МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК
Кафедра программной и системной инженерии

РЕКОМЕНДОВАНО К ЗАЩИТЕ В ГЭК
Заведующий кафедрой
Д.т.н., профессор
_____ *А.Г. Ивашко*
_____2021 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
магистерская диссертация

РАСПОЗНАВАНИЕ СЛУХОВ В РУССКОЯЗЫЧНЫХ СОЦИАЛЬНЫХ СЕТЯХ

09.04.03 Прикладная информатика
Магистерская программа «Информационные системы анализа данных»

Выполнил (а) работу
Студент (ка) 2 курса
магистратуры
очной формы обучения

Ким
Владимир
Аркадьевич

Научный руководитель
кандидат филологических
наук, доцент по кафедры ИС

Бидуля
Юлия
Владимировна

Рецензент
к.ф-м.н., доцент

Ступников
Александр
Анатольевич

г. Тюмень, 2021

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ 2
ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 4
СОКРАЩЕНИЯ И ОБОЗНАЧЕНИЯ 6
ГЛАВА 1. ЛИТЕРАТУРНЫЙ ОБЗОР 7
ГЛАВА 2. МЕТОДОЛОГИЯ РАСПОЗНАВАНИЯ СЛУХОВ 11
2.1 Семантические признаки 11
2.1.1 Матрица весов TF-IDF 11
2.1.2 Понижение размерности 12
2.2 Определение тональности сообщения 14
2.3 Описание набора данных 15
2.4 Дополнительные признаки 16
2.5 Выбор информативных признаков 17
2.6 Выбор метрик оценки модели 19
ГЛАВА 3 ВЫЧИСЛИТЕЛНЫЕ ЭКСПЕРИМЕНТЫ 21
3.1 Используемые методы обучения 21
3.2 Предобработка данных и выделение признаков 24
3.3 Вычислительные эксперименты 25
ЗАКЛЮЧЕНИЕ 31
БИБЛИОГРАФИЧЕСКИЙ СПИСОК 32

ВВЕДЕНИЕ

Повсеместное использование социальных сетей привело к наличию огромного количества данных. В связи с открытостью и доступностью этих данных они быстро распространяются на различных платформах социальных сетях вне зависимости от их достоверности.

Слух – это непроверенная информация, достоверность которой невозможно на момент публикации, но с возможностью определить достоверность позднее, передается от одного человека к другому через социальные сети. Слухи имеют тенденцию процветать в неоднозначных и / или представляющих угрозу ситуациях. Из-за отсутствия информации и недоверия к доступным источникам официальных каналов, люди зачастую ищут информацию по различным непроверенным каналам, и в конце концов, могут распространять утвердительные слухи, основанную на любых доказательствах и рамках понимания. Распространение дезинформации может привести к нежелательным последствиям как для отдельных лиц, так и общества в целом.

Исследованиями по распространению информации в социальных сетях занимаются уже достаточно давно, так, например, работа Свита Фувипадавата и Тсуоши Мурата по детектированию экстренных новостей в твиттере было опубликовано еще в 2010 году. Однако автоматическое обнаружение слухов в социальных сетях и микроблогах остается одной из наиболее востребованных областей исследований в области аналитики социальных сетей и по сей день, исследователи модифицируют алгоритмы, уточняют методы машинного обучения, более углубленно подходят к выбору признаков, решают задачи несбалансированности данных.

Распознавание слуха как правило включает в себя несколько подзадач: определение слуха, отслеживание слуха, отношение, достоверность. В данной мы будем рассматривать задачу определения слуха.

Стоит отметить большая часть исследований по распознаванию слухов использует англоязычные датасеты. Из работ по распознаванию слухов в русскоязычном сегменте можно выделить работу [4].

Цель работы

Исследование методов машинного обучения для распознавания слухов в русскоязычных социальных сетях на основе текстов этих сообщений.

Задачи

1. Выделить семантические признаки из текстов сообщений.

2. Рассмотреть дополнительные лингвистические признаки.

3. Выделить признаки тональности сообщений.

4. Произвести нормализацию и выбрать наиболее информативные признаки.

5. Определить методы машинного обучения.

5. Провести эксперименты по обучению на тренировочной выборке с использованием подбора параметров по сетке и произвести верификацию на тестовой выборке.

6. На основании критериев классификации сделать вывод об наиболее эффективных методах.

Для успешной подготовки и защиты выпускной квалификационной работы обучающимся использовались средства и методы физической культуры и спорта с целью поддержания должного уровня физической подготовленности, обеспечивающую высокую умственную и физической работоспособность. В режим рабочего дня включались различные формы организации занятий физической культурой (физкультпаузы, физкультминутки, занятия избранным видом спорта) с целью профилактики утомления, появления хронических заболеваний и нормализации деятельности различных систем организма.

В рамках подготовки к защите выпускной квалификационной работы автором созданы и поддерживались безопасные условия жизнедеятельности, учитывающие возможность возникновении чрезвычайных ситуаций.

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящей работы используются следующие термины и определения:

Социальная сеть - онлайн-платформа (сайт) для общения, знакомств, создания социальных отношений между людьми.

Слух – это непроверенная информация, достоверность которой невозможно на момент публикации, но с возможностью определить достоверность позднее, передается от одного человека к другому через социальные сети.

Твиттер – социальная сеть публикации коротких сообщений.

Твит (сообщение) – текстовое сообщение в Твиттере, состоящее не более чем из 280 символов.

Ретвит — это повторная публикация твита.

Хэштег - ключевое слово (так называемая «метка»), которое используется в социальных сетях для облегчения поиска сообщений.

Нейронная сеть – в данной работе под понятием нейронной сети подразумевается искусственная нейронная сеть, т.е. математическая модель, а также ее программное воплощение, построенная по принципу биологических нейронных сетей.

Признак – измеримое свойство или характеристика объекта наблюдения, исследования.

Целевой признак (или целевая переменная) - переменная, которая описывает результат процесса.

Вес – числовое значение, отражающее значимость, относительную важность признака.

Матрица весов – представляет собой отношение между объектами и признаками в наборе данных, где объект представляется в виде строки (вектора) с числовыми весами признаков.

Обучающая выборка - выборка из первоначального набора данных, на которой происходит обучение моделей.

Тестовая выборка – выборка из первоначального набора данных, по которой оценивается качество построенной модели.

Датасет - обработанная и структурированная информация в табличном виде.

СОКРАЩЕНИЯ И ОБОЗНАЧЕНИЯ

TF-IDF (term frequency - inverse document frequency) – мера, используемая для оценки важности слова на основе частоты встречаемости слова в документе и низкой частотой появления в других документов, которые составляют исследуемый корпус документов.

t-SNE (t-distributed Stochastic Neighbor Embedding) – стохастическое вложение соседей с распределением Стьюдента.

PCA (Principal Component Analysis) – метод главных компонент

SVD (Singular Value Decomposition) –сингулярное разложение.

LSA (Latent Semantic Analysis) - латентно-семантический анализ.

PMI (Pointwise Mutual Information) – поточечная взаимная информация.

CRF (Conditional Random Field) – условные случайные поля.

LASSO (Least Absolute Shrinkage and Selection Operator) – метод оценивания коэффициентов линейной регрессионной модели.

BAC (Balanced accuracy score) – это макросреднее полноты по классу или, что то же самое, грубая точность, где каждая выборка взвешивается в соответствии с обратной распространенностью ее истинного класса.

MLP (Multi-layer Perceptron) – многослойный персептрон.

SVM (Support Vector Machine) – метод опорных векторов.

RNN (Recurrent neural network) - рекуррентная нейронная сеть.

NB (Naive Bayes) – наивный байесовский классификатор.

SMOTE (Synthetic Minority Oversampling Technique) – метод увеличения числа примеров миноритарного класса.

NA (not available) – нет данных

ГЛАВА 1. ЛИТЕРАТУРНЫЙ ОБЗОР

Одной из значимых работ в области классификации распознавания слухов можно выделить работу Аркаица Зубияга «Распознавание и анализ слухов в социальных сетях» [17]. Авторы работы провели масштабное исследования научных работ на данную тематику с целью построить систему классификации слухов. Работа имеет основательных подход к данной проблеме, начиная с определения самого понятия слуха и заканчивая обзорами различных подходов на каждом этапе распознавания слухов. Представленная архитектура распознавания представляет из себя 4 этапа, представленными на схеме 1.

Схема 1. Архитектура распознавания слуха

Определение слуха – для начала система классификации слухов должна распознать является ли информация слухом. Выходом данного этапа является поток постов, каждый из которых может быть помечен как слух.

Отслеживание слуха – данный компонент отслеживает социальные сети на предмет поиска постов, обсуждающих слухи, удаляет неподходящие посты.

Отношение – определяет каким образом в посте выражено отношение к слуху. Выделено 4 вида отношения к слуху: поддержка, отрицание, сомнение, комментирование

Достоверность – последний компонент определят правдивость или ложность слуха

В публикациях по данным направлениям можно выделить несколько основных подходов к решению указанных выше задач:

1. Извлечение временных и причинно-следственных связей между событиями [13]. Основой данного метода является связывание и сопоставление сообщений, посвященных одному событию, посредством выявления причинных и временных связей, а также построение графа распространения слуха с присвоением соответствующих параметров каждому сообщению;

2. Семантический анализ сообщений. Основой данных методов является разбитие сообщения на составляющие и применение сравнительного анализа сообщений [1]. Так же берется за факт присутствие противоречия между смысловыми данными [15];

3. Методы машинного обучения. В этих работах оценивается вероятность наличия слуха в сообщении путем применения методов классификации. Для данных работ важной особенностью является правильно организованная выборка данных для обучения [11]. Такие методы, как правило, требуют выполнения следующих операций: поиск источников сообщений, семантический анализ сообщений, обработка метаданных, идентификация пользователей, извлечение признаков слуха для обучения.

Для классификации сообщений (слух или не слух, поддержка или отрицание, достоверный или недостоверный) применяются всевозможные методы: от классических вероятностных до различных типов нейронных сетей.

В [3] используется рекуррентная нейронная сеть, на вход которой подаются необработанные параметры, полученные из API социальных сетей и исследуется изменение этих параметров во времени. Предлагаемый подход позволяет находить слухи, но не определять являются ли полученные системой сообщений слухами. Авторы [9] предлагают также применяют RNN, но предлагают обрабатывать весь набор сообщений за определенный временной промежуток.

Ряд работ посвящен обработке лингвистических особенностей текста, а именно поиск «сигнальных» слов, указывающих на возможный слух. Пример

такого подхода отражен в [16]. Авторы [1] развивают эту идею, но используют не только сигналы, но и параметры, описывающими текст в целом, такие как стиль написанного текста, количество символов и другие.

В целом определение слухов осуществляется методами классификации с использованием лингвистических характеристик текста, временных факторов распространения сообщений и особенностей поведения авторов сообщений.

В работе [4] для определения слухов в сообщениях тексты проходили два этапа классификации: по семантическим признакам, а затем по признакам, связанным с характеристиками распространения в социальной сети. Для первого типа использовался метод опорных векторов, для второго – многослойный персептрон.

Имеются работы, посвященные несбалансированности данных. Так, например, в [6] при соотношении сообщений 2000 одного класса против 98000 другого для экспериментов использовался оверсемплинг, который показал улучшение почти по всем классификаторам. Еще один пример применения передискретизации можно увидеть в работе [7], где для повышения точности модели классификации был предложен усовершенствованный алгоритм SMOTE.

Одним из примеров использования понижения векторного пространства признаков полученных является работа [10]. Помимо использования усеченного сингулярного разложения, а авторы также модифицировали метод TF-IDF, который показал улучшение качества работы модели на ~4%.

В недавней работе 2021 года предложена двухэтапная модель распознавания слухов. На первом этапе авторы работы предлагается выявлять аномальных пользователей на основании пользовательских характеристик. На втором, используя лингвистические признаки, признаки на основе психологии и признаки на основе суперсетей обучать второй классификатор для обнаружения слухов в обычной и аномальной группе пользователей.

В таблице 18 представлена информация об основных исследованиях: методах, используемых признаков для обучения, параметров данных и полученных результатов.

