Министерство образования и науки Российской Федерации

Федеральное государственное автономное
образовательное учреждение высшего образования
«Московский физико-технический институт
(государственный университет)»

Факультет инноваций и высоких технологий
Кафедра компьютерной лингвистики

ОПРЕДЕЛЕНИЕ СЕМАНТЕМ ОТРИЦАНИЯ ДИСТРИБУЦИОННЫМИ МЕТОДАМИ

Выпускная квалификационная работа бакалавра
по направлению
01.03.02 «Прикладная математика и информатика»

Выполнила:
студентка 495а группы
Сопильняк О. А.

Научный руководитель:
Новицкий В. И.

Москва 2018

Содержание

Введение 3

1. Постановка задачи 3
   1.1. Общая постановка задачи 3
   1.2. Предметная область 4
   1.3. Формальная постановка задачи 5
   1.4. Область применения 6
   1.5. Обзор существующих работ 8

2. Корпус 13
   2.1. Обучающий корпус на русском языке 13
   2.2. Обучающий корпус на английском языке 14

3. Определение семантем отрицания с помощью нейронных сетей 16
   3.1. Используемые инструменты 16
   3.2. Предобработка корпуса 20
   3.3. Эксперименты 21

4. Результаты 26
   4.1. Методы оценки 26
   4.2. Результаты на тестовой выборке 27
   4.3. Анализ ответов и ошибок 29
   4.4. Объединение с информацией об антонимической полярности 32

Заключение 34

Список литературы 35

Введение

Отрицание — феномен, присутствующий во всех языках и необходимый для понимания смысловой нагрузки текста. Изучением отрицания достаточно плотно занимаются в рамках задач обработки естественного языка, используя различные подходы и эвристики. Однако большинство решений, связанных с автоматическим определением семантем отрицания, используют признаки, получение которых занимает большое количество времени и вычислительных мощностей, зависимы от инструментов, доступных для ограниченного набора языков, или игнорируют отрицаемые события и свойства, рассматривая лишь более широкое понятие — область действия отрицания. В данной работе представлен метод нахождения отрицаемых событий и свойств с помощью нейронных сетей, а также рассмотрено прикладное применение полученного метода в виде объединения информации об отрицаниях с антонимической полярностью слов.

1. Постановка задачи

1.1. Общая постановка задачи

Цель данной работы заключается в исследовании и реализации методов детектирования семантем отрицания в произвольных текстах с помощью нейронных сетей. Реализованный алгоритм должен для данного слова в предложении определять, в каком значении слово употребляется в предложении: положительном или отрицательном. В работе рассматриваются русский и английский языки.

1.2. Предметная область

Прежде чем сформулировать поставленную задачу в терминах предметной области, дадим некоторые определения основным семантемам отрицания, встречающимся в данной работе, а также приведем несколько примеров, необходимых для понимания задачи и цели работы.

Областью действия отрицания (negation scope) называется часть предложения, которая находится под отрицанием. [1]

Фокусом отрицания (focus of negation) называется часть области действия отрицания, которая отрицается наиболее явно. [1]

Меткой отрицания (negation cue) называется слово (например, *не*, *no*), несколько слов (например, *ни за что*, *no longer*) или часть слова (например, *не-*, *im-*), которая является индикатором отрицания. [2]

Отрицаемое событие или свойство (negated event or property) — это часть области действия отрицания, которая выражает отрицание некоторого события или свойства объекта. [2]

Рассмотрим пример для русского языка.

> **Пример:**
> Надеюсь, что реформа **не** *носит* ограничительный характер.

Слово *не* является меткой отрицания, *реформа* <...> *носит ограничительный характер* — область действия отрицания для метки *не*, а глагол *носит* — отрицаемое событие или свойство. Действительно, высказывание было бы абсолютно верно, если бы главным участником не была реформа, ее характер не был бы ограничительным и событие *носить* не имело бы место.

Теперь рассмотрим предложение с двумя метками отрицания на английском языке.

> **Пример:**
> After his habit he *said* **nothing**, and after mine I *asked* **no** questions.

Слова *no* и *nothing* являются метками отрицания, *after his habit he said* и *after mine I asked* <. . . > *questions* — области действия отрицания для меток *no* и *nothing* соответственно, а слова *said* и *asked* — отрицаемое событие или свойство.

В данной работе главным образом рассматривается семантема *отрицаемое событие или свойство*. В дальнейшем для простоты изложения будем понимать под отрицанием именно отрицаемое событие или свойство.

1.3. Формальная постановка задачи

Обучающий корпус: текстовые документы, разбитые на токены, в которых для токенов, находящихся под отрицанием, размечен класс отрицания (обозначим его классом *NEGATED*), а для всех остальных — противоположный ему (обозначим его классом *NOT_NEGATED*).

Вход: текстовый документ.

Выход: текстовый документ, в котором размечены классом отрицания те и только те токены, которые находятся под некоторым отрицанием.

Таблица 1 — Пример разметки

Word	Category
Реформа	NOT_NEGATED
не	NOT_NEGATED
носит	NEGATED
ограничительный	NOT_NEGATED
характер	NOT_NEGATED

1.4. Область применения

Несмотря на то, что отрицание является актуальным и сложным семантическим аспектом языка, текущие подходы заключаются либо в том, чтобы рассматривать его частично, либо в том, чтобы вовсе отбросить его. [2] В последние годы различные семантемы отрицания, в особенности область действия отрицания, интересуют научное сообщество с лингвистической точки зрения. Однако очень мало усилий было приложено к тому, чтобы разработать быстрый и простой в реализации способ детектировать отрицания в произвольных предложениях на различных языках. Большинство подходов либо используют уникальные эвристики для конкретного языка, либо требуют вычислительно сложного полного синтаксического разбора предложений. Все эти факторы несомненно затрудняют внедрение существующих алгоритмов в системы с высокой нагрузкой, предназначенные для конечного пользователя.

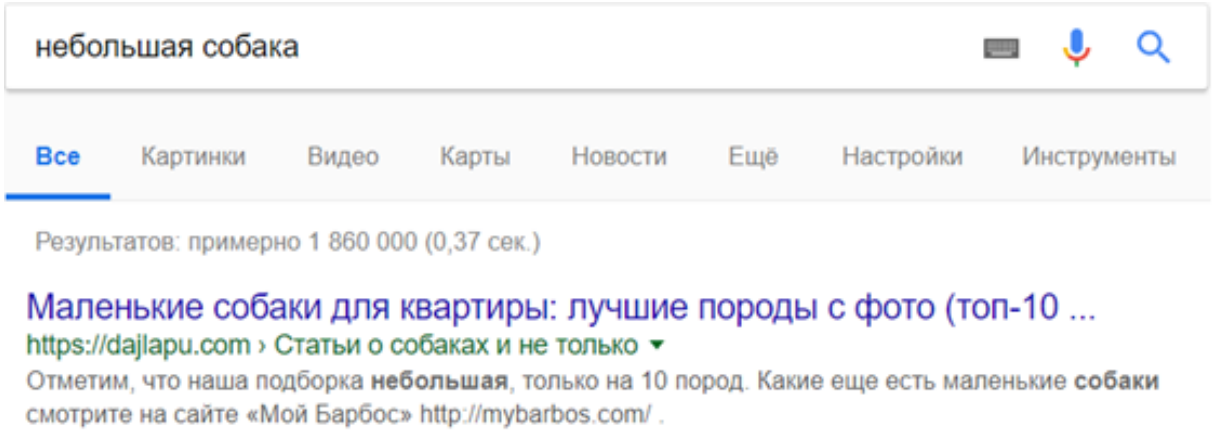

Рисунок 1 — Запрос с отрицанием на русском языке

Например, некоторые современные поисковые системы на момент написания данной работы не учитывают отрицание вовсе. Примеры запросов на русском и английском языках, содержащих отрицание, в одной из крупнейших на российском рынке поиско-

