Санкт-Петербургский государственный университет

Математическое обеспечение и администрирование информационных систем
Информационно-аналитические системы

Гарипов Эмиль Ильдарович

Автоматические методы анализа социологических данных

Бакалаврская работа

Научный руководитель:
ст. преп. Ярыгина А. С.

Рецензент:
ст. преп. Сартасов С. Ю.

Санкт-Петербург
2016

SAINT-PETERSBURG STATE UNIVERSITY

Software and Administration of Information Systems
Analytical Information Systems

Emil Garipov

Automated techniques of sociological data analysis

Graduation Thesis

Scientific supervisor:
senior assistant professor Anna Yarygina

Reviewer:
senior assistant professor Stanislav Sartasov

Saint-Petersburg
2016

Оглавление

Введение

За последние несколько лет можно наблюдать большой рост в использовании социальных сетей и платформ для микроблоггинга, которые стали очень популярным инструментом для общения среди пользователей Интернета. Миллионы сообщений появляются каждый день на таких сервисах как Twitter, Facebook, Vk. Авторы таких сообщений пишут об их жизни, делятся своими мнениями на различные темы и обсуждают злободненвные вопросы.

Так как все больше и больше пользователей пишут о продуктах, услугах, которые они используют, или выражают свои политические и религиозные взгляды, сайты для микроблоггинга становятся ценным источником информации о людях. Такие данные могут эффективно использоваться во многих областях, таких как маркетинг, реклама, медицина, психология и социологические исследования.

Обработка текстов вручную с целью извлечения полезной информации требует слишком много времени и человеческих ресурсов. В большинстве случаев данны так много, что это делает данную задачу неосуществимой. Для решения этой проблемы существуют различные автоматические методы анализа текста на естестественных языках, в том числе автоматический анализ тональности текстов.

Основной задачей анализа тональности текстов является извлечение эмоциональной окраски из текстов. Эмоциональная окраска может определяться как "положительная", "отрицательная" или же может принимать значения из некоторого промежутка, например от 0 до N. В данной области проводится все больше и больше исследований, и на данный момент существует много различных подходов, которые уже применяются при решении большого количества практических задач.

С помощью такого анализа текстов компании могут отслеживать то, как потребители относятся к их продукции и предоставляемым услугам. Полученные знания могут быть использованны для дальнейшего анализа и разработки новых маркетинговых стратегий. Также широкое применение можно найти в области социологических исследований. С

помошью анализа эмоциональной окраски текстов можно узнавать о том как люди относятся к тому или иному событию, явлению, предмету. Используя данные из социальных сетей и автоматические методы обработки текстов, возможно получить эффективный и быстрый инструмент для извлечения наиболее релевантной информации об обществе.

Одной из наиболее популярных платформ, где пользователи постоянно делятся своими мнениями, является Twitter. Twitter предоставляет удобный API для быстрого сбора сообщений пользователей - твитов. У твитов существует ряд особенностей, которые могут представлять сложности в использовании традиционных методов анализа текстов. На длину твита существует ограничение - 140 символов, также используется неформальная речь с обилием сокращений, хэштегов, эмотиконов и упоминаний других пользователей.

В данной работе будет рассматриваться задача автоматического анализа тональности русскоязычных текстов, собранных с платформы Twitter. Информация, полученная при применении методов анализа эмоциональной окраски текстов, в дальнейшем может использоваться в социологических исследованиях.

1. Постановка задачи

В рамках данной работы ставились следующие задачи:

- Рассмотреть и изучить основные методы автоматического определения тональности текстов.
- Собрать данные для обучения и тестирования классификаторов.
- Провести предобработку данных.
- Построить модели и оценить их качество работы.

2. Обзор литературы

Методы решения проблемы анализа тональности текстов можно разделить на три основные группы:

- Подходы, основанные на правилах и словарях. [2]
- Машинное обучение с учителем.[3][4]
- Машинное обучение без учителя. [6]

2.1. Подходы, основанные на правилах и словарях

Подходы, основанные на правилах и словарях являются наиболее точными, но требуют значительных трудозатрат и, как правило, очень сильно привязаны к конкретной предметной области.

2.2. Машинное обучение с учителем

Машинное обучение с учителем является наиболее распространенным методом. Его суть состоит в том, чтобы построить модель на коллекции заