РЕФЕРАТ

Выпускная квалификационная работа содержит 30 страниц, среди которых 8 рисунков, 10 таблиц и 8 использованных источников.

МАШИННОЕ ОБУЧЕНИЕ, АНАЛИЗ ТОНАЛЬНОСТИ, КЛАССИФИКАЦИЯ ТЕКСТОВ, СЛОВАРИ ТОНАЛЬНОСТИ, ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

Целью данной работы является разработка приложения для осуществления сентимент-анализа текста художественного произведения.

Объект исследования: задача определения тональности текста русскоязычной художественной литературы.

Субъект исследования: размеченные предложения произведения М. Булгакова «Мастер и Маргарита».

Предмет исследования: классические методы машинного обучения и методы, основанные на искусственных нейронных сетях.

Для достижения поставленной цели выполнены следующие подзадачи: изучен теоретический материал, относящийся к методам машинного обучения, глубокому обучению нейронными сетями, а также их особенности в обработке естественных языков; изучены документации различных библиотек, используемых для работы с естественным языком, методами машинного обучения и нейронными сетями; разработаны модели классификаторов; произведен сравнительный анализ точности моделей классификаторов по нескольким метрикам качества.

В результате реализован метод автоматической классификации предложений текста художественного произведения, содержащих эмотивную составляющую.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc45874993)

[ОСНОВНАЯ ЧАСТЬ 6](#_Toc45874994)

[1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 7](#_Toc45874995)

[1.1 Задача классификации текста 7](#_Toc45874996)

[1.1.1 Предобработка текста 8](#_Toc45874997)

[1.1.2 Работа классификатора 12](#_Toc45874998)

[1.1.3 Оценка качества работы классификатора 13](#_Toc45874999)

[1.2 Задача анализа тональности текста 16](#_Toc45875000)

[1.2.1 Проблематика в задаче определения тональности текста 17](#_Toc45875001)

[1.3 Используемые методы машинного обучения 18](#_Toc45875002)

[2 ПРАКТИЧЕСКАЯ ЧАСТЬ 21](#_Toc45875003)

[2.1 Язык программирования и библиотеки 21](#_Toc45875004)

[2.2 Используемый словарь тональности 22](#_Toc45875005)

[2.3 Используемая модель Word2Vec 24](#_Toc45875006)

[2.4 Исследуемые методы классификации 25](#_Toc45875007)

[2.5 Эксперименты 25](#_Toc45875008)

[2.5.1 Сравнение и оценка результатов 29](#_Toc45875009)

[ЗАКЛЮЧЕНИЕ 30](#_Toc45875010)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 31](#_Toc45875011)

ВВЕДЕНИЕ

Информация играет огромную роль в жизни людей. Но с развитием информационных технологий, её потоки значительно увеличились в последние несколько лет. В связи с этим в нашей жизни возникла потребность автоматически анализировать потоки текстовых и не только данных по разным направлениям, одним из которых является сентимент-анализ.

Анализ тональности текста ­­­– это развивающиеся направление компьютерной лингвистики, задачей которого является выявление в документе эмоционально окрашенной лексики или эмоциональной оценки авторов по отношению к объектам, присутствующим в тексте.

Сентимент-анализ актуален как методика для мониторинговых, аналитических и сигнальных систем, для рекламных платформ, нацеленных на контекстную тематику страниц. Он применяется в таких областях как, к примеру, маркетинг, социология, рекомендательные сервисы, психология и другие. В сентимент-анализе классическим подходом служит классификация ключевых слов на две категории – негативные и позитивные, в некоторых случаях добавляется третья – нейтральная. В данной работе рассматривается иной подход – многоклассовый вариант.

Обычно в качестве исследуемых данных используются комментарии в социальных сетях, рецензии, пользовательские отзывы. В представленной работе исследуется текст художественной литературы, который имеет преимущество в наличии широкого спектра эмоций и в чистоте и грамотности естественного языка.

В работе рассматривается задача классификации художественных текстов, а именно предложений, содержащих в себе сложный сентимент: злость, страх, радость, неудовольствие, ожидание, грусть, доверие, удивление. Исследуются методы классификации на основе словаря тональностей, машинного обучения и нейронных сетей.

Новизна исследуемой области проявляется в проведении многоклассовой классификации на нетипичных данных в виде художественной литературы на предмет выявления восьми основных эмоций и изучении русскоязычного текста.

Интерес ученых к сфере эмоционального интеллекта обуславливается, например, актуальностью тестирования людей с целью отбора соискателей на определённую работу или школьников в соответствующие специализированные классы. Тесты, популярные в XX веке, в настоящее время всё реже применяются, поэтому возникает потребность в новом подходе к оценке успешности личности. Ранее интеллектуальная сторона жизни человека противопоставлялась его эмоциональной составляющей. В настоящее время признаётся, что эмоция, как особый тип знания, может дать человеку возможность успешно адаптироваться к условиям окружающей среды и соотносится с категорией интеллект.

Отправной точкой для работы послужил вычислительный подход, рассмотренный в [7], нацеленный на оценку технического качества и тонкости рифм. В результате проделанной работы был разработан сервис, способный визуализировать стихотворения в виде так называемого «поэтического штрих-кода», при котором каждая строчка представляет собой горизонтальную линию, окрашенную в определённый цвет, характеризующий ее эмотивную составляющую. При таком подходе легко проследить, как меняется тональность по ходу стихотворения.

В качестве результата было разработано приложение, в основе которого лежит классификатор, способный предсказывать эмотивную составляющую художественного произведения с определенной точность, полученной в результате сравнения метрик качества. Результаты работы могут быть использованы в дальнейшем для выявления эмоций между двумя определенными персонажами художественного произведения. Это позволит извлечь эмотивную часть картины какого-либо конечного мира.

# ОСНОВНАЯ ЧАСТЬ

# 1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1.1 Задача классификации текста

С каждым годом объем информации в сети Интернет увеличивается, в связи с этим экспертам из разных областей требуется постоянно проводить анализ, фильтрацию и обработку данных. Раздел компьютерной лингвистики, в частности, задача классификации текстов, позволяет раскрыть различные аспекты текстовых данных, такие как: поиск смысловых и семантических цепочек, определение темы, установление авторства или выявление тональной окраски.

Определение 1.1 Задача классификации текстов (документов) – задача компьютерной лингвистики, заключающаяся в отнесении текста к одной из нескольких категорий на основании содержания документа (его вектора признаков).

Определение 1.2 Классификатор – это алгоритм, способный отнести произвольный объект из исходной выборки к одному или нескольким классам, множество которых определено заранее.

Пусть имеется множество классов (меток) , зафиксированное заранее, множество документов , множество всех употребляемых в этих документах слов W. Документ 𝑑 ∈ 𝐷 представляет собой последовательность слов из словаря W, где – длина документа по количеству слов. Необходимо присвоить булево значение каждой паре , где значение 1 означает, что документ относится к классу , а значение 0 означает обратное. То есть имеется неизвестная целевая функция :

Требуется найти функцию максимально близкую к . Функция и есть искомый классификатор.

Изначально имеем размеченную коллекцию документов . Делим её на две непересекающиеся части – «обучающую» и «тестовую». «Обучающая» используется для обучения классификатора , «тестовая» – для тестирования эффективности построенного классификатора. Каждый тестовый документ подается на вход , затем результат классификатора сравнивается с известным значением функции . Чем чаще эти значения совпадают, тем эффективнее классификатор.

Для решения задачи классификации используются методы машинного обучения.

Определение 1.3 Машинное обучение – класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение в процессе применения решений множества сходных задач.

Для автоматической классификации текстов могут использоваться такие методы как Метод k-ближайших соседей (k-Nearest Neighbors, kNN), Метод Байеса (Naïve Bayes, NB), Метод опорных векторов (Support Vector Machine, SVM). В [1] приводится описание и сравнение описанных подходов, рассматриваются их преимущества и недостатки.

Для решения более сложных задач классификации используются нейронные сети.

Определение 1.4 Искусственные нейронные сети – семейство математических моделей, построенных по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма

Классификацию можно разделить на три основных этапа:

1. Предобработка текста (документа)
2. Работа классификатора
3. Оценка качества работы классификатора

1.1.1 Предобработка текста

На первом этапе обработки необходимо удалить из текста все слова, не имеющие какой-либо эмотивной составляющей, так называемые «стоп-слова». Это могут быть, например, предлоги, союзы, местоимения и знаки препинания.

Далее необходимо привести все слова к нормальной форме – осуществить лемматизацию.

В некоторых случаях дополнительным шагом является использование стемминга (процесс поиска основы для выбранного слова). Этот прием помогает избежать присутствия нескольких форм одного и того же слова, например, «тяжело» и «тяжелый».

Любой алгоритм, который работает с текстовыми данными, нуждается в некотором представлении слов в виде чисел, поскольку компьютеры не понимают текст напрямую. Таким образом, входные слова должны быть преобразованы в форму, понятную для алгоритма. Для этого существует несколько способов.

Bag of words

Можно представить документ в виде вектора, в котором каждый элемент соответствует слову из словаря по всем документам, а численное значение сообщает, в каком количестве конкретное слово встретилось в текущем документе.

Таблица 1.1 - Пример векторного представления предложения

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Маргарита | Николаевна | не | нуждалась | в | деньгах |
| Документ | 1 | 0 | 0 | 1 | 1 | 1 |

Для предложения «Маргарита нуждалась в деньгах» матрица документ-слово будет выглядеть следующим образом. (Таблица 1.1). Такой подход называется «мешок слов» (*Bag of Words, BoW*).

TF-IDF

Другой подход представления документа в виде вектора заключается в получении частоты встречаемости слов в данном документе в виде индекса 𝑇𝐹×𝐼𝐷𝐹 (term frequency × inverse document frequency). Term frequency – частота слова, которая рассчитывается как количество вхождений конкретного слова в конкретный документ, делённое на общее количество слов в этом документе:

Определение 1.5 Document frequency (частота документа, DF) – количество документов, в которых данный термин встречается, деленное на общее количество документов.

Определение 1.6 Inverse document frequency (величина, обратная частоте документа) ) . Обычно вместо самого значения берётся его логарифм, чтобы сгладить результат действия на общий результат.

чем выше индекс TF-IDF, тем больше релевантность данного термина в конкретном документе.

Word2Vec

Описанные выше подходы хороши для текстов, имеющих ограниченный словарь небольшого объема. Как только словарь начнет заметно увеличиваться в объеме, мы получим матрицы признаков огромного размера, что усложняет обработку входных данных для алгоритмов машинного обучения. Но главный недостаток таких методов заключается в том, что они не учитывают контекста анализируемого слова в предложении. Тогда на замену пришел новый подход, основанный на гипотезе – “слова, которые встречаются в одинаковых окружениях, имеют близкие значения”.

Определение 1.7 Word2Vec – совокупность моделей на основании искусственных нейронных сетей, предназначенных для получения векторных представлений слов на естественном языке.

Работа осуществляется следующим образом: word2vec принимает большой текстовый корпус в качестве входных данных и сопоставляет каждому слову вектор, выдавая координаты слов на выходе. Сначала он генерирует словарь корпуса, а затем вычисляет векторное представление слов, «обучаясь» на входных текстах. Векторное представление основывается на контекстной близости: слова, встречающиеся в тексте рядом с одинаковыми словами (а следовательно, имеющие схожий смысл), будут иметь близкие (по косинусному расстоянию) векторы (Рис. 1.1).