вой системе представлены на рисунках 1 и 2. В примере русского запроса (рисунок 1) слово «маленькая» хоть и присутствует в тексте, но не попадает под поисковую выдачу. А в случае запроса на английском языке (рисунок 2) выдача совершенно нерелевантная.

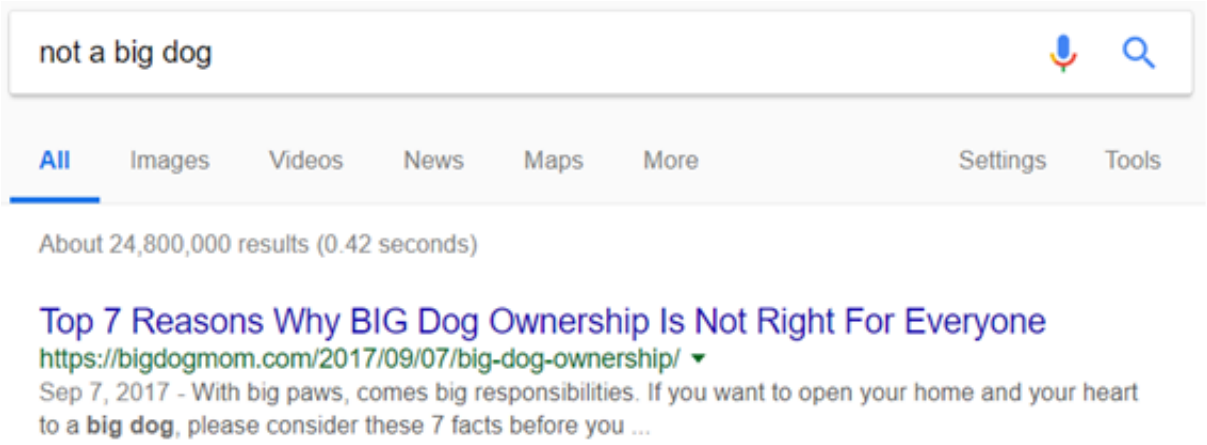

Рисунок 2 — Запрос с отрицанием на английском языке

Используя алгоритм, представленный в данной работе, можно объединить информацию об отрицаниях и антонимической полярности для различения противоположных по смыслу слов запроса и искомого текста. Это позволило бы поисковым системам на этапе индексации быстро определять слова, находящиеся под действием отрицания, и формировать поисковую выдачу, включая в том числе семантически близкие после получения информации об отрицании слова.

> **Пример:**
> Завтрашний ветер не будет сильным.
> Завтрашний ветер будет слабым.

В данном примере полученный алгоритм обозначил слово *сильным* как имеющий класс NEGATED и подобрал к нему антоним *слабый*.

1.5. Обзор существующих работ

В этом разделе рассматриваются наиболее влиятельные из существующих на настоящий момент работ по определению области действия отрицания, детектированию меток отрицания и отрицаемых событий или свойств. Для каждой из них кратко описаны особенности подхода к решению, используемые признаки, архитектура решения, корпус, на котором осуществлялось обучение и тестирование, а также полученные авторами результаты.

Нейронные сети для определения области действия отрицания

Основным опорным материалом для данной работы являлась статья [3] «Нейронные сети для определения области действия отрицания» (Neural Networks for Negation Scope Detection), в которой авторы рассматривали способы использования нейронных сетей для определения области действия отрицания. Их целью было показать, что даже простая нейронная сеть справляется с задачей поиска области действия отрицания не хуже существующих методов, основанных на правилах, а также обладают такими преимуществами, как простота реализации и независимость от конкретного языка и литературного жанра.

Задача. Для данного предложения с размеченными метками отрицания определить *область действия отрицания*, то есть часть предложения, на которую распространяется отрицание. Авторы вводят понятие *контекста отрицания*, под которым подразумевается набор слов, семантически связанных со своей меткой отрицания. Также их решение может корректно обрабатывать случаи, когда в одном предложении есть несколько отрицаний или область действия отрицания разрывна.

Признаки. Величина *I(n, c)* определена как пара числа n и вектора c длины n, в котором компонента равна 1, если соответствующее слово является частью метки отрицания и 0 иначе. Такой

признак позволит с легкостью определить, где лежит слово: внутри контекста отрицания или вне его.

Каждое слово w представлено в виде 50-мерного вектора x_i, а также в виде вектора c_i той же размерности. Последний вектор является индикатором того, является ли слово частью метки отрицания. Также авторы определяют матрицу эмбеддингов слова E_w размерности $w \times d$ и матрицу меток отрицания E_c размерности $2 \times d$, где $d = 50$ — размерность эмбеддинга.

Архитектура. В работе рассматривается две архитектуры нейронных сетей.

Первая модель — это сеть прямого распространения (Feedforward) с одним скрытым слоем. В данном случае из-за того, что такая сеть не имеет памяти и обрабатывает все вхождения по отдельности, авторы предлагают конкатенировать каждое слово с его соседями в рамках контекстного окна длины l.

Вторая модель является двунаправленной LSTM (Long Short Term Memory). Конкатенация в данном случае не производится, так как внутренняя структура сети уже работает с последовательностями.

Корпус. Для обучения нейронной сети и получения результатов авторы выбрали корпус, представленный на **SEM 2012 Shared Task* [2]. Тот же корпус рассматривается и в данной работе в качестве англоязычного, поэтому подробное его описание можно найти ниже.

Результат. Авторы успешно показали, что даже простая нейронная сеть является неплохой альтернативой существующим алгоритмам, а архитектура LSTM работает лучше, чем все другие.

Особенности. Метод, представленный авторами статьи, является наиболее выигрышным по простоте использования и потенциальной переносимости на другие языки. Однако авторами не рассмотрено детектирование меток отрицания, а также отрицаемых событий и свойств и, соответственно, не получены результа-

ты для этих задач, а для решения задачи определения области действия отрицания использованы векторы, включающие в себя информацию о метках отрицания.