Таблица 18. Примеры результатов других исследований

Ссылка	Объем данных	Признаки	Методы	Метрики и результаты			
[1]	2391 слухов	Лингвистические и контент Тональность Тематические Временные	SVM, LIWC	выявлена между достоверностью и каждой из 4 групп лингвистических признаков			
[2]	NA	Лингвистические и контент	LightGBM, Gradient Boosting, SVM, TF-IDF boosted features	NA			
[3]	168 тыс сообщений Weibos	Пользовательские	RNN	Acc	Pre	Rec	F1
				92,4	90,3	87,9	89,1
[4]	NA	Лингвистические и контент Пользовательские	SVM, MLP	Acc	Pre	Rec	F1
				92,7	93,16	89,7	91
[5]	219 тыс сообщений и информация об их учетных записях	Пользовательские Основанные на психологии Лингвистические и контент Признаки суперсетей	NN, Neural Network, SVM, LogReg	Acc	Pre	Rec	F1
				85	68	84	75
[6]	дата сет 1 - 100880 датасет 2 - 101844 твиты	Лингвистические и контент user-based features	CART, SVM, MLP, MaxEnt, Xgboost, SMOTE		Pre	Rec	F1
					95	94,4	94,7
[7]	NA	Лингвистические и контент Пользовательские	Xgboost, SMOTE		Pre	Rec	F1
					82,7	0,837	82,5
[8]	9059 твитов	Лингвистические и контент Пользовательские Тональность Эмотиконы	SVM	Pre		Rec	
				91		91	
[9]	1101 тыс сообщений 992 событий	NA	RNN	Acc	Pre	Rec	F1
				88,1	89	87,5	87,9
[10]	NA	TF-IDF	SVD, SVM	f1-score			
				84			
[12]	информация из профилей 54 мил. пользователей	Лингвистические Пользовательские Временные	SVM, DT, RF, LIWC	Acc	Pre	Rec	F1
				90	93,5	89,2	89,3
[16]	9000	Статистические	SVM, DT, Ngram	Precision			
				~75			

ГЛАВА 2. МЕТОДОЛОГИЯ РАСПОЗНАВАНИЯ СЛУХОВ

2.1 Семантические признаки

С точки зрения машинного обучения распознавание слухов представляет из себя задачу классификации. Традиционно для классификации текстов на естественном языке для построения некоторой числовой модели текстов используют мешок слов, векторные представления слов, n-граммы.

В модели «мешок слов» каждому слову, находящемуся в тексте, сопоставляется число – количество раз, которое данное слово встречается в тексте.

Векторное представление слов – общее название для различных подходов представлений естественного языка, заключавшегося в сопоставлении слову некоторого словаря векторов небольшой размерности.

N-грамма - это непрерывная последовательность n элементов из заданного образца текста. Элементами могут быть фонемы, слоги, буквы, слова или пары оснований в зависимости от приложения. N-граммы обычно собираются из корпуса текста.

Одним из распространенным способом выделения весов признаков из документов является вычисление функции TD-IDF.

2.1.1 Матрица весов TF-IDF

Основная идея вычисления меры TF-IDF является получение большего веса слова с высокой частотой в документе и с низкой частотой появления в других.

Частота TF вычисляется как отношение вхождения слова t в документ к общему числу слов в документе d.

$$TF(t,d) = \frac{n_t}{n_d}$$

IDF (обратная частота документа) призвана уменьшить вес общеупотребительных слов.

$$IDF(t,D) = \log\frac{|D|}{D_t}$$

где $|D|$ – общее количество документов

D_t - число документов, в которых встречается t

В итоге мера TF-IDF является произведением двух множителей:

$$TFIDF(t, d, D) = TF(t, d) \times IDF(t, D)$$

Недостатком такого подхода можно отменить, что значимость слов не учитывает порядок слов в документе и лексической сочетаемости слов.

Из других способов выделения признаков, можно отметить латентно-семантический анализ (LSA), в основе которого лежат принципы факторного анализа, латентных связей изучаемых объектов, поточечная взаимная информация (PMI), использует сортировку списков важных соседних слов двух целевых слов из большого корпуса, условные случайные поля (CRF), вероятностная графическая модель, являющаяся разновидностью марковских случайных полей.

2.1.2 Понижение размерности

Ввиду того, что количество слов в документах во много раз больше, чем количество документов, число столбцов в матрице весов TF-IDF может во много раз превышать количество строк.

Для более эффективной работы классификаторов машинного обучения и снижения эффекта переобучения часто используется сокращение числа столбцов признаков. Переобученная модель хорошо работает на обучающей выборке, но может значительно хуже работать на тестовых экземплярах.

Уменьшение размерности используется в различных приложениях, таких как поиск информации, классификация текста и интеллектуальный анализ данных. Основная цель - уменьшить данные большой размерности до подпространства меньшей размерности, сохранив при этом основные характеристики исходных данных в максимально возможной степени.

2.1.2.1 Анализ главных компонент (PCA)

Метод главных компонент построен на предположении, что чем больше дисперсия рассматриваемого признака, тем он более значим для целевого признака, и представляет собой ортогональное линейное преобразование.

Первую ось обновленной системы координат представляют таким образом, что чтобы дисперсия данных вдоль неё была бы максимальна. Вторая ось проектируется ортогонально первой так, чтобы дисперсия данных вдоль неё, была максимальной их оставшихся возможных. Следящие оси строятся по тому же принципу. При этом первая ось называется первой главной компонентой, вторая — второй и т.д.

Данный способ хорошо подходит для аппроксимации данных в более низкой размерности (как правило двух- или трехмерной).

2.1.2.2 Стохастическое вложение соседей с t-распределением (t-SNE).

Основное преимущество t-SNE заключается в том, что он является нелинейным, в отличии от метода главных компонент. Основной принцип работы t-SNE состоит в том, что он пытается сохранить расстояния между каждым вектором. Сначала t-SNE создаёт распределение вероятностей по парам объектов высокой размерности таким образом, что похожие объекты будут выбраны с большой вероятностью, в то время как вероятность выбора непохожих точек будет мала. Затем t-SNE определяет похожее распределение вероятностей по точкам в пространстве малой размерности и минимизирует расстояние Кульбака — Лейблера между двумя распределениями с учётом положения точек.

Из недостатков метода стоит отметить, что он достаточно медлителен и для своей работы требует огромного количества оперативной памяти

2.1.2.3 Усеченное сингулярное разложение (TruncatedSVD)

Сингулярное разложение использует метод разложения матрицы:

$$A = U\,\boldsymbol{\Sigma}\,V^T$$

где A – матрица $n \times m$ которую мы хотим разложить,

U - матрица $m \times m$,

Σ - диагональная матрица $m \times n$,

V^T транспонирование матрицы $n \times n$.

Усеченное сингулярное разложение представляет из себя низко ранговое приближение для матрицы A. В результате получается матрица с более низким рангом, которая, как говорят, аппроксимирует исходную матрицу.

В работе [20] применяя данное понижение размерности в совокупности модифицированной мерой TF-IDF получилось улучшить точность модели на 3-4%.

2.2 Определение тональности сообщения

Определение тональности – это метод контент-анализа текста в компьютерной лингвистике. Основная задача определения тональности определение эмоциональной окраски текстов, сообщений. Как правило выделяют 3 категории тональности: позитивные (пример: «у меня отличное настроение», «мы посмотрели шикарный фильм», негативные («в магазине была страшная очередь», «шумный сосед мешает спать»), и также выделяют нейтральную тональность, которая не содержит эмоциональной окраски.

Определение эмоциональной окраски сообщений в социальных сетях может служить индикатором определения слуха, так, например, в работе [14] было выяснили, что использование анализа тональности сообщений в твиттере существенно помогает в дифференциации слухов на правдивые и ложные.

В русскоязычном сегменте имеется несколько открытых библиотек для определения тональности русскоязычных текстов.

Для определения тональности сообщения была использована библиотека Библиотека python dostoevsky.

Данная библиотека использует модель fastText, на которая была обучена на самом большом русскоязычном датасете «RuSentiment», представляющий из себя более 30,521 размеченных сообщений, эффективность модели составила: f1 score ~ 0.71.

Библиотека dostoevsky классифицирует тональность сообщения на 5 категорий:

- Негативное настроение;
- Позитивное настроение;

- Нейтральное поведение;

- речевой акт (формальные поздравления, благодарственные и поздравительные посты);

- Класс «пропустить» для неясных случаев.

2.3 Описание набора данных

Для исследования был взят датасет: *2013_Russia_meteor-tweetids_entire_period.csv,* который представляет из набор сообщений из социальной сети твиттер.

В датасет входит 1444 размеченных сообщений о челябинском метеорите, представленных на нескольких языках.

Таблица 2. Количество сообщения на разных языках в датасете.

Язык	Количество	%
Русский	263	18%
Английский	1094	76%
Прочие	87	6%
Всего	1444	100%

Таблица 3. Поля набора данных.

Название поля	Описание
Tweet Text	Текст сообщения
Information Type	Тип информации: затронутые лица, инфраструктура и коммунальные услуги, пожертвования и волонтерство, осторожность и советы, симпатия и поддержка, другая полезная информация, неприменимо.
Information Source	Источник информации: очевидцы, правительство, общественные организации, бизнес, СМИ, посторонние, неприменимо.
Informativeness	Информативность:

	Related and informative (связано и информативно), Related - but not informative (связано - но не информативно), Not related (не связано), Not applicable (не применимо)

В связи с небольшим количеством русскоязычных твитов было принято решение использовать перевод сообщений на английском языке с помощью сервиса translate.google.ru. Таким образом русскоязычная выборка расширилась до 1357 объектов.

Кодирование целевого признака на размеченном массиве произведено в соответствии с таблицей 4. Слухом будем считать сообщения, имеющие информационную ценность и имеющие отношения к событию.

Таблица 4. Кодирование целевого признака

Метка: 1	Related and informative (связано и информативно) Related - but not informative (связано - но не информативно),	1077
Метка: 0	Not related (не связано), Not applicable (не применимо)	280

2.4 Дополнительные признаки

Помимо матрицы весов TF-IDF и тональности сообщений некоторые признаки можно вычленить из текстов сообщений (таблица 4).

Таблица 4. Примеры текстов сообщений

№	Сообщение
1	RT @oleg_kozyrev: Появилось качественное видео падения объекта: http://t.co/cKlrp5en
2	#челябинск после метиоритного дождя http://t.co/HUI662mD
3	RT @tepokohtp: да уж, точно:D #метеорит http://t.co/0zid97d2

Как видно из примеров, помимо стандартных признаков, таких как: количество символов в сообщении, количество специальных знаков, количество заглавных и строчных букв, количество слов в сообщении, можно выделить некоторые признаки, которые как правило предоставляет API социальной сети.

Так, например, ретвит в тексте сообщения обозначается двумя символами «RT» с дальнейшим указанием имени пользователя @username. Помимо этого, можно вычленить отдельные упоминания пользователей, через такую же конструкцию @username без стоящих символов «RT» перед ней.

Отдельным признаком было выбрано наличие нецензурной лексики, которая была осуществлена с помощью корпуса из 624 бранных слов.

Таблица 5. Дополнительные признаки

1	Количество восклицательных знаков в твите	integer
2	Количество вопросительных знаков в твите	integer
3	Количество символов в твите	integer
4	Количество слов в твите	integer
5	Количество заглавных букв в твите	integer
6	Количество строчных букв в твите	integer
7	Количество хэштегов в твите	integer
8	Количество упоминаний в твите (поиск конструкций @username в тексте сообщения регулярным выражением)	integer
9	Наличие ссылок в твите (поиск URL регулярным выражением)	binary
10	Наличие ретвита в сообщении (имеет пометку ‘RT’ в начале сообщения)	binary
11	Наличие ненормативной лексики	binary

2.5 Выбор информативных признаков

Зачастую отдельным этапом в задачах классификации является выбор наиболее информативных признаков. Это позволяет убрать признаки, которые зашумляют данные, избежать переобучения модели, кроме того, большое

количество признаков делает модель громоздкой, трудоемкой и трудной для внедрения в производство.

Как правило переобучение проявляется в том, что модели имеют слишком большие значения параметров. В связи с этим необходимо добавить штраф в целевую функцию. Наиболее используемые виды регуляризации l1 и l2.

l1 регуляризация или LASSO позволяет уменьшить некоторые коэффициенты до нуля. Для некоторого функционала ошибки L мы добавляем сумму модулей параметров (весов).

$$\sum L(y_i, \langle W, x_j \rangle) + \lambda \sum |w_j| \to min$$

где W – вектор параметров,

λ – коэффициент регуляризации

Для l2 регуляризации мы добавляем сумму квадратов $\lambda \sum W^2$, и если l1-регуляризация сводит незначимые веса к нулю, то смысл l2 регуляризации в запрете на на непропорционально большие весовые коэффициенты.

В разрезе рассмотрения признаков, можно использовать LASSO регрессию для обнуления незначимых признаков.

2.6 Кросс валидация

Кросс валидация (или перекрестная проверка) – это метод оценки моделей машинного обучения путем обучения моделей на нескольких подвыборок входных данных.

Различают несколько стратегий разбиения выборки.

Kfold – стратегия подразумевает разбиение на k групп, каждая из которых один раз участвует в тестировании и k-1 в обучении.

StratifiedKFold – очень похож на Kfold, но в данном случае разбиение сохраняет соотношение классов в обучающих и тестовых подвыборках.

ShuffleSplit – позволяет строить так называемые случайные перестановки. Таким образом мы можем получить очень много выборок, при этом мы можем специфицировать размер обучающей выборки, и у нас нет никаких ограничений на то, сколько раз каждый объект должен появиться в обучении или в тесте.

ShuffleSplit также можно стратифицировать используя StratifiedShuffleSplit.

2.6 Выбор метрик оценки модели

Традиционно в задачах классификации используется метрики типа: оценка точности (accuracy), матрица ошибок, точность (precision), полнота (recall), f1-мера.

