-2. 2008. P. 1–135.
29. *Patwardhan S., Banerjee S., Pedersen T.* Using Measures of Semantic Relatedness for Word Sense Disambiguation // *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*. 2003. P. 241–257.
30. Porter Stemmer – Электронный ресурс: http://qaa.ath.cx/porter_js_demo.html (доступ 27.01.2013).
31. *Reyes A., Rosso P.* Mining Subjective Knowledge from Customer Reviews: A Specific Case of Irony Detection // *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (ACL-HLT 2011)*. 2011. P. 118–124.
32. Reuters – Электронный ресурс: <http://www.reuters.com/> (доступ 27.01.2013).
33. *Schutze H.* Automatic word sense discrimination // *Computational Linguistics*. V. 24(1). 1998. P. 97–123.
34. Slashdot – Электронный ресурс: <http://slashdot.org/> (доступ 27.01.2013).
35. *Sperber D., Wilson D.* On verbal irony // *Lingua*. V. 87. 1992. P. 53–76.
36. Thesaurus – Электронный ресурс: <http://thesaurus.com/> (доступ 27.01.2013).
37. *Tsur O., Davidov Dm., Rappoport A.* ICWSM — A Great Catchy Name: Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews // *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*. 2010. P. 162–169.
38. Twitter – Электронный ресурс: <http://twitter.com/> (доступ 27.01.2013).
39. *Utsumi A.* A unified theory of irony and its computational formalization // *Proceedings of the 16th conference on Computational Linguistics*. 1996. P. 962–967.

40. *Utsumi A.* Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony // *Journal of Pragmatics*. V. 32(12). P. 1777–1806.
41. *Veale T., Hao Y.* Detecting Ironic Intent in Creative Comparisons // *Proceedings of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence*. 2010. P. 765–770.
42. *The Wall Street Journal* – Электронный ресурс: <http://europe.wsj.com/home-page/> (доступ 27.01.2013).
43. *Yi J., Nasukawa T., Bunesco R., Niblack W.* Sentiment Analyzer: Extracting Sentiments about a Given Topic Using Natural Language Processing Techniques // *Proc. Third IEEE Int'l Conf. Data Mining*. 2003. P. 427–434.

Торопова Александра Витальевна — аспирант кафедры информатики математико-механического факультета С.-Петербургского государственного университета (СПбГУ), младший научный сотрудник лаборатории теоретических и междисциплинарных проблем информатики СПИИРАН. Область научных интересов: комплекс программных средств управления депозитарием электронных научных изданий. Число научных публикаций — 1. alexandra.toropova@gmail.com, СПИИРАН, 14-я линия В.О., д. 39, г. Санкт-Петербург, 199178, РФ; р.т. +7(812)328-3337, факс +7(812)328-4450. Научный руководитель — А.Л. Тулупьев.

Toropova Alexandra Vitalievna — PhD student of Computer Science Department, SPbGU, junior researcher, Laboratory of Theoretical and Interdisciplinary Computer Science, SPIIRAS. Research interests: electronic scientific publications depository magement software set. The number of publications — 1. alexandra.toropova@gmail.com, SPIIRAS, 14-th line V.O., 39, St. Petersburg, 199178, Russia; office phone +7(812)328-3337, fax +7(812)328-4450. Scientific advisor — A.L. Tulupjev.

Рекомендовано ТИМПИ СПИИРАН, зав. лаб. А.Л. Тулупьев, д.ф.-м.н., проф.
Работа поступила в редакцию 23.01.2013.

РЕФЕРАТ

Торопова А.В. **Обнаружение иронии на основе семантической близости.**

В некоторых областях исследований, например в тональном анализе или анализе отзывов, крайне важно уметь правильно давать оценку высказываниям, одним из важнейших факторов при этом является определение присутствия иронии. Ведь присутствие иронии во многих случаях может кардинально изменить смысл высказывания.

Цель данной работы — предложить алгоритм, позволяющий автоматически определить, употреблено ли слово в заданном контексте с иронией или в буквальном смысле, используя свойства семантической близости. Применения такому алгоритму можно найти и в других областях, например в задачах определения авторского идиостиля.

Сделан обзор предыдущих подходов к данной и смежным с этой проблематикам.

Сформулировано определение понятия «ирония».

В основе решения этой задачи использована следующая гипотеза: слово в данном контексте употреблено иронически тогда и только тогда, когда смысл слова, противоположного ему по значению, является семантически ближе к смыслу данного контекста, чем смысл данного слова. Семантическая близость измеряется тем, как часто встречаются вместе целевое слово и слова из контекста.

Так как иронию можно рассматривать как дополнительное значение слова, имеющее смысл, противоположный буквальному, задача, поставленная в этой работе, рассматривается как подзадача разрешения лексической неоднозначности. В связи с этим использованные в работе методы, NGD и алгоритм Леска, принадлежат этой области исследований.

Для проведения экспериментов были собраны две коллекции. Первая коллекция состояла из высказываний, в которых одно из слов было употреблено иронически, а во второй коллекции высказывания, в которых слова, которые в первой коллекции были употреблены с иронией, употреблялись в буквальном смысле.

Проведенные эксперименты показали, что только использование алгоритма Леска дает удовлетворительные для данной области исследований результаты, в то время как использование NGD не принесло никакой пользы при решении поставленной задачи.

На основе полученных результатов был сформулирован алгоритм, использующий свойства семантической близости и позволяющий автоматически определять, употреблено ли слово в заданном контексте с иронией или в прямом смысле.

SUMMARY

Toropova A.V. **Irony detection based on semantic similarity.**

In some research areas, such as sentiment analysis or opinion mining it is important to know how to evaluate statements. One of the most important factors in this case is to detect the presence of irony. Indeed, it can change in many cases the whole meaning of the utterance.

The goal of this work is to propose an algorithm to automatically determine whether the word in the given context used ironically or literally using semantic similarity. Applications of this algorithm can be found in other areas, such as authorship detection.

Previous approaches to this problem are reviewed.

The term “irony” is defined.

The solution of this problem is based on the following hypothesis: the opposite of the word used ironically is semantically closer to the context than the word itself. Semantic similarity in this case is describing by how often the target word and the words from the context are found together.

Since irony can be seen as an additional meaning contrary to the literal meaning, the posed problem is considered as a sub-problem of the word sense disambiguation. Therefore, used here NGD and Lesk algorithm methods belong to this research area.

For research two sets are collected. The first contains statements in which of them there is one word used ironically, the second contains statements with the same words but used literally.

The experiments showed that only the use of Lesk algorithm gives satisfactory results, while using NGD doesn't avail in solving the problem.

On the basis of these results a based on semantic similarity algorithm to determine whether the word used ironically or literally in the given context is formulated.