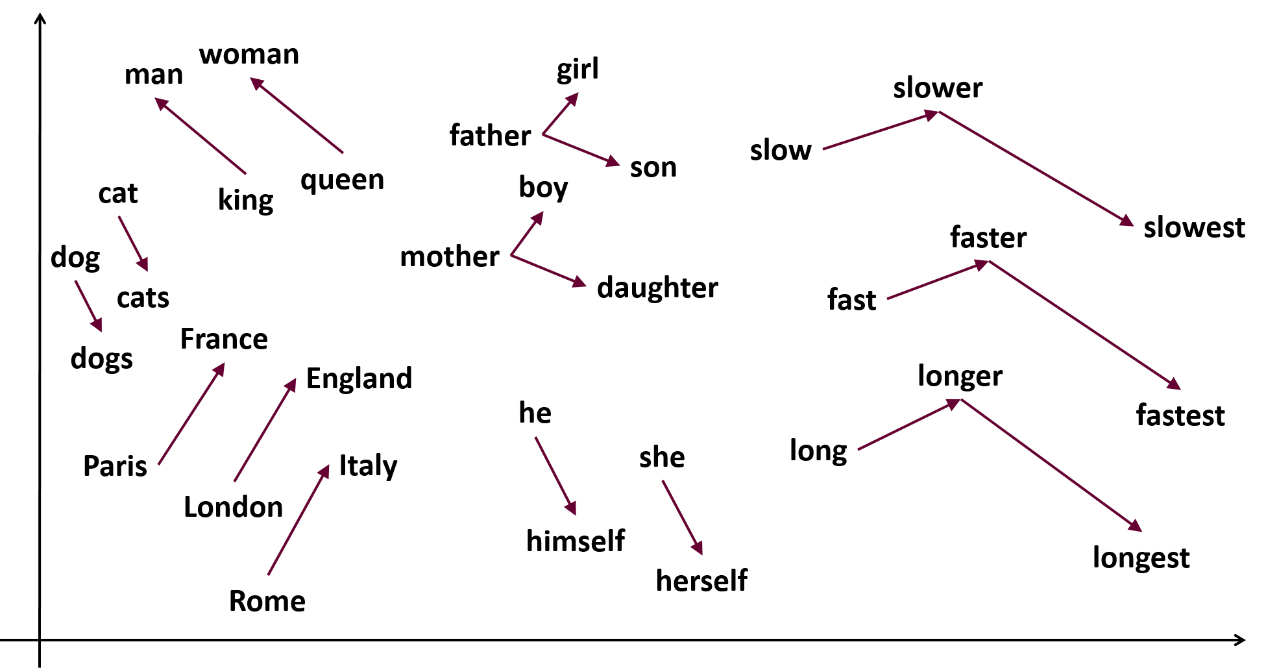


Рис. 1.1 Пример расположения слов в двумерном пространстве

Например, для единственного предложения «Маргарита не нуждалась в деньгах.» получим словарь с соответствующими векторами «one-hot encoding»

Таблица 1.2 - Пример кодирования слов по методу «one-hot encoding»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Маргарита | 1 | 0 | 0 | 0 | 0 |
| не | 0 | 1 | 0 | 0 | 0 |
| нуждалась | 0 | 0 | 1 | 0 | 0 |
| в | 0 | 0 | 0 | 1 | 0 |
| деньгах | 0 | 0 | 0 | 0 | 1 |

Проблема такого кодирования в том, что оно не улавливает отношения между словами, так как все векторы независимы между собой. Косинусное расстояние(угол) между такими векторами будет равно 0.

Word2Vec существует в двух вариантах: подход, основанный на модели Skip Gram и на модели Continuous Bag of words.

Continuous Bag of words

В данной модели мы пытаемся предсказать слово, опираясь на окружающие его слова (слова контекста). На вход нейронной сети прямого распространения подается one-hot encoded вектор четырех ближайших слов (два слева и два справа). На выходе получаем один вектор – вектор предсказанного слова.

Skip Gram

Здесь мы пытаемся предсказать контекст по данному слову. На вход подается вектор слова, на выходе получаем вектора контекста (Рис. 1.2).

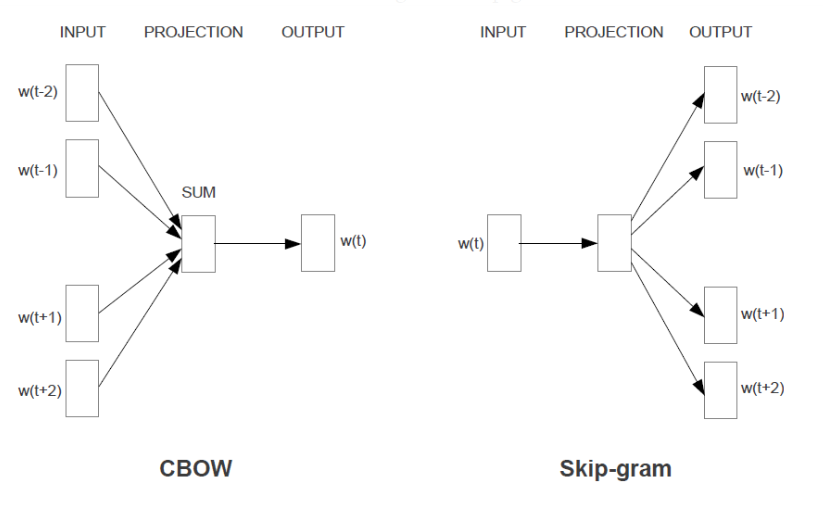


Рис. 1.2 Архитектура моделей CBOW и Skip-gram

Получить вектор для целого предложения можно путем усреднения векторов по каждому слову этого предложения.

### 1.1.2 Работа классификатора

В машинном обучении:

Текст классифицируется путем сравнения с ранее размеченным экспертом корпусом по выбранной мере близости. Далее он относится к какому-либо классу на основании полученного результата сравнения.

Достоинства:

* работает достаточно быстро

Недостатки:

* требует заранее размеченного экспертом корпуса
* неполнота лексического покрытия, которая приводит к потере точности

В словарях тональностей:

Поиск эмотивной лексики в тексте происходит при помощи тональных словарей с проведением лингвистического анализа.

Определение 1.8 Словарь тональной лексики – список слов или словосочетаний, каждому из которых присваивается оценка, характеризующая его тональную окраску.

В методе используются как списки паттернов, подставляемые в регулярные выражения, так и некоторые правила соединения тональной лексики непосредственно внутри документа.

Достоинства:

* более гибок в сочетании с синтаксическим и морфологическим словарем
* при хорошем наполнении словаря тональности позволяет достичь хорошей полноты (покрытия эмотивной лексики)

Недостатки:

* трудоемок в составлении тональных словарей
* сложно дать количественную оценку негативности-позитивности текста

### 1.1.3 Оценка качества работы классификатора

Главная задача алгоритмов обучения – способность обобщаться, то есть хорошо работать на новых данных. Так как на новых данных мы не можем проверить качество построенного классификатора (на них делается прогноз, то есть данные о классах отсутствуют), нужно отделить порцию данных от исходной разметки, чтобы на ней проверять качество.

Есть два основных способа сделать это:

* отложенная выборка (*hold-out*), некоторая доля от общего набора данных (20-40%). Обучаем модель на остальных данных (60-80%) и считаем некоторую метрику качества на отложенных
* кросс-валидация (cross-validation), например, K-fold кросс-валидация. На Рис. 1.3 модель обучается K=5 раз на четырех подвыборках исходной выборки (каждый раз на разной, оранжевый цвет). Получается K=5 оценок качества модели, которые усредняются, выдавая среднюю оценку качества классификации (Рис. 1.3).

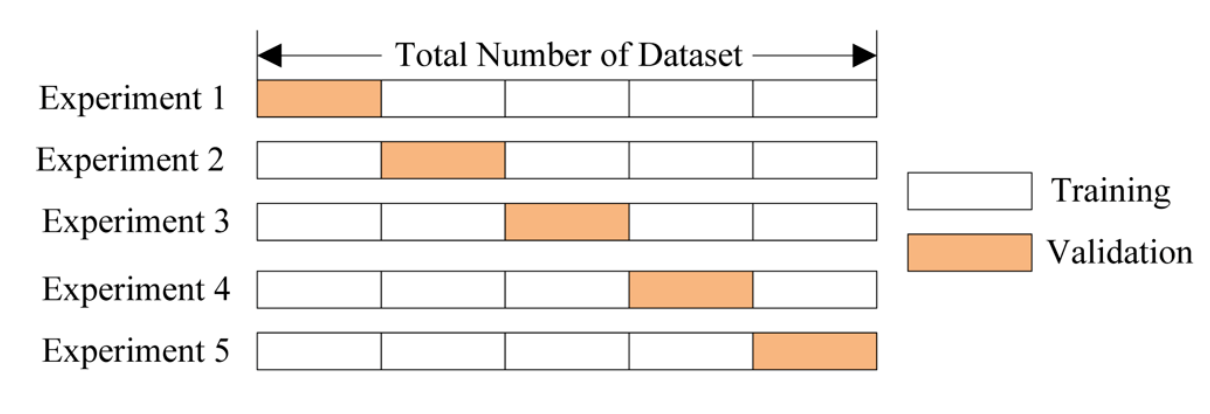


Рис. 1.3 Визуальное представление разбиения исходной выборки для кросс-валидации

Метрики качества.

Наиболее распространенная метрика оценки качества классификации включает в себе оценку двух характеристик классификатора – точности и полноты.

Определение 1.9 Точность (*precision*) – доля документов, действительно принадлежащих данному классу относительно всех документов, которые система отнесла к этому классу.

Определение 1.10 Полнота (*recall*) – доля найденных классификатором документов, принадлежащих классу относительно всех документов этого класса.

Таблица 1.3 - Категории, получаемые в результате работы классификатора

|  |  |  |
| --- | --- | --- |
|  | Ожидалось | |
| Получили | TP  (*true positive*) | FP  (*false positive*) |
| FN  (*false negative*) | TN  (*true negative*) |

Таблица 1.3:

* истинно-положительные (*true positive*) – категории, которые мы ожидали увидеть и получили на выходе
* ложно-положительные (*false positive*) – категории, которых быть на выходе не должно, но классификатор их ошибочно вернул на выходе
* ложно-отрицательные (*false negative*) – категории, которые должны быть на выходе, но классификатор их не определил
* истинно-отрицательные (*true negative*) – категории, которых быть на выходе не должно и на выходе классификатора они отсутствуют

Левая колонка таблицы – документы и категории, которые мы ожидаем на выходе, правая – неправильные сочетания. Верхняя строка – положительные ответы классификатора, нижняя – отрицательные.

Метрика F1

Мера точности определяется как:

Чем больше точность, тем меньше число ложных попаданий. Но мера точности не дает представление о том, все ли правильные ответы вернул классификатор. Тогда дополнительно определяется полнота:

Мера полноты характеризует способность классификатора определять наибольшее число положительных ответов из ожидаемых.

При определении качества оценки классификатора приходится находить баланс между двумя этими метриками, так как если увеличить полноту, произойдет уменьшение точности из-за увеличения числа ложно-положительных ответов. А при росте точности произойдет падение полноты из-за отсеивания некоторого числа правильных ответов.

Таким образов, обычно используется метрика F1, являющаяся средним гармоническим величин *P* и *R*.

## 1.2 Задача анализа тональности текста

Определение 1.11 Анализ тональности текста(сентимент-анализ) – класс методов, предназначенный для автоматического выявления эмотивно окрашенной лексики в текстах, а также эмоциональной оценки авторов по отношению к каким-либо объектам.

Важно понимать что для сентимент-анализа основными понятиями являются субъект тональности, объект тональности и тональность высказывания.

Определение 1.12 Субъект тональности – сущность, которой принадлежит мнение.

Определение 1.13 Объект тональности – сущность, по отношению к которой высказывается мнение.

Определение 1.14 Тональность – это эмотивное отношение автора высказывания к некоторому объекту (объекту реального мира, событию, процессу или их свойствам), выраженное в тексте. Лексической тональностью (или сентиментом) называется выраженная на уровне лексемы или фрагмента текста эмотивная составляющая.

Анализ тональности может проводиться на трех уровнях:

* Уровень документа (общая тональность текста или отношения автора к объекту)
* Уровень предложения (общая тональность, данный уровень тесно связан с задачей определения субъективности)
* Аспектуальный уровень (задача определения того, что именно нравится или не нравится автору)

### 1.2.1 Проблематика в задаче определения тональности текста

Основной проблемой в задаче определения тональности текста является субъективность оценки. Один и тот же текст может по-разному оцениваться разными людьми. Мнения экспертов могут носить полярно противоположный характер – одни относят текст к положительному классу, другие – к отрицательному.

Также актуальна проблема различия структуры естественных языков. Методы, разработанные для одного языка, могут не работать для другого. Это касается, например, подходов, применяемых для текстов на английском языке, которые показывают низкое качество работы для текстов на русском языке. В [2] описывается методика и промежуточные результаты создания словаря тональностей русского языка для его дальнейшего использования в задачах сентимент-анализа. Сперва осуществлялась ручная разметка тональности прилагательных и наречий, основывавшаяся на гипотезе о том, что оценочные значения чаще всего выражаются в языках при помощи признаковых слов. Далее словарь был расширен и ориентирован на тексты социальных медиа, при помощи тематического моделирования, путем вычленения характерной для данной предметной области слов. Наконец, данный словарь был вынесен на краудсорсинговую платформу, на которой в настоящее время проводится разметка эмоциональной окраски самих слов и отобранных текстов.

## 1.3 Используемые методы машинного обучения

Метод ближайших соседей

Метод ближайших соседей (k Nearest Neighbors, или kNN) — популярный метод и один из самых понятных подходов к классификации. На уровне интуиции суть метода такова: посмотри на соседей, какие преобладают, таков и ты. Формально основой метода является гипотеза компактности: если метрика расстояния между примерами введена достаточно удачно, то схожие примеры гораздо чаще лежат в одном классе, чем в разных.

Для классификации каждого из объектов тестовой выборки необходимо последовательно выполнить следующие операции:

* Вычислить расстояние до каждого из объектов обучающей выборки
* Отобрать k объектов обучающей выборки, расстояние до которых минимально
* Класс классифицируемого объекта — это класс, наиболее часто встречающийся среди k ближайших соседей

В чистом виде kNN может служить хорошим стартом (baseline) в решении какой-либо задачи. Качество классификации по данному методу зависим от некоторых параметров:

* Чисто соседей k
* Метрика расстояния между объектами (часто используется метрика Хэмминга, евклидово расстояние, косинусное расстояние и расстояние Минковского)
* Веса соседей

Дерево решений

Дерево решений как алгоритм машинного обучения – объединение логических правил вида "Значение признака *a* меньше *x* И Значение признака *b* меньше *y* … => Класс 1" в структуру данных "Дерево". Огромное преимущество деревьев решений в том, что они легко интерпретируемы, понятны человеку.

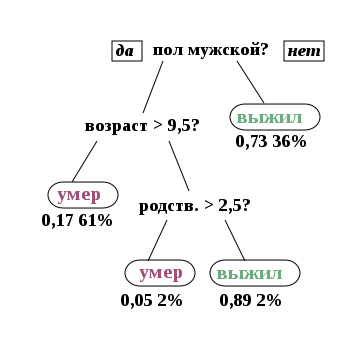


Рис. 1.4 Пример простого дерева решений

Интуитивно качество разбиения зависит от понятия прироста информации, основанного на понятии энтропии.

Определение 1.15 Информационная энтропия – в теории информации мера неопределенности источника сообщений, определяемая вероятностями появления тех или иных символов при их передаче.

Энтропия может интерпретироваться как мера неопределённости (неупорядоченности) некоторой системы, например, какого-либо опыта (испытания), который может иметь разные исходы, а значит, и количество информации. Определяется как:

где – вероятности нахождения системы в *i*-ом состоянии.

Дерево решений можно построить до такой глубины, чтобы в каждом листе был ровно один объект. Но такое дерево будет переобученным – оно слишком настроится на обучающую выборку и будет плохо работать на прогноз на новых данных. Где-то внизу дерева, на большой глубине будут появляться разбиения по менее важным признакам.

Есть два исключения, ситуации, когда деревья строятся до максимальной глубины:

* Случайный лес (композиция многих деревьев) усредняет ответы деревьев, построенных до максимальной глубины
* Стрижка дерева (*pruning*). При таком подходе дерево сначала строится до максимальной глубины, потом постепенно, снизу вверх, некоторые вершины дерева убираются за счет сравнения по качеству дерева с данным разбиением и без него (сравнение проводится с помощью кросс-валидации).

В работе используется дерево решений и случайный лес с подобранными гиперпараметрами.

2 ПРАКТИЧЕСКАЯ ЧАСТЬ

## Язык программирования и библиотеки

Язык программирования:

* Python – высокоуровневый язык программирования, ориентированный на повышенную производительность разработки программного обеспечения и читаемость программного кода. Позволяет оперативно работать с большими объемами данных.