Accuracy является наиболее простой метрикой, одна показывает отношение точных предсказаний классов общему числу предсказаний. Очевидно, что данная метрика имеет существенный недостаток в случае несбалансированности меток классов.

Рассмотрим матрицу ошибок в таблице 6.

Таблица 6. Матрица ошибок

	Истинная метка класса = 1	Истинная метка класса = 0
Предсказание класса = 1	True Positive (TP)	False Positive (FP)
Предсказание класса = 0	False Negative (FN)	True Negative (TN)

Для оценки качества работы алгоритма на каждом из классов используются метрики точность(precision) и полнота(recall).

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Precision не позволяет нам записывать все объекты в один класс, так как в этом случае мы получаем рост уровня False Positive. Recall демонстрирует способность алгоритма обнаруживать данный класс вообще, а precision способность отличать этот класс от других классов.

Имеется способ объединить precision и recall в одну меру, так, например, среднее гармоническое двух метрик называется F-мерой.

$$F_\beta = (1 + \beta^2) \cdot \frac{precision + recall}{\beta^2 \cdot precision + recall}$$

Соответственно F1-мера это F-мера с β^2=1.

Balanced accuracy score (BAC) – это макросреднее полноты по классу или, что то же самое, грубая точность, где каждая выборка взвешивается в соответствии с обратной распространенностью ее истинного класса.

$$BAC = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right)$$

Индекс Фаулкса-Маллоуза (FMI) определяется как среднее геометрическое для попарной точности и полноты:

$$FMI(Gmean) = \sqrt{\frac{TP}{TP + FP} \cdot \frac{TP}{TP + FN}}$$

Этот метод оценки, который используется для определения сходства между двумя кластерами.

ГЛАВА 3 ВЫЧИСЛИТЕЛНЫЕ ЭКСПЕРИМЕНТЫ

3.1 Используемые методы обучения

3.1.1 Логистическая регрессия (Logistic regression) - представляет собой линейную модель классификации, в которой вероятности, описывающие возможные результаты одного испытания, моделируются с использованием логистической функции. Класс sklearn.linear_model.LogisticRegression реализует регуляризованную логистическую регрессию с использованием библиотеки liblinear, решателей newton-cg, sag, saga и lbfgs. В таблице 7 представлено описание параметров логистической регрессии, используемых для настройки.

Таблица 7. Параметры логистической регресии

Наименование параметра	Описание
'C'	Параметр регуляризации, сила регуляризации обратно пропорциональна C
'penalty'	Штраф на размер коэффициентов
'solver'	Алгоритм используемый для оптимизации ('newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga')

3.1.2 Метод опорных векторов (support vector machine, SVM) – метод машинного обучения, идея которого заключается в построении гиперплоскости, разделяющей объекты выборки оптимальным способом. В работе использовалась реализация sklearn.svm.SVC с различными типами ядра, задающими спрямляющие пространства для нелинейно разделимых выборок: linear, rbf, sigmoid.

Таблица 8. Параметры метода опорных векторов

Наименование параметра	Описание
'C'	Параметр регуляризации, сила регуляризации обратно пропорциональна C
'kernel'	Задает тип ядра, который будет использоваться в алгоритме. 'linear' – линейное ядро

	'rbf' - ядро радиальной базисной функцией 'sigmoid'- сигмоидная функция

3.1.3 Наивный байесовский классификатор (Naive Bayes) - набор алгоритмов обучения с учителем, основанных на применении теоремы Байеса с «наивным» предположением об условной независимости между каждой парой характеристик с учетом значения переменной класса.

В работе использовались методы, реализованные в sklearn:

BernoulliNB() - для данных, которые распределяются согласно многомерному распределению Бернулли.

3.1.4 Деревья решений (Decision Trees) – это непараметрический метод обучения, который предсказывает значение целевой переменной, используя простые правила принятия решений, выведенные из характеристик данных. В работе применялся класс sklearn.tree.DecisionTreeClassifier с расщеплением по индексу Gini.

Таблица 9. Параметры дерева решений

Наименование параметра	Описание
'criterion'	Функция измерения качества разделения. Поддерживает два классификационных критерия: gini, 'entropy'
'max_depth'	Обучающиеся дереву решений могут создавать слишком сложные деревья, которые плохо обобщают данные. Чтобы избежать этой проблемы, необходимы такие механизмы как установка максимальной глубины дерева.

3.1.5 XGboost

XGBoost это оптимизированная распределенная библиотека повышения градиента. В основе лежит алгоритм градиентного бустинга деревьев решений. Градиентный бустинг строит модель предсказания в форме ансамбля слабых предсказывающих моделей. На каждой итерации вычисляются отклонения предсказаний уже обученного ансамбля на обучающей выборке. Следующая модель, которая будет добавлена в ансамбль будет предсказывать эти отклонения. Таким образом, добавив предсказания нового дерева к

предсказаниям обученного ансамбля мы можем уменьшить среднее отклонение модели, которое является целью оптимизационной задачи. Новые деревья добавляются пока ошибка уменьшается, либо пока не выполняется одно из условий "ранней остановки".

Таблица 10. Параметры XGboost

Наименование параметра	Описание
'min_child_weight':	Если на этапе разбиения дерева получается листовой узел с суммой веса экземпляра меньше min_child_weight, то процесс построения откажется от дальнейшего разбиения
'gamma'	Минимальное снижение потерь, необходимое для дальнейшего разбиения на листовом узле дерева. Чем больше гамма, тем более консервативным будет алгоритм

3.1.6 Модель многослойного персептрона (MLP), который обучается методом обратного распространения. Архитектура нейронной сети включает входной, выходной слои и один (по умолчанию) скрытый слой (Рисунок 1).

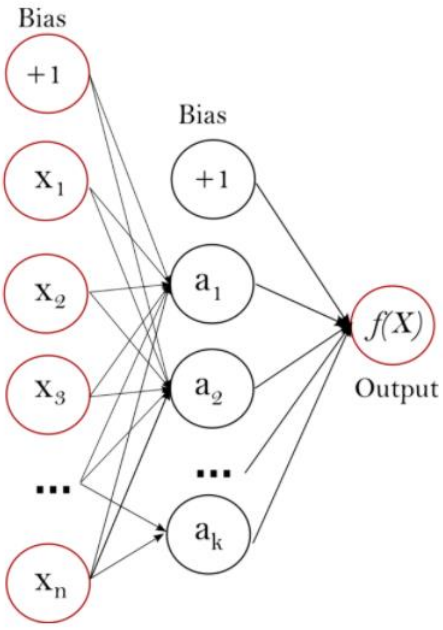


Рисунок 1. Многослойный персептрон

Таблица 11. Параметры многослойного персептрона

Наименование параметра	Описание

'max_iter'	Максимальное количество итераций
'solver'	Алгоритм решения: lbfgs – оптимизатор из семейства квазиньютоновских методов sgd - относится к стохастическому градиентному спуску adam - относится к оптимизатору на основе стохастического градиента, предложенному Кингмой, Дидериком и Джимми Ба
'hidden_layer_sizes'	Количество нейронов в i-м скрытом слое.