Дискриминационное ранжирование на основе составляющих для разрешения отрицания

Метод, описанный в работе «Дискриминационное ранжирование на основе составляющих для разрешения отрицания» (UiO_1: Constituent-Based Discriminative Ranking for Negation Resolution) [4] опирается на некоторые специфические для английского языка эвристики для нахождения области действия отрицания, меток отрицания, а также отрицаемых событий и свойств. Авторы используют информацию о синтаксической структуре предложения, о частях речи, а также признаки по n-граммам слов и лемм для решения поставленной задачи. Они расширили существующий метод определения меток отрицания и области действия отрицания, основанный на методе опорных векторов (Support Vector Machine, SVM), до успешного детектирования разрывных областей действия отрицания и определения отрицаемых событий и свойств. Рассмотрим более подробно их способ определения отрицаемых событий и свойств.

Задача. Для данного предложения определить слова, которые являются *отрицаемым событием или свойством*.

Архитектура и признаки. Авторы решили использовать SVM-классификатор, который размечает фактическую зависимость от некоторого высказывания в предложении. В качестве положительного примера они взяли слова, которые принадлежат отрицаемому событию внутри области действия отрицания, а в качестве отрицательного — область действия отрицания без отрицаемого события. В качестве признаков авторы использовали мешок слов (bag-of-words), состоящий из меток отрицания, словоформ, лемм, частей речи, униграмм и биграмм. Признаки извлекались как из

всего предложения, так и из локального окна в шесть токенов вокруг метки отрицания. Далее авторы используют SVM для обучения ранжирующей функции, которая могла бы определять отрицаемые события среди токенов, лежащих в области действия отрицания.

Корпус. Для обучения и получения результатов также был использован корпус, представленный на **SEM 2012 Shared Task*.

Результат. Полученные результаты оказались лучшими по F_1 мере среди представленных на соревновании как в определении области действия отрицания, так и в детектировании отрицаемых событий и свойств.

Особенности. Рассмотренное решение опирается на специфические для английского языка эвристики, а также использует множество признаков, иногда достаточно нетривиально извлекаемых. Однако именно такой подход дал лучшие результаты на момент 2012 года.

Использование фразовых и контекстных признаков для определения области действия отрицания

В статье «Использование фразовых и контекстных признаков для определения области действия отрицания» (FBK: Exploiting Phrasal and Contextual Clues for Negation Scope Detection) [5] наряду с признаками, полученными из токена, использовалась информация о контексте слова. Авторы представили новый способ кодирования контекстуальной информации, который заключается, например, в признаках-подсказках.

Задача. Для данного предложения определить метки отрицания и слова, находящиеся в области действия отрицания или являющиеся отрицаемыми событиями или свойствами.

Признаки. Для определения меток отрицания использовался словарь наиболее часто встречающихся меток, а также такие признаки, как леммы, суффиксы и префиксы, входящие в составленный словарь. В качестве признаков использовалась информация о

леммах, частях речи, позиции как текущего токена, так и тех, что находятся в некотором контексте. Также учитывались разнообразные комбинации этих признаков.

Архитектура. Условные марковские поля (Conditional Random Fields, CRF).

Корпус. Для обучения и получения результатов также был использован корпус, представленный на **SEM 2012 Shared Task*.

Результат. Подход к определению меток отрицания дал лучший результат на соревновании, а результаты определения области действия отрицания и отрицаемых событий и свойств также оказались в числе лучших.

Особенности. Самое простое из рассмотренных решений, использует понятные и легко извлекаемые признаки, однако извлечение некоторых из признаков может потребовать больших вычислительных мощностей. Также это решение использует эвристики, присущие английскому языку, например, схему составления меток отрицания.

Простое разрешение области действия отрицания с помощью глубокого разбора: семантическое решение семантической задачи

В отличие от рассмотренных ранее подходов к решению задачи, авторы статьи «Простое разрешение области действия отрицания с помощью глубокого разбора: семантическое решение семантической задачи» (Simple Negation Scope Resolution through Deep Parsing: A Semantic Solution to a Semantic Problem) [6] используют глубокий семантический анализ для определения области действия отрицания и осуществляют переход к формальному логическому представлению отрицания в виде инструментов алгебры логики — кванторов и операторов. Кодируя семантемы отрицания именно таким способом, они добиваются лучшего на момент 2014 года результата на рассматриваемом корпусе.

Задача. Для данного предложения определить область действия отрицания.

Архитектура и признаки. Решение полностью основано на выведенных правилах и использует полный семантический разбор предложения, а также формальное логическое представление семантем отрицания.

Корпус. Тексты, взятые с соревнования **SEM 2012 Shared Task*.

Результат. Этот подход дал наилучшую F_1 меру из всех опубликованных на момент 2014 года работ на корпусе.

Особенности. Самое очевидное преимущество такой системы в том, что она не требует обучающего корпуса: весь метод состоит из набора правил. Однако такое решение использует инструменты, доступные лишь для ограниченного набора языков, а также требующие больших вычислительных мощностей (например, осуществляющие синтаксический и семантический разборы предложения). Также авторы используют для своего решения эталонную разметку меток отрицания.

2. Корпус

2.1. Обучающий корпус на русском языке

Обучающий корпус на русском языке представляет собой новости агенства Reuters за 1996–1997 гг., размеченные семантическими классами отрицания с помощью технологии ABBYY Compreno.

Каждая новость представлена в отдельном файле, в котором слова находятся на отдельных строках, а предложения разделены между собой пустой строкой. Каждому слову, находящемуся под действием отрицания, соответствует метка класса отрицания.

Файлы содержат следующие столбцы:

Offset: индекс первого символа от начала файла;

Text: символьное представление токена (чаще всего слово)

Polarity: семантическое значение полярности токена. Если поле пусто, то соответствующий токен не находится под влиянием отрицания. Если же в поле указано значение Polarity=Negative, это указывает на то, что данный токен находится в области действия какого-либо отрицания.

Предложения в каждом документе отделены друг от друга пустой строкой.

В целях получения общего представления о содержании обучающего корпуса была собрана статистика по корпусу, представленная в таблице 2.

Таблица 2 — Обучающий корпус на русском языке

Метрика	Количество
Количество слов	3 222 657
Всего предложений	206 856
Количество отрицаний / количество слов	0.8%
Предложений с отрицаниями	9%
Количество тегированных слов в окне +/-3 без «не» и «нет»	55
Отношение числа вхождений «не» и «нет», вокруг которых в окне +/-3 нет тега отрицания, к общему числу вхождений «не» и «нет»	2%

2.2. Обучающий корпус на английском языке

В качестве обучающего корпуса на английском языке был взят корпус, представленный на **SEM 2012 Shared Task — Resolving the Scope and Focus of Negation* в рамках конференции NAACL-HLT 2012, прошедшей в 2012 году в Канаде.

Корпус представляет собой рассказы Конана Дойла из цикла о Шерлоке Холмсе. Для обучающего корпуса были взяты главы 1-14

из рассказа «Собака Баскервилей», для валидационной выборки — рассказ «В Сиреневой Сторожке», а для тестовой — рассказы «Алое кольцо» и «Картонная коробка».

Каждая выборка представляет собой отдельный текстовый файл. Внутри каждое слово находится на отдельной строке, предложения между собой разделены пустой строкой. Для выделения токенов использовался токенизатор, являющийся частью LinGO English Resource Grammar. Он сохраняет всю пунктуацию и считает ее знаки отдельными токенами, а также принятые в английском языке сокращения отрицаний разделяет на несколько токенов (например, don't будет разделено на do и n't).

После токенизации каждое слово лемматизировалось с помощью GENIA tagger, а также были получены части речи и синтаксический разбор предложения. Для каждой метки отрицания размечены области действия отрицания, а также отрицаемое событие или свойство.

Затем обучающий корпус был преобразован в более подходящий для обучения нейронной сети формат: в первом столбце находится слово в исходной словоформе, затем флаг, равный NEGATED, если слово является отрицаемым событием или свойством, или NOT_NEGATED в противном случае. Если метка отрицания является приставкой или суффиксом к слову, на которое она влияет (например *im-* или *-less*), то они были разделены на два слова.

Для удобства и быстроты применения обученной нейронной сети, информация о дереве синтаксического разбора предложения была опущена, и ее использование не планировалось при обучении и получении результатов в данной работе. Затем весь корпус был разделен на обучающую (55%), валидационную (15%) и тестовую (30%) части.

В таблице 3 представлена статистика для корпуса на английском языке.

Таблица 3 — Обучающий корпус на английском языке

Метрика	Количество
Количество слов	57 392
Всего предложений	5 518
Количество отрицаний / количество слов	0.4%
Предложений с отрицаниями	15%
Количество тегированных слов в окне +/-3 без «no», «not» и «n’t»	427
Отношение числа вхождений «no», «not» и «n’t», вокруг которых в окне +/-3 нет тега отрицания, к общему числу вхождений «no», «not» и «n’t»	47%

3. Определение семантем отрицания с помощью нейронных сетей

3.1. Используемые инструменты

Описание токенизатора

В качестве токенизатора в работе использовался Natural Language Toolkit (NLTK) [7].

Библиотека NLTK реализована на языке Python и является open-source проектом. Она предоставляет несколько видов токенизаторов, принимающих на вход текст и возвращающих список токенов:

`tokenize.word_tokenize`: по словам. Под токенами в данном случае понимаются слова, пунктуация, числа и спецсимволы. Морфология слова не учитывается. Для задачи определения области действия отрицания такого токенизатора достаточно.

`tokenize.sent_tokenize`: по предложениям. Под токенами понимаются предложения, разделенные пунктуацией, обозначающей конец предложения. Однако пунктуация внутри и в конце предложения не различается: например, сокращение «т.е.» будет разбиваться на отдельные предложения по точкам.

`tokenize.wordpunct_tokenize`: по пробелам и знакам препинания. Разбиение происходит с помощью регулярных выражений по пробелам и знакам препинания.

Таким образом, объединив `tokenize.word_tokenize` и `tokenize.sent_tokenize`, можно получить разбиение на слова внутри предложений. В работе применяется именно такой вариант использования NLTK.

Рассмотрим подробнее выбранный токенизатор и выделим его свойства, знание которых необходимо для дальнейшей работы:

- Знаки препинания считаются отдельными токенами;
- Композитные слова (например, *плоскопараллельный*) обрабатываются без разделения, морфология слова, за исключением отдельных случаев в английском языке, указанных ниже, не учитывается при токенизации;
- Устойчивые словосочетания (например, *так как*) игнорируются и обрабатываются как раздельные слова;
- Пунктуаторы внутри слова (например, *какой-то*) учитываются: слова с пунктуаторами считаются за один токен;
- В английском языке сокращения (например, *can't*, *don't*, *I'm*) разделяются и считаются отдельными токенами (например, `['ca', 'n't'], ['do', 'n't'], ['I', ''m']`).

Описание эмбеддингов

Для получения эмбеддингов (векторных представлений слов) в работе рассматривались следующие предобученные модели.

SketchEngine: Russian (Web, 2011), English (Web, 2013) [8]

Корпус. Модели для русского и английского языков были обучены с использованием fastText [9] на корпусе TenTen, собранном с веб-страниц на момент 2011 года для русского языка и на момент 2013 года для английского языка и размещенном на веб-сервисе SketchEngine [8]. Корпус содержит в том числе информацию об их грамматических значениях и морфологически размечен с помощью TreeTagger [10].

Архитектура. Модель SkipGram.

Результат. Размерность итогового вектора: 100. В итоговом корпусе содержится 18 миллиардов значений для русского языка и 20 миллиардов для английского языка. В их число входят как стоп-слова, так и словоформы.

RusVectores: News [11]

Корпус. Модель обучалась на случайно отобранных русскоязычных новостях с сентября 2013 по ноябрь 2016 с приблизительно 1500 русскоязычных новостных сайтов. Размер корпуса, на котором проходило обучение, содержит около пяти миллиардов слов и 194 058 уникальных слов. Предобработка корпуса включала в себя токенизацию, разбиение на предложения, лемматизацию, а также были размечены части речи.

Архитектура. Continuous bag of words.

Результат. Размерность итогового вектора: 300. Итоговый корпус не содержит стоп-слова, а также словоформы. Часть речи у слова не указана.

Модель создана в феврале 2017 года.

RusVectores: Ruwikiruscorpora [11]

Корпус. Национальный корпус русского языка (НКРЯ) в полном объеме и дамп русской Википедии за ноябрь 2016 года. Корпус содержит 600 миллионов слов, из которых 392 339 уникальных.

Архитектура. Continuous bag of words.

Результат. Размерность итогового вектора: 300. Корпус не содержит стоп-слова и словоформы. Каждый токен содержит тег части речи, соответствующий формату Universal POS Tags. Пример токена: *печь_NOUN*.

Модель создана в январе 2017 года.

RusVectores: Web [11]

Корпус. Случайные русскоязычные веб-страницы на момент декабря 2014 года. Корпус содержит 900 миллионов слов, из которых 267 540 уникальных.

Архитектура. Continuous bag of words.

Результат. Размерность итогового вектора: 300. Корпус не содержит стоп-слова и словоформы. Каждый токен содержит тег части речи, соответствующий формату Universal POS Tags. Пример токена: *печь_NOUN*.

Модель также создана в январе 2017 года.

Итоговый выбор

Для задачи определения семантем отрицания в качестве основных были выбраны эмбеддинги от SketchEngine. Решение обусловлено тем, что только в них содержатся словоформы и информация о грамматических значениях слов, а для задачи определения области действия отрицания важны именно грамматические значения слов, а не только лексические. Также корпус достаточно велик, чтобы покрыть почти все слова, которые может содержать выбранный для обучения текстовый корпус.

3.2. Предобработка корпуса

Выбранные эмбеддинги хранятся в формате fastText [9]. В данной модели каждое слово представлено в виде суммы символьных n-грамм. Таким образом, в отличие от простых векторных представлений, учитывается морфология слов, а также предоставляется возможность получать векторное представление слов и словоформ, которых нет в исходном словаре.

Для загрузки эмбеддингов в формате fastText была использована библиотека gensim [12]. Файл с эмбеддингами загружается в оперативную память в бинарном формате, затем с помощью библиотеки gensim преобразуется в словарь, к которому можно обращаться по ключу, представляющему из себя словоформу. [1]

Формат разметки корпуса на английском языке оказался неудобным для поставленной задачи, поэтому его разметка была приведена к аналогичной корпусу на русском языке. Максимальный размер предложения выбран равным 30, а все предложения, содержащие меньше слов, выравнены нулями. Слова, содержащие посторонние символы, не являющиеся буквами алфавита, дефисом или апострофом в английском языке, были выброшены из корпуса для простоты.

В итоге была получена матрица X размерности $S \times W \times E$, где S — это количество предложений в корпусе, $W = 30$ — максимальное количество слов в предложении, а $E = 100$ — размерность эмбеддинга. Также была составлена матрица y ответов размерности, соответственно, $S \times W \times 2$, так как ответы представляют собой one-hot encoding вектор из двух классов, соответствующий каждому слово в предложении.

Далее полученные матрицы делятся на две части — обучающие X_{train}, y_{train} и тестовые X_{test} и y_{test} — так, чтобы размер обучающей выборки составлял 0.7 от размера X. В таком виде данные уже готовы для подачи в нейронную сеть.

[1] Исходный код всех экспериментов доступен на github.com/sopilnyak/negation-detection

3.3. Эксперименты

В данной работе использовались несколько различных архитектур нейронных сетей, а также было реализовано базовое решение для понимания сложности задачи и сравнения результатов на русскоязычном корпусе.

BiLSTM. Основной архитектурой нейронной сети в данной работе являлась Bidirectional Long Short Term Memory (далее BiLSTM) [13]. Выбор этой архитектуры обусловлен тем, что она лучше всего подходит для нашей задачи. Рекуррентные нейронные сети (RNN) — в настоящее время стандартный метод решения задач для любых последовательностей, например, предложений в естественном языке. Среди всех типов рекуррентных сетей была выбрана именно LSTM, так как она лучше других справляется с долговременными зависимостями внутри предложения. Обратный проход (случай Bidirectional LSTM) важен для решения задачи нахождения семантем отрицания, потому что слово, находящееся под отрицанием, может быть и перед его триггером (например, меткой отрицания).

На вход нейронной сети подается трехмерная матрица X_{train} (подробности ее получения рассматривались ранее) размерности $S \times W \times E$, где S — это количество предложений в корпусе, $W = 30$ — максимальное количество слов в предложении, а $E = 100$ — размерность эмбеддинга. Матрица y_{train} ответов имеет размерность $S \times W \times 2$. Таким образом, каждое слово было представлено только своим эмбеддингом.

Вычисление скрытого слоя в момент времени t происходит следующим образом:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f),$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i),$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o),$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W_c x_t + U_c h_{t-1} + b_c),$$

$$h_t = o_t \cdot \tanh(c_t),$$

где x_t — входной вектор, h_t — выходной вектор, c_t — вектор состояний, W, U, b — матрицы и вектор параметров, f_t — вектор вентиля забывания (forget gate), i_t — вектор входного вентиля (input gate), o_t — вектор выходного вентиля (output gate), σ — функция активации на основе сигмоиды, $\tanh$ — функция активации на основе гиперболического тангенса.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

В качестве функции потерь была взята бинарная перекрестная энтропия (binary cross-entropy) $H(p, q)$:

$$H(p, q) = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right],$$

где y_i — предсказанная вероятность принадлежности классу, $\hat{y}_i$ — реальное значение класса, а N — длина предложения.

Размерность скрытого слоя составляла 128 единиц (соответственно, 256 для двух сконкатенированных LSTM-слоев в одном Bidirectional LSTM). Для предотвращения быстрого переобучения был применен dropout для Bidirectional LSTM-слоя в размере 20%, а также применялась L2-регуляризация. В качестве оптимизатора был взят Adam [14] с параметрами learning rate $= 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. На выходе, чтобы получить вероятности принадлежности каждому классу, использовался слой активации, представляющий собой сигмоиду:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Все использовавшиеся слои были реализованы с помощью библиотеки Keras [15] с бэкендом на TensorFlow [16]. Для отслежива-

ния прогресса обучения был использован инструмент TensorBoard [16], который позволяет в реальном времени после каждой эпохи визуализировать заданные метрики и таким образом принимать решение об окончании процесса обучения и о ходе обучения в целом. В качестве таких метрик были взяты значения функции потерь и доля правильных ответов, отдельно для валидационной и тестовой выборок.

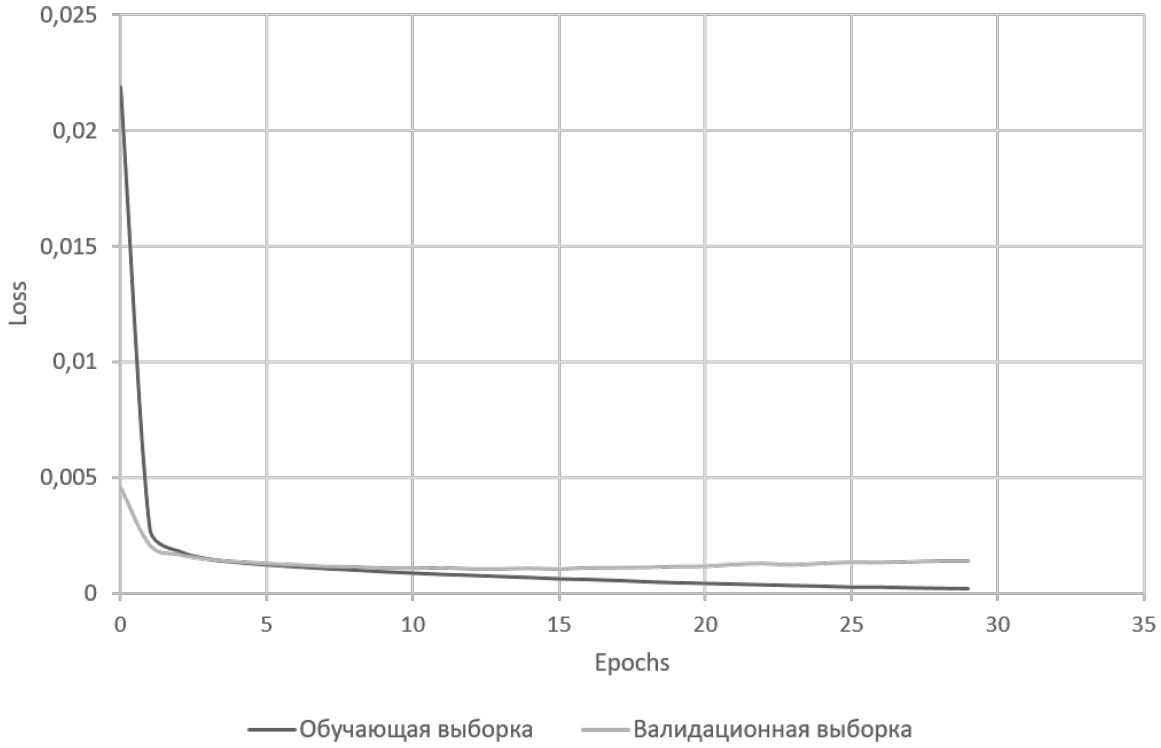

Рисунок 3 — График функции потерь

Обучение для русскоязычного корпуса проходило в течение 2 часов 15 минут, а для англоязычного — в течение 1 часа 7 минут. В обоих случаях в качестве вычислительного устройства использовалась GPU NVIDIA Tesla K80 на виртуальной машине, принадлежащей платформе Google Cloud. На рисунках 3 и 4 представлены графики процесса обучения на обучающей и валидационной выборках. По оси x обозначено количество эпох, а по оси y — соответствующая метрика.

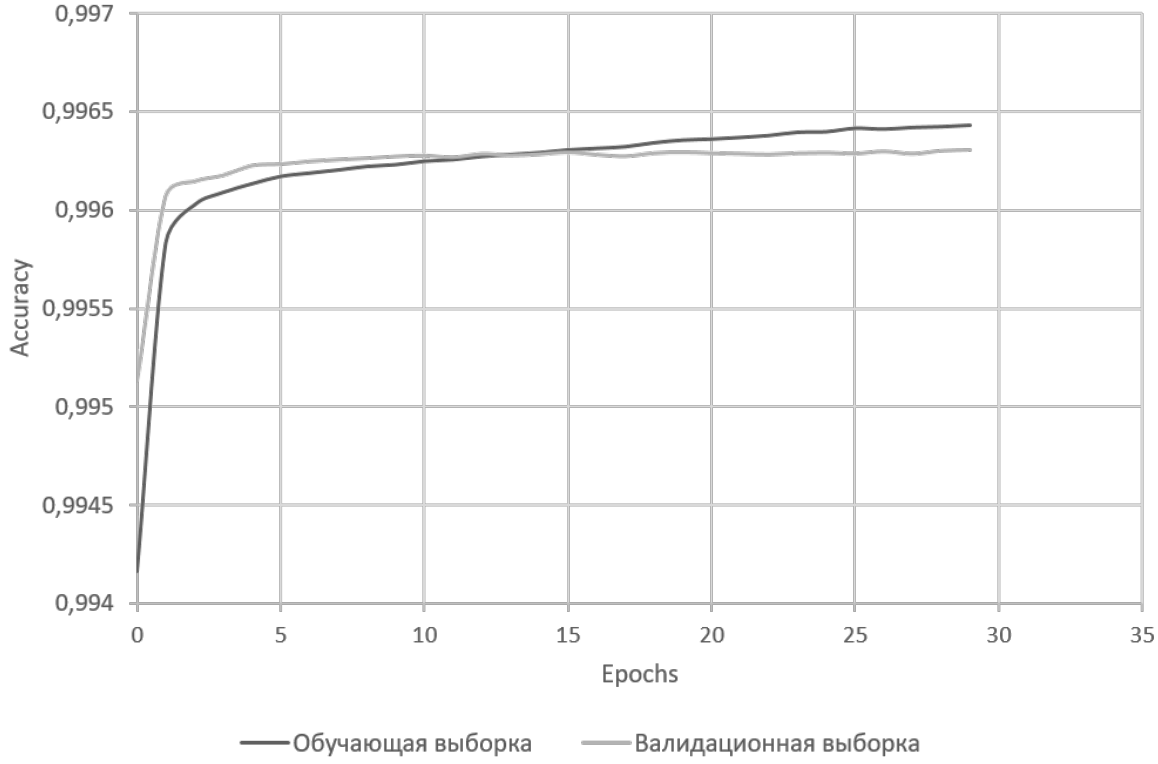

Рисунок 4 — График доли правильных ответов

Базовое решение. Для сравнения результатов на тестовой выборке корпуса на русском языке было реализовано базовое решение на основе выведенных правил. Оно было получено путем анализа примеров размеченных данных: какие части речи никогда не находятся под действием отрицания, какие слова являются триггерами для проставления слову метки класса отрицания, а какие служат разделителями между словом, находящимся в области отрицания, и соответствующим триггером, но сами не принадлежат классу отрицания.

В результате оно формулируется следующим образом.

Слово принадлежит классу отрицания тогда и только тогда, когда оно:

- внутри предложения находится непосредственно после частицы *не*;

- внутри предложения отделено от частицы *не* глаголом *быть* в любой форме или любым количеством предлогов;
- является частицей *нет*.

Метрики качества для оценки базового решения совпадают с общими метриками качества: в качестве метрик качества использовались точность, полнота и F1-мера.

Также при создании алгоритма базового решения была произведена попытка считать любые прилагательные разделителями слова, находящегося в области отрицания, и соответствующего триггера, однако такое решение дало качество хуже, чем описанное выше.

Для определения части речи слова и его нормальной формы использовался морфологический анализатор pymorphy2 [17], использующий словарь OpenCorpora и написанный на языке Python. Из всех возможностей pymorphy2 в данной работе используется приведение слова к нормальной форме и получение тега части речи для слова. В исходном словаре OpenCorpora слова объединены в лексемы, и библиотека pymorphy2 предоставляет быстрый доступ к всем возможным вариантам разбора слова, причем варианты отсортированы по частоте встречаемости в порядке убывания. Так, для словоформы «стали» можно получить следующие разборы:

```
    \Parse(word='стали', tag=OpencorporaTag('VERB,
perf, intr plur, past, indc'), normal_form='стать',
score=0.983766, methods_stack=((<DictionaryAnalyzer>,
'стали', 884, 4),)),
    \Parse(word='стали', tag=OpencorporaTag('NOUN,
inan, femn sing, gent'), normal_form='сталь',
score=0.003246, methods_stack=((<DictionaryAnalyzer>,
'стали', 12, 1),)),
```

Можно увидеть, что самый частотный разбор — это глагол с нормальной формой «стать», а вторым по частотности является разбор в качестве существительного с нормальной формой «сталь». Исходя из этого разбора, для каждого слова была получена его нормальная форма и часть речи, причем всегда берется первый, самый частотный, разбор.

Сравнение качества базового алгоритма с качеством предсказания нейронной сети будет проведено далее. Для корпуса на английском языке базовое решение не потребовалось, потому что результаты на нем были доступны для сравнения в рамках **SEM 2012 Shared Task*.

4. Результаты

4.1. Методы оценки

Все методы оценки результатов, использовавшиеся в данной работе, можно разделить на используемые для отслеживания прогресса обучения — *доля правильных ответов (accuracy)* и для получения итового качества — *точность (precision)*, *полнота (recall)* и *F_1-мера (F_1-score)*. В работе решается задача бинарной классификации: классификатор должен определить для данного слова, принадлежит ли оно классу NEGATED или классу NOT_NEGATED.

Воспользуемся стандартными обозначениями. Обозначим за TP количество слов, принадлежащих классу NEGATED, для которых классификатор ответил верно, за FN — для которых классификатор ответил неверно. Аналогично, пусть TN — верные ответы классификатора для класса NOT_NEGATED, а FP — неверные ответы для этого класса.

Метрика *доля правильных ответов (accuracy)* вводится следующим образом:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

Другими словами, это отношение количества правильных ответов ко всем ответам классификатора. С помощью такой метрики удобно отслеживать прогресс обучения на обучающей и валидационной выборках и строить соответствующие графики, однако о качестве результата в нашей задаче она не несет никакой информации.

Понятно, что из-за сильной несбалансированности классов такая метрика не подходит для оценки итоговых результатов, поэтому введем еще три общепринятые для подобных задач метрики: *точность (precision)*, *полноту (recall)* и F_1*-меру (*F_1*-score)*:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F_1\text{-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{(\text{Precision} + \text{Recall})}$$

4.2. Результаты на тестовой выборке

Как было упомянуто выше, и для русского, и для английского корпусов размер тестовой выборки равен 0.3 от общего количества предложений. Результаты были получены отдельно для корпусов на русском и английском языках. Обучение проводилось раздельно, но архитектура нейронной сети и общий подход к предобработке корпуса и оценке результатов одинаков для русского и английского языков. В данном разделе результаты будут рассмотрены по отдельности.

Корпус на русском языке. Для русского языка результат, полученный обучением нейронной сети, сравнивался с собственным базовым решением, описанным выше. Такое решение было принято из-за отсутствия корпусов на русском языке в свободном доступе и результатов, полученных на них. Метрики точность, полнота и F_1-мера, полученные на тестовой выборке корпуса на русском языке в результате предсказаний классификатора (BiLSTM) и в результате применения базового алгоритма (Baseline), описанного выше, сравниваются в таблице 4.

Таблица 4 — Результаты на русском корпусе

	Baseline	BiLSTM
Точность	0.929	**0.961**
Полнота	**0.969**	0.965
F_1-мера	0.949	**0.963**

Корпус на английском языке. Рассматривающийся в данной работе корпус на английском языке был представлен на открытом контесте **SEM 2012 Shared Task*, прошедшем в 2012 году и приуроченном к конференции *SEM 2012 в рамках конференции NAACL-HLT 2012, прошедшей в Канаде.

Организаторы предоставили обучающий и тестовый корпус, принимали работы и оценивали результаты своими алгоритмами. В рамках контеста участникам было предложено определить область действия отрицания, метки отрицания, отрицаемое событие или свойство, а также фокус отрицания. По итогам соревнования организаторы представили результаты 14-ти различных прогонов, причем 12 из них касаются задачи определения области действия отрицания и отрицаемых событий и свойств.

Для оценки результатов использовались абсолютно те же метрики, что и в данной работе, а также для обучения и предсказания использовался тот же корпус. Именно поэтому возможно сравни-

вать результаты, полученные в данной работе, и результаты, доступные по итогам проведения контеста **SEM 2012 Shared Task*.

В таблице 5 представлено качество определения отрицаемого события или свойства в вышеупомянутом контесте, а также результаты, полученные в данной работе (BiLSTM).

Таблица 5 — Результаты на английском корпусе

Название	Точность	Полнота	F_1 мера
UiO1 r2	0.606	**0.750**	**0.670**
UiO1 r1	0.601	0.729	0.661
FBK	0.641	0.567	0.602
BiLSTM	0.628	0.578	0.602
UiO2	**0.682**	0.526	0.594
UWashington	0.580	0.509	0.543
UMichigan	0.500	0.522	0.511
UABCoRAL	0.650	0.385	0.483

4.3. Анализ ответов и ошибок

В таблицах 6 и 7 обозначены результаты для корпуса на русском языке, разделенные на случаи, которые могут представлять особый интерес. Метрики качества аналогичны использовавшимся для оценки общих результатов на тестовой выборке. В таблице проведено сравнение базового алгоритма (Baseline) и классификатора (BiLSTM), а также указан процент встречаемости конкретного случая среди всех размеченных классом NEGATED токенов.

Видно, что во всех случаях F_1-мера, полученная классификатором, выше, чем у базового решения. В то же время легко отследить особенности отрицания в русском языке и разметки: чаще всего слово, размеченное классом NEGATED, находится сразу после частицы *не*, что с легкостью покрывает базовое решение. Однако особый интерес представляют более сложные случаи, например, когда

Таблица 6 — Анализ ответов и ошибок — базовое решение

Случай	% от разм.	Baseline		
		Точность	Полнота	F_1 мера
Размеченное слово сразу после «не»	94.0%	0.97	0.98	0.97
Размеченное слово сразу перед «не»	0.9%	0.00	0.00	0.00
Размеченное слово через 1 после «не»	5.6%	0.56	**0.86**	0.67
Размеченное слово через 2–3 слова после «не»	0.1%	0.24	**0.89**	0.37
Размеченное слово является словом «нет»	4.0%	0.88	1.00	0.93

слово под классом NEGATED не находится в непосредственной близости от своей метки отрицания. Как видно, нейронная сеть справляется с такими случаями достаточно успешно.

Теперь можно рассмотреть нетривиальные примеры верных ответов классификатора, когда размеченное классом NEGATED слово находится не сразу после *не*. Здесь и далее полужирным шрифтом обозначены ответы нейронной сети, а подчеркнуты элементы эталонной разметки.

Пример:

- Надеюсь, что реформа не будет **носить** ограничительный конфискационный характер.
- Полученный купонный доход налогом **облагаться** не будет.

Далее приведем примеры ошибок (несовпадения с эталонной разметкой) классификатора, определим возможные причины их возникновения, а также рассмотрим различные классы ошибок. В

Таблица 7 — Анализ ответов и ошибок — BiLSTM

Случай	% от разм.	BiLSTM		
		Точность	Полнота	F_1 мера
Размеченное слово сразу после «не»	94.0%	**0.97**	**0.99**	**0.98**
Размеченное слово сразу перед «не»	0.9%	**0.90**	**0.77**	**0.81**
Размеченное слово через 1 после «не»	5.6%	**0.84**	0.79	**0.81**
Размеченное слово через 2–3 слова после «не»	0.1%	**0.83**	0.73	**0.78**
Размеченное слово является словом «нет»	4.0%	**1.00**	**0.99**	**0.99**

первую очередь, это неоднозначные случаи, когда без полного контекста предложения неясно, к чему относится отрицание.

> **Пример:**
>
> - Депутаты при голосовании руководствовались не **политическими** мотивами, а личной заинтересованностью в распределении собственности.
> - Государство несет обязательства по вкладам граждан в Сбербанк, а не по **его кредитной** деятельности, поэтому необходимо канонизировать действия Сбербанка.

В то же время из-за того, что эталонная разметка проводилась не вручную, а алгоритмически, допустимы ошибки эталонной разметки из-за особенностей алгоритма. Некоторые из них приведены в следующем примере.

Пример:

- Это не **первый** объект Министерства энергетики и промышленности Туркмении, реализация которого осуществляется за счет внешних займов.
- Не **главное**, что конкурс и аукцион выиграла Лагуна, — главное, что кредит правительству будет предоставлен банком Менатеп, — сказал Кох.

И в заключение, приведем в таблице 8 некоторую статистику по всем ошибкам классификатора и разделим их на категории.

Таблица 8 — Анализ ошибок классификатора

Случай	BiLSTM
Ошибки эталонной разметки	29%
Неоднозначные случаи	6%
Явные ошибки классификатора	65%

4.4. Объединение с информацией об антонимической полярности

Определение семантем отрицания само по себе, безусловно, интересная с лингвистической точки зрения задача, однако особенный прикладной интерес представляет объединенная информация об отрицаниях и антонимической полярности. В рамках данной работы такое объединение являлось не первостепенной задачей, поэтому для ее решения был взят «Словарь антонимов русского языка» М. Р. Львова, содержащий 2128 антонимических пар [18].

На вход реализованному алгоритму можно подать любое предложение, содержащее в себе отрицание или не содержащее. Алгоритм определяет, какие слова принадлежат классу NEGATED, и

подбирает к нему антонимическую пару из словаря. Таким образом получается синонимичное исходному предложение.

> **Пример:**
> Наша забастовка *не* от *хорошей* жизни: *не главное*, что конкурс и аукцион выиграла Лагуна; главное, что многие будут закупаться *недешевыми* бумагами.
> Наша забастовка от *плохой* жизни: *второстепенное*, что конкурс и аукцион выиграла Лагуна; главное, что многие будут закупаться *дорогими* бумагами.

Вспомним, что основная область применения поиска отрицаний и объединения информации о них с антонимической полярностью — поисковые системы. Поэтому приведем также пример, связанный с поисковыми запросами. Допустим, существует следующий проиндексированный текст:

> **Пример:**
> По его словам, генеральный директор и коллектив АО считают, что государство *не сохранит* за собой большую часть акций Рыбинских Моторов. Срок внесения инвестиций — *не позднее* февраля 1996 года. Андрей Даценко *вспомнил*, сколько компаний находится на листинге РТС.

Этот текст будет включен в поисковую выдачу благодаря полученному алгоритму в результате следующих запросов:

> **Пример:**
>
> - Государство *откажется* от большей части акций Рыбинских Моторов.
> - Срок внесения инвестиций — *до* февраля 1996 года.

- Андрей Даценко *не забыл*, сколько компаний находится на листинге РТС.

Безусловно, когда дело касается антонимической полярности, возникает множество неоднозначностей. Например, вследствие того факта, что слово «осень» не является однозначным антонимом к слову «весна», очевидно, что выражение «не осень» вовсе не означает «весна». Однако такие случаи выходят за рамки данной работы.

Заключение

В результате проделанной работы можно сделать следующие выводы: рекуррентные нейронные сети успешно справляются с задачей определения отрицаемых событий или свойств в большинстве случаев, однако существует множество неоднозначностей и сложных способов выражения отрицания на естественном языке, которые нейронная сеть с простой архитектурой определить не может.

В то же время видно, что в случае русского языка даже самое простое базовое решение дает неплохой результат, а нейронная сеть его улучшает лишь в некоторых сложных случаях. Поэтому возможные пути улучшения результатов работы классификатора — добавление информации, связанной с синтаксической структурой предложения, или каких-либо других лингвистических признаков. Но при этом важно избегать вычислительно сложных алгоритмов (например, полного построения дерева синтаксического разбора) и инструментов, работа которых основана на правилах, специфических для конкретных языков.

Также рассмотрен пример прикладного использования полученного способа определения слов, находящихся под отрицанием: объединение информации об отрицании и антонимической поляр-

ности. Однако и в этом случае возникает множество неоднозначностей, которые могут стать темой для дальнейших исследований.

Список литературы

1. *Huddleston R. D.*, *Pullum G. K.* The Cambridge Grammar of the English Language. — Cambridge University Press, 04.2002.
2. *Morante R.*, *Blanco E.* *SEM 2012 Shared Task: Resolving the Scope and Focus of Negation // Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. — Montréal, Canada : Association for Computational Linguistics, 2012. — С. 265—274. — (SemEval '12). — URL: http://dl.acm.org/citation.cfm?id=2387636.2387679.
3. *Fancellu F.*, *Lopez A.*, *Webber B. L.* Neural Networks For Negation Scope Detection // ACL. — 2016.
4. UiO1: Constituent-based Discriminative Ranking for Negation Resolution / J. Read [и др.] // Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. — Montréal, Canada : Association for Computational Linguistics, 2012. — С. 310—318. — (SemEval '12). — URL: http://dl.acm.org/citation.cfm?id=2387636.2387686.
5. *Chowdhury M. F. M.* FBK: Exploiting Phrasal and Contextual Clues for Negation Scope Detection // *SEM@NAACL-HLT. — Association for Computational Linguistics, 2012. — С. 340—346.

6. Simple Negation Scope Resolution through Deep Parsing: A Semantic Solution to a Semantic Problem / W. Packard [и др.] // ACL. — 2014.
7. *Loper E.*, *Bird S.* NLTK: The Natural Language Toolkit // In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics. — 2002.
8. The Sketch Engine: ten years on / A. Kilgarriff [и др.] // Lexicography. — 2014. — С. 7—36.
9. Bag of Tricks for Efficient Text Classification / A. Joulin [и др.] // CoRR. — 2016. — Т. abs/1607.01759. — arXiv: 1607.01759. — URL: http://arxiv.org/abs/1607.01759.
10. *Schmid H.* Probabilistic Part-of-Speech Tagging Using Decision Trees. — Manchester, UK., 1994. — URL: http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/.
11. *Kutuzov A.*, *Kuzmenko E.* WebVectors: A Toolkit for Building Web Interfaces for Vector Semantic Models // Analysis of Images, Social Networks and Texts: 5th International Conference, AIST 2016, Yekaterinburg, Russia, April 7-9, 2016, Revised Selected Papers / под ред. D. I. Ignatov [и др.]. — Cham : Springer International Publishing, 2017. — С. 155—161. — ISBN 978-3-319-52920-2. — DOI: 10.1007/978-3-319-52920-2_15. — URL: http://dx.doi.org/10.1007/978-3-319-52920-2_15.
12. *Řehůřek R.*, *Sojka P.* Software Framework for Topic Modelling with Large Corpora // Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. — Valletta, Malta : ELRA, 05.2010. — С. 45—50. — http://is.muni.cz/publication/884893/en.
13. *Hochreiter S.*, *Schmidhuber J.* Long Short-term Memory. — 1997.

14. *Kingma D. P.*, *Ba J.* Adam: A Method for Stochastic Optimization // CoRR. — 2014. — Т. abs/1412.6980.

15. Keras / F. Chollet [и др.]. — 2015. — https://keras.io.

16. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems / Martín Abadi [и др.]. — 2015. — URL: https://www.tensorflow.org/ ; Software available from tensorflow.org.

17. *Korobov M.* Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts. Т. 542 / под ред. M. Y. Khachay [и др.]. — Springer International Publishing, 2015. — С. 320—332. — (Communications in Computer and Information Science). — ISBN 978-3-319-26122-5. — DOI: 10.1007/978-3-319-26123-2_31. — URL: http://dx.doi.org/10.1007/978-3-319-26123-2_31.

18. *Львов М.*, *Новиков Л.* Словарь антонимов русского языка: более 2,000 антонимических пар. — Русский язык, 1988. — URL: https://books.google.nl/books?id=JfNJAAAAYAAJ.