Среда разработки:

* PyCharm – кросс-платформенная среда разработки для языка программирования Python, предоставляющая средства для анализа кода, графический отладчик и инструмент для запуска юнит-тестов.
* Jupyter Notebook – командная оболочка для интерактивных вычислений. Часто используется для работы с данными и машинным обучением.

Методы машинного обучения:

* Skikit-learn – библиотека машинного обучения для языка программирования Python, включающая в себя различные алгоритмы классификации, регрессии и кластеризации и предназначенная для взаимодействия с библиотеками Python NumPy и SciPy.

Работа с данными:

* NumPy – библиотека, предоставляющая реализацию вычислительных алгоритмов, оптимизированных для работы с многомерными массивами.
* Pandas – библиотека для обработки и анализа данных, предоставляющая специальные структуры данных и операции для работы с числовыми таблицами и временными рядами.

Предобработка текстовых данных на естественном языке:

* NLTK – пакет библиотек и программ для символьной и статистической обработки естественного языка, написанных на языке программирования Python.
* Mystem – морфологический анализатор русского языка с поддержкой снятия морфологической неоднозначности. Программа работает на основе словаря и способна формировать морфологические гипотезы о незнакомых словах.
* Gensim – библиотека для обработки естественного языка, предназначенная для «тематического моделирования». С ее помощью можно обрабатывать тексты, работать с векторными моделями слов (Word2Vec, FastText и т.д.) и создавать тематические модели текстов.

## 2.2 Используемый словарь тональности

NRC Word-Emotion Association Lexicon (EmoLex)

Словарь представляет собой список слов на английском языке, где каждое слово помечено одной или несколькими эмоциями из восьми базовых: злость (anger), страх (fear), ожидание (anticipation), доверие (trust), удивление (surprise), грусть (sadness), радость (joy), неудовольствие (disgust), а также двумя оценочными: позитивной (positive), негативной (negative).

Разметка слов была выполнена вручную при помощи краудсорсинга [3]. Подобная градация основана на диаграмме Р. Плутчика [4] (Рис. 2.1).

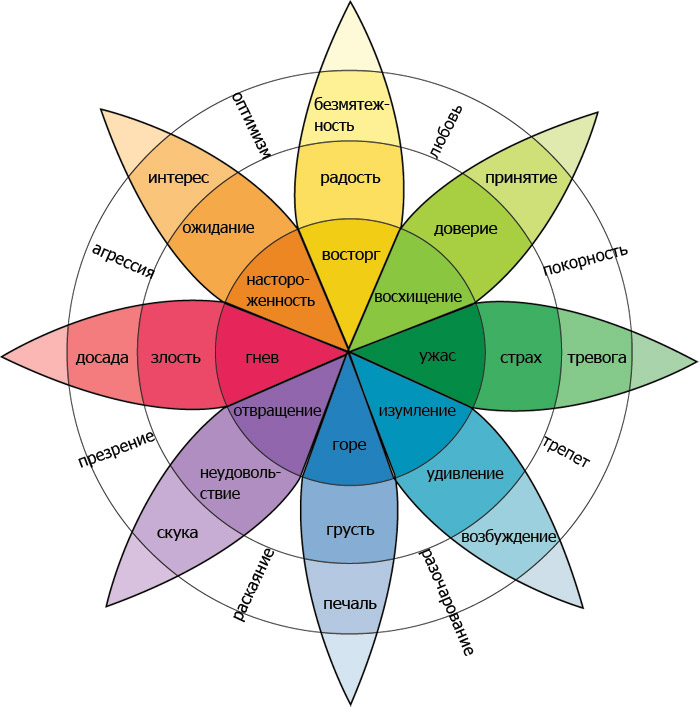


Рис. 2.1 Диаграмма эмоций Р. Плутчика

Оригинальный язык словаря – английский, но он переведен на несколько языков, в том числе и на русский. Перевод был осуществлен с помощью переводчика Google Translate. В связи с этим в работе с таким словарем с русскоязычными текстами могут возникать неточности.

Размер словаря составляет 14182 слова. Соотношение категорий можно увидеть наглядно на Рис. 2.2:

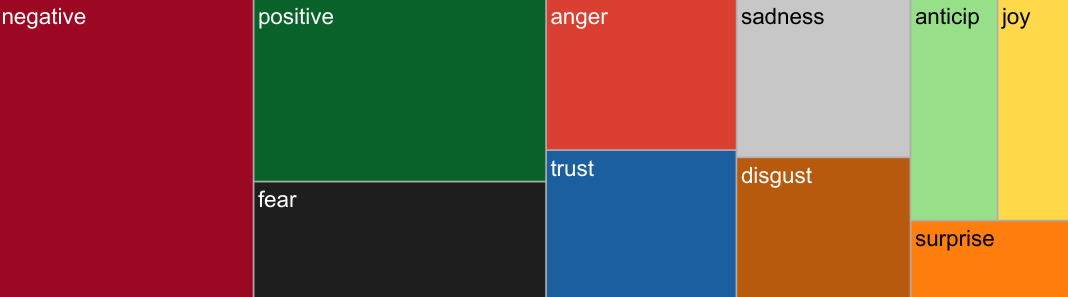


Рис. 2.2 Древовидная диаграмма распределения слов по эмоциям в словаре EmoLex

## 2.3 Используемая модель Word2Vec

В работе для получения векторного представления слов была использована статическая модель *tayga\_none\_fasttextcbow\_300\_10\_2019* [5], которая реализует алгоритм CBOW, описанный в разделе 1.1.1. Модель использует корпус *Тайга* – открытый и структурированный веб-корпус русского языка, снабженный морфологической и синтаксической разметкой.

К настоящему времени около 5 миллиардов слов корпуса составляют 77% литературных текстов (33 литературных журнала), 19% наивной поэзии, 2% новостей (4 популярных сайта) и 2% других (научно-популярные, культурные журналы, социальные сети, любители). стихи и проза) (Рис.. 2.3).