3.2 Предобработка данных и выделение признаков

На этапе предварительной обработки данных проведена следующая работ:

1) Выделение признаков (таблица 4);
2) Очистка сообщений от специальных символов, ссылок;
3) Понижение размерности матрицы весов TF-IDF;
4) Нормализация данных.

Для выделения признаков из таблицы 4 под номером использовались стандартные функции python, а также модуль для работы с регулярными выражениями python re.

Дальнейшая обработка данных производилась преимущественно с помощью библиотеки scikit-learn.

Для преобразования текстов сообщений в матрицу TF-IDF был использована функция TfidfTransformer(). В результате обработки получилась матрица размерностью 1357 строк, 4600 столбцов.

Для достижения соотношения размерности количества признаков к количеству объектов не менее чем в 10 раз матрица весов TF-IDF была приведена к размерности (1357,100) с помощью функции TruncatedSVD().

Тональность сообщения получена с использованием библиотеки dostoevsky. Библиотека имеет встроенный токенизатор tokenization.RegexTokenizer() и модель определения тональности models.FastTextSocialNetworkModel().

Функция models.FastTextSocialNetworkModel().predict() с интересующими нас всеми пятью признаками. Каждый из пяти признаков тональности

оценивается числом от 0 до 1, результат работы модели представляет из себя список словарей в формате:

[{' neutral ':0.6926519870758057,' skip ': 0.09535945951938629, ' negative ': 0.09269777685403824, 'positive': 0.0695517510175705, ' speech ': 0.014967083930969238}, ……..,{' neutral ':0.6926519870758057,' skip ': 0.09535945951938629, ' negative ': 0.09269777685403824, 'positive': 0.0695517510175705, ' speech ': 0.014967083930969238}]

Для преобразования списков, массивов, словарей используются библиотеки numpy и pandas. Признаков сохраняются в формате *.csv также с использованием модуля pandas.

В основе многих методов машинного лежит нормальное распределение, поэтому для адекватности работы данных алгоритмов требуется нормализация признаков. За масштабирование признаков по норме отвечает preprocessing.StandardScaler().

3.3 Вычислительные эксперименты

Выбор информативных признаков осуществлен с помощью l1-регуляризации. Для реализации feature_selection.SelectFromModel() в основе которого лежит метод опорных векторов с линейным ядром. В результате регуляризации признаков из 16, были выделены 7 признаков.

Таблица 12. Информативные признаки

1	Тональность	Негативное настроение	да
2		Позитивное настроение	нет
3		Нейтральное поведение	нет
4		речевой акт (формальные поздравления, благодарственные и поздравительные посты);	нет
5		класс «пропустить»	нет
6	Дополнительные признаки	Наличие ссылок в твите	нет
7		Наличие ретвита в сообщении	да
8		Наличие ненормативной лексики	нет
9		Количество восклицательных знаков	нет

10		Количество вопросительных знаков в твите	нет
11		Количество хэштегов в твите	да
12		Количество упоминаний в твите	да
13		Количество символов	нет
14		Количество слов в твите	да
15		Количество заглавных букв в твите	да
16		Количество строчных букв в твите	да

Регуляризация матрицы весов TF-IDF привело к сокращению размерности до (1357,56).

Разделение на обучающую и тестовую выборки производились в соотношении train/test= 75/25 % при помощи функции train_test_split библиотеки sklearn.

Для подбора оптимальных параметров, указанных в 3.1 классификаторов, была использована функция model_selection.GridSearchCV().

Данная функция удобна, тем, что осуществляет поиск по заданным значениям параметров для классификатора с использованием кросс валидации. GridSearchCV применяет стратегию кросс валидации StratifiedKfold.

В таблице 13 приведены параметры методов машинного обучения для поиска по сетке. Ввиду несбалансированности классов для реализации поиска параметров была использована метрика BAC.

Таблица 13. Параметры поиска по сетке

Наименование метода	Параметры поиска	Параметры лучшего оценщика	BAC
LogisticRegression	'C':(1, 0.1, 0.01), 'penalty':('l1','l2','elasticnet'), 'solver':('newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'),	C=1,penalty='l1', solver='liblinear'	0,66
svm.SVC	'kernel':('linear', 'rbf','sigmoid'), 'C':[1, 10]	C=10, kernel='linear'	0,66
BernoulliNB	'alpha' : [0.1, 0.5, 1.0, 2.0, 10.0]	alpha=10	0,70

tree.DecisionTreeClassifier	'criterion':['gini','entropy'], 'max_depth':[4, 5 , 6, 7, 8, 9, 10 ,11 ,12 ,15 ,20 ,30 ,40 , 50, 70, 90, 120, 150]}	criterion='entropy', max_depth=6	0,62
xgboost.XGBClassifier	'min_child_weight': np.arange(1, 5, 0.5).tolist(), 'gamma': [5], 'subsample': np.arange(0.5, 1.0, 0.11).tolist(), 'colsample_bytree': np.arange(0.5, 1.0, 0.11).tolist()	colsample_bytree=1, subsample=1,	0,65

Для модели многослойного персептрона проводились опыты на той же наборе признаков с прогоном различных сочетаний параметров, характеризующих как архитектуру нейронной сети, так и параметры обучения.

Вызов функции GridSearchCV осуществлялся с настройками параметров в следующей последовательности (Таблица 8).

Как видно из таблицы 14 наилучшие показатели точности получены на параметрах: {'activation': 'logistic', 'hidden_layer_sizes': (600,), 'max_iter': 300}.

Таблица 14. Подбор параметров многослойного персептрона

Параметры перебора	Параметры лучшего оценщика	BAC
'max_iter':[200,300,400], 'solver': ['lbfgs', 'sgd', 'adam'], 'hidden_layer_sizes': [(100,)],	{'hidden_layer_sizes': (100,), 'max_iter': 300, 'solver': 'lbfgs'}	0,65
'activation': ['logistic', 'tanh', 'relu'], 'max_iter':[300,], 'hidden_layer_sizes': [(100,)],	{'activation': 'tanh', 'hidden_layer_sizes': (100,), 'max_iter': 300}	0,67
'activation': ['logistic', 'tanh', 'relu'], 'max_iter':[300,], 'hidden_layer_sizes': [(200,), (300,), (600,), (1000,)],	{'activation': 'logistic', 'hidden_layer_sizes': (600,), 'max_iter': 300}	0,68

После того как были найдены оптимальные параметры классификаторов, для полной оценки модели выводились следующие метрики: матрица ошибок (confusion_matrix), Accuracy, Balanced accuracy score (BAC), комплексный отчет

по критериям каждого класса (classification_report), включающий Precision, Recall, F1-score, а также критерии, усредненные по всей тестовой выборке с учетом вклада каждого класса (взвешенное среднее).

Дополнительно для определения влияния информативности признаков было использовано два набора признаков. Первый до использования l1-регуляризации матрица размерностью (1357, 116), состоящую из матрицы весов TF-IDF (1357, 100) и матрицей остальных признаков (1357,16) (результаты приведены в таблице 15). Второй набор признаков представляет из себя выбранные с помощью регуляризации (1357, 63), где TF-IDF (1357, 56), и остальные признаки (1357, 7). Результаты работы алгоритмов на втором наборе признаков представлены в таблице 16.

Таблица 15. Результаты экспериментов до регуляризации

Метод	precision	Recall	f1-score	accuracy	BAC	G-mean
LogisticRegression	**0,77**	**0,80**	**0,78**	**0,80**	0,61	0,78
0	0,51	0,30	0,38			
1	0,84	0,93	0,88			
svm.SVC	0,73	0,76	0,74	0,76	0,57	0,75
0	0,38	0,23	0,29			
1	0,82	0,90	0,86			
BernoulliNB	0,79	0,73	0,75	0,73	0,69	0,68
0	0,40	0,63	0,49			
1	0,89	0,76	0,82			
DecisionTreeClassifier	0,79	0,81	0,79	0,81	0,64	0,78
0	0,54	0,37	0,44			
1	0,85	0,92	0,88			
XGBClassifier	**0,80**	**0,82**	**0,79**	**0,82**	0,62	0,80
0	0,69	0,29	0,40			
1	0,84	0,96	0,89			
MLPClassifier	0,75	0,77	0,76	0,77	0,61	0,75
0	0,43	0,34	0,38			
1	0,84	0,88	0,86			

Как видно из таблицы на первом наборе выделяются логистическая регрессия и XGboost.XGBClassifier. Также заметно, что класс 0 определятся значительно хуже, имея средние значения по метрикам precision, recall и f1-score 0.49, 0.36, 0.40 соответственно, тогда как класс 1 определяется со средними показатели данных метрик 0.85, 0.89, 0.87, что является хорошим результатом.

Данные результаты можно объяснить несбалансированностью классов в исходных данных.

Таблица 16. Результаты экспериментов после l1-регуляризации

Метод	precision	Recall	f1-score	accuracy	BAC	G-mean
LogisticRegression	**0,81**	**0,83**	**0,80**	**0,83**	0,63	0,81
0	0,70	0,30	0,42			
1	0,84	0,97	0,90			
svm.SVC	0,78	0,81	0,78	0,81	0,60	0,80
0	0,56	0,26	0,35			
1	0,83	0,95	0,89			
BernoulliNB	0,79	0,75	0,76	0,75	0,69	0,70
0	0,42	0,60	0,50			
1	0,88	0,79	0,83			
DecisionTreeClassifier	0,76	0,79	0,77	0,79	0,62	0,77
0	0,48	0,33	0,39			
1	0,84	0,91	0,87			
XGBClassifier	0,78	0,81	0,78	0,81	0,60	0,80
0	0,56	0,26	0,36			
1	0,83	0,95	0,89			
MLPClassifier	**0,80**	**0,82**	**0,80**	**0,82**	0,65	0,80
0	0,60	0,36	0,45			
1	0,85	0,94	0,89			

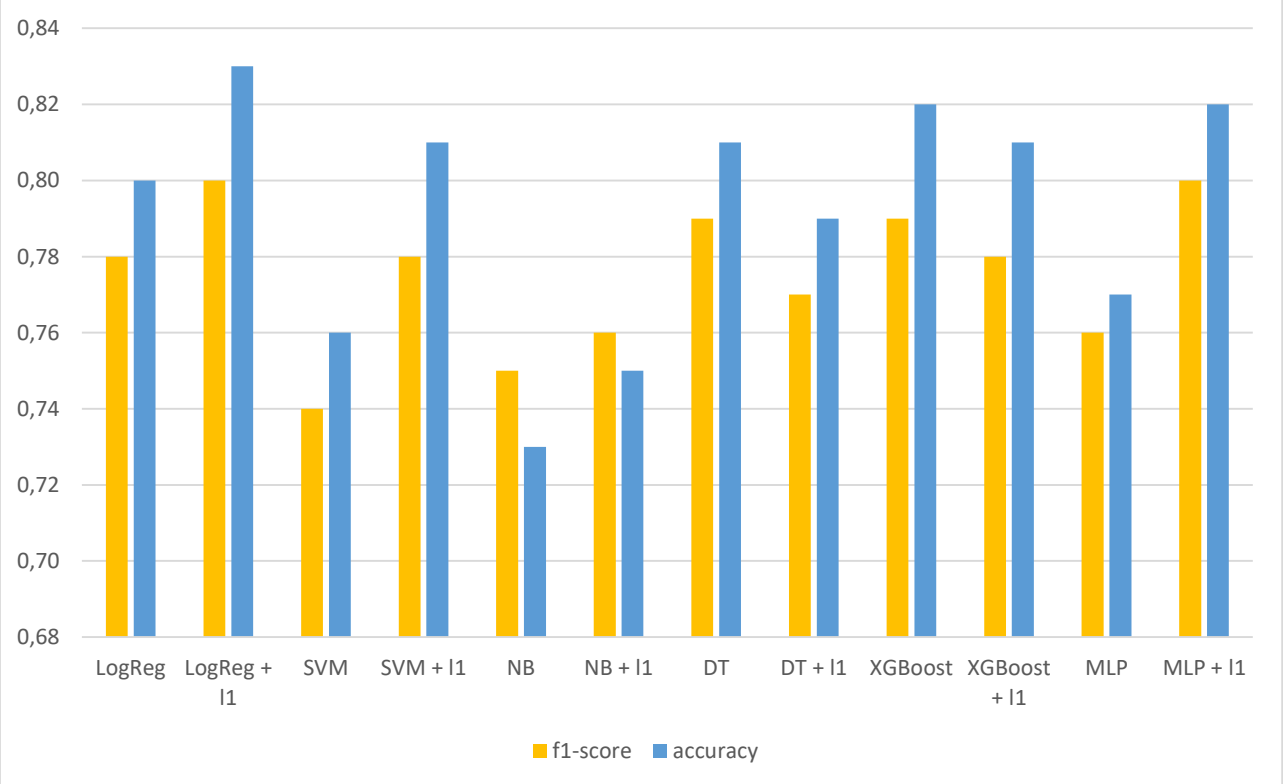
После регуляризации можно выделить результаты работы логистической регрессия с показателями precision=0.81, recall=0.83, f1-score=0.80,

accuracy=0.83, также многослойный персептрон с показателями precision=0.80, recall = 0.82, f1-score=0.80, accuracy=0.82.