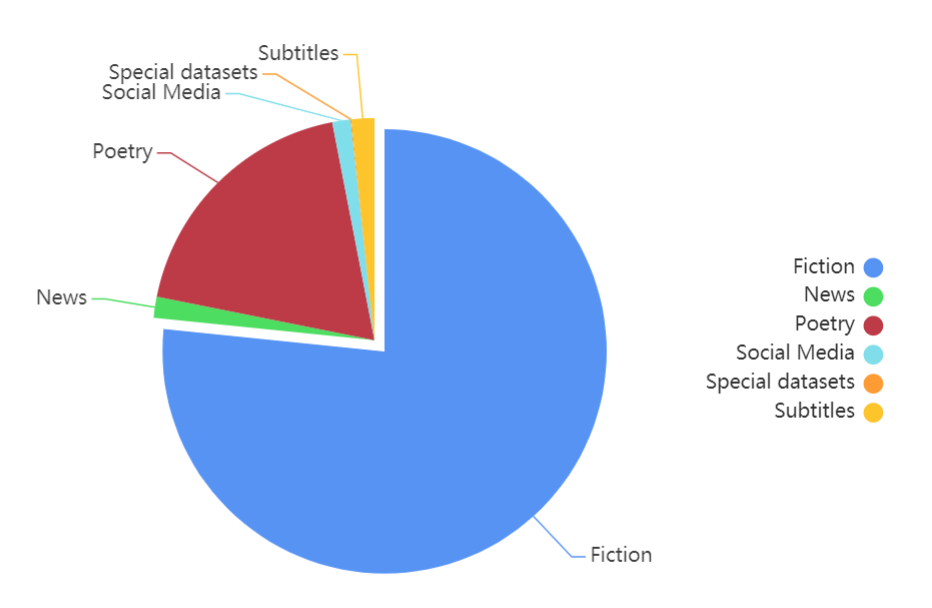


Рис. 2.3 Визуализация наполнения корпуса Тайга

## 2.4 Исследуемые методы классификации

Реализации алгоритмов для получения признаков для классификатора:

* CountVectorizer – sklearn.feature\_extraction.text.CountVectorizer
* TF-IDF Vectorizer – sklearn.feature\_extraction.text.TfidfVectorizer
* Word2Vec – genism.models.Word2Vec

Реализации алгоритмов классификации:

* kNN – sklearn.neighbors.KNeighborsClassifier
* Decision Tree – sklearn.tree.DecisionTreeClassifier
* Random forest – sklearn.ensemble.RandomForestClassifier

## 2.5 Эксперименты

Оценка качества предсказания классификаторов осуществлялась при помощи метрик качества, подробно описанных в разделе 1.1.3

Классификаторы, обученные на векторах встречаемости слов, продемонстрировали очень низкие показатели качества (таблица 2.1):

Таблица 2.1 - Сравнение классификаторов по точности

|  |  |
| --- | --- |
| Метод классификации | Точность |
| kNN | 0.27 |
| Decision Tree | 0.26 |
| Random Forest | 0.29 |

Далее вектора встречаемости слов были подкреплены векторами частоты встречаемости эмоций в предложениях, полученные с помощью эмотивного словаря, описанного в разделе 2.2. Результаты не изменились:

Таблица 2.2 - Сравнение классификаторов по точности

|  |  |
| --- | --- |
| Метод классификации | Точность |
| kNN | 0.28 |
| Decision Tree | 0.27 |
| Random Forest | 0.29 |

Обученные классификаторы демонстрируют качество прогноза, приведенное в таблицах 2.3, 2.4, 2.5.

Таблица 2.3 - Метод k ближайших соседей (kNN)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Класс | Мера точности | Мера полноты | Мера F1 | Носитель меры |
| Грусть | 0.33 | 0.05 | 0.08 | 21 |
| Доверие | 0.00 | 0.00 | 0.00 | 7 |
| Злость | 0.00 | 0.00 | 0.00 | 30 |
| Неудовольствие | 0.00 | 0.00 | 0.00 | 13 |
| Ожидание | 0.00 | 0.00 | 0.00 | 6 |
| Радость | 0.20 | 0.07 | 0.10 | 15 |
| Страх | 0.28 | 1.00 | 0.44 | 40 |
| Удивление | 0.00 | 0.00 | 0.00 | 20 |
| Точность 0.28 | | | | |

Таблица 2.4 - Метод решающих деревьев (Decision Tree)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Класс | Мера точности | Мера полноты | Мера F1 | Носитель меры |
| Грусть | 0.00 | 0.00 | 0.00 | 21 |
| Доверие | 0.00 | 0.00 | 0.00 | 7 |
| Злость | 0.50 | 0.03 | 0.06 | 30 |
| Неудовольствие | 0.00 | 0.00 | 0.00 | 13 |
| Ожидание | 0.00 | 0.00 | 0.00 | 6 |
| Радость | 0.00 | 0.00 | 0.00 | 15 |
| Страх | 0.26 | 0.95 | 0.41 | 40 |
| Удивление | 0.40 | 0.10 | 0.16 | 20 |
| Точность 0.27 | | | | |

Таблица 2.5 - Метод случайный лес (Random Forest)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Класс | Мера точности | Мера полноты | Мера F1 | Носитель меры |
| Грусть | 0.00 | 0.00 | 0.00 | 21 |
| Доверие | 0.00 | 0.00 | 0.00 | 7 |
| Злость | 0.62 | 0.17 | 0.26 | 30 |
| Неудовольствие | 0.00 | 0.00 | 0.00 | 13 |
| Ожидание | 0.00 | 0.00 | 0.00 | 6 |
| Радость | 0.00 | 0.00 | 0.00 | 15 |
| Страх | 0.27 | 0.95 | 0.42 | 40 |
| Удивление | 0.50 | 0.05 | 0.09 | 20 |
| Точность 0.29 | | | | |

Видно, что классы сильно не сбалансированы, так как в произведении «Мастер и Маргарита» преобладают такие эмоции как «злость», «страх» и «удивление» (Рис. 2.4). Ситуацию могла бы исправить добавочная разметка из других произведений, где присутствуют предложения с позитивной тональной окраской, но в связи с ограниченными ресурсами автора работы, дальнейшие результаты получены на имеющейся разметке.