Таблица 17. Улучшения результатов на информативных признаках

Метод	precision	recall	f1-score	accuracy	BAC	G-mean
LogisticRegression	0,04	0,03	0,02	0,03	0,02	0,03
smv.SVC	0,05	0,05	0,04	0,05	0,03	0,04
BernoulliNB	0,00	0,02	0,01	0,02	0,00	0,02
DecisionTreeClassifier	-0,03	-0,02	-0,02	-0,02	-0,03	-0,01
XGBClassifier	-0,02	-0,01	-0,01	-0,01	-0,02	0,00
MLPClassifier	0,05	0,05	0,04	0,05	0,04	0,05

График 1. Сравнение результатов до и после l1-регуляризации



Оценивая результата l1-регуляризации признаков (таблица 1, график 1), мы видим существенное улучшение результатов для логистической регрессии, метода опорных векторов и многослойного персептрона на 2-5%. Для дерева решений и XGBoost классификатора мы получили ухудшение в районе 1-2%.

ЗАКЛЮЧЕНИЕ

В ходе проделанной работы был произведен литературных обзор, выделены основные подходы и методы, используемые в аналогичных исследованиях. Для реализации поставленных задач был найден набор данных, содержащих сообщения сети Twitter, посвященных конкретному событию, размеченных экспертами. Объем выборки составил 1357 сообщений. Из текстов сообщений твитов были извлечены несколько групп признаков: матрица весов TF-IDF, тональность сообщения, дополнительные признаки на основе анализа текстов сообщений. Для уменьшения количества столбцов в матрице весов TF-IDF применен метод усеченного сингулярного разложения. Далее был произведён выбор информативных признаков с использованием l1-регуляризатора. На основе 6 наиболее зарекомендовавших себя методов машинного обучения подобраны оптимальные параметры классификаторов и проанализированы полученные показатели оценки моделей.

В результате проделанной работы эффективность определения слуха составила: точность 83% и f1-score 80%. В связи с тем, что извлечение признаков было непосредственно из текстов сообщений, не прибегая к использованию API социальных сетей, примененный подход может быть использован для более широкого класса задач.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Alton Y.K., Snehasish B. Linguistic Predictors of Rumor Veracity / Y.K.Alton, B. Snehasish // Linguistic Predictors of Rumor Veracity on the. Hong Kong. - 2016. - №1. – С. 387-391.

2. Bhattacharjee U., P.K. Srijith, Maunendra Sankar Desarkar. Term Specific TF-IDF Boosting for Detection of Rumours in Social Networks. January 2019

3. Cheng W., Zhanga Y., Tong C., Bu S. Unsupervised rumor detection based on users' behaviors using neural networks. 2017.

4. Chernyaev A., Spryiskov A., Ivashko A., Bidulya Yu. A Rumor Detection in Russian Tweets. Speech and Computer: 22nd International Conference, SPECOM 2020. St. Petersburg, Russia

5. Dong X., Ying Lian, Yuxue Chi, Xianyi Tang & Yijun Liu. A two-step rumor detection model based on the supernetwork theory about Weibo. The Journal of Supercomputing (2021)

6. Ebrahimi Fard A., Mohammadi M., van de Walle B.A. Detecting Rumours in Disasters: An Imbalanced Learning Approach. Delft University of Technology, Delft, The Netherlands 2020

7. Geng Y.; Sui J.; Zhu Q. Rumor Detection of Sina Weibo Based on SDSMOTE and Feature Selection. April 2019 Conference Location: Chengdu, China

8. Hamidian S. and Diab M., Rumor Detection and Classification for Twitter Data. Department of Computer Science The George Washington University 2015

9. Jing M., Wei G., Prasenjit M., Sejeong K., Bernard J.J., Wong K.F., Cha. Detecting Rumors from Microblogs with Recurrent Neural Networks / M. Jing, G. Wei, M. Prasenjit, K. Sejeong, J.J. Bernard, K.F. Wong// Proceedings of the Twen-ty-Fifth International Joint Conference on Artificial Intelligence. - 2016. - С. 3818-3824.

10. Kadhim A. Improving TF-IDF with Singular Value Decomposition (SVD) for Feature Extraction on Twitter. International Engineering Conference 2017. 144-152

11. Kirk M. Thoughtful Machine Learning with Python / M. Kirk // O'Reilly Me-dia. -2017. №1.

12. Kwon S., Cha M., Jung K., Chen W., Wang Y. Prominent Features of Rumor Propagation in Online Social Media. Conference: 2013 IEEE International Conference on Data Mining (ICDM)

13. Paramita M. Extracting Temporal and Causal Relations between Events / M. Paramita // 2014. C. 10-17.

14. Sivasangari V., Mohan A.K., Suthendran K., Sethumadhavan M., Isolating Rumors Using Sentiment Analysis. Journal of Cyber Security and Mobility 2018. 181-200

15. Qazvinian V., Rosengren E., Redev D., Qiaozhu M. Rumor has it: Identifying Misinformation in Microblogs / V. Qazvinian, E. Rosengren, D. Redev, M. Qi-aozhu // Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. Scotland. 2011. C. 1589-1599.

16. Zhao Z., Resnick P., Mei Q. Enquiring Minds: Early Detection of Rumors in Social Media from Enquiry Posts / Z. Zhao, P. Resnick, Q. Mei.// IW3C2. Италия, Флоренция. - 2015. 8 Jin , Cao , Zhang , Luo. News Verification by Exploiting Conflicting / Jin , Cao , Zhang , Luo. // 2016. – C. 2972 – 2978.

17. Zubiaga A., Ahmet Aker A., Bontcheva K., Liakata M., Procter R. 2018. Detection and Resolution of Rumours in Social Media: A Survey. ACM Comput. Surv. 51, 2, Article 32 (February 2018), 36 pages.

18. Батура Т.В. Методы автоматической классификации текстов. Международный журнал Программные продукты и системы. 2017 г. с.85-99

19. Документация библиотеки Scikit-learn