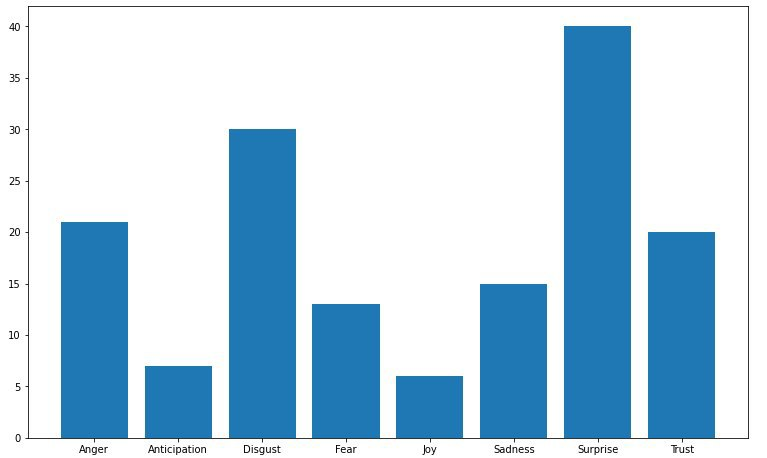


Рис. 2.4 Соотношения классов в отложенной выборке

Классификаторы, обученные на векторах TF-IDF, имеют показатели качества, не сильно отличающиеся от предыдущих (таблица. 2.6):

Таблица 2.6 - Сравнение классификаторов по точности

|  |  |
| --- | --- |
| Метод классификации | Точность |
| kNN | 0.29 |
| Decision Tree | 0.31 |
| Random Forest | 0.31 |

В данном подходе все еще не учитывается контекст встречающихся в предложениях слов. Далее мы меняем модель для получения пространства признаков на Word2Vec (подход описан в разделе 1.1.1). Результаты можно увидеть в таблице 2.7.

Таблица 2.7 - Сравнение классификаторов по точности

|  |  |
| --- | --- |
| Метод классификации | Точность |
| kNN | 0.32 |
| Decision Tree | 0.43 |
| Random Forest | 0.51 |

На данном шаге можно наблюдать улучшение результатов, но не настолько характерное, как ожидалось.

### 2.5.1 Сравнение и оценка результатов

Для классических алгоритмов классификации наилучшей моделью векторного представления слов оказалась Word2Vec, максимальная

точность предсказаний составила 51% (метод случайного леса), минимальная точность составила 32% (метод k ближайших соседей).

Полученные в ходе экспериментов результаты показывают неэффективность эмотивного анализа на уровне предложений с использованием моделей *Bag of Words*, *TF-IDF* и *Word2Vec* для векторного представления документа. Однако такой подход имеет место быть с использованием *BERT* [6] – нейронной сети, разработанной в Google, которая позволяет получать векторное представление для целого предложения, а не одного лишь слова. Также следует отметить, что объем разметки был не достаточно велик для получения результатов высокого качества, а также входные данные не были подкреплены автоматической разметкой.

ЗАКЛЮЧЕНИЕ

В ходе проделанной работы было проведено исследование различных алгоритмов машинного обучения, позволяющих определять эмотивный характер предложения, взятого из художественного произведения.

Сравнительный анализ показал, что модели для векторного представления пространства признаков, не учитывающие контекста встречаемых в предложениях слов, показывают результаты качества ниже, чем модели с учетом контекста.

Согласно результатам исследования, нужно двигаться в сторону анализа объектов большего объема, таких как абзац или целая глава произведения или же поменять модель для векторизации предложений.

Дальнейшим направлением в масштабировании данной работы может стать: разметка нескольких художественных произведений, исследование больших объемов данных с применением искусственных нейронных сетей, расширение эмотивного словаря для переобучения имеющихся моделей. Решения упомянутых задач позволит перейти к задаче шире – распознавания эмотивной составляющей в отношениях между какими-либо двумя героями художественного произведения для последующего построения карты художественного произведения или другого конечного мира.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Батура Т. В. Методы автоматической классификации текстов //Программные продукты и системы. – 2017. – Т. 30. – №. 1
2. Koltsova O. Y., Alexeeva S., Kolcov S. An opinion word lexicon and a training dataset for russian sentiment analysis of social media //Computational Linguistics and Intellectual Technologies: Materials of DIALOGUE 2016 (Moscow). – 2016. – С. 277-287.
3. Mohammad S. M., Turney P. D. Crowdsourcing a word–emotion association lexicon //Computational Intelligence. – 2013. – Т. 29. – №.3. – С. 436-465.
4. Plutchik R., Kellerman H. Emotion, theory, research, and experience. – Academic press, 1980.
5. RusVectores – источник статических моделей для обработки естественного языка [Электронный ресурс]. URL: https://rusvectores.org/ru/models/ (дата обращение: 7.03.2020).
6. BERT – Репозиторий нейронной сети, написанной в Google [Электронный ресурс]. URL: https://github.com/google-research/bert (дата обращение: 17.05.2020).
7. Opara K. R. Grammatical rhymes in Polish poetry: A quantitative analysis //Digital Scholarship in the Humanities. – 2014. – Т. 30. – №. 4. – С. 589-598.
8. GitHub – Репозиторий проекта [Электронный ресурс]. URL: https://github.com/blizliz/sentiment-analysis (дата обращение: 24.05.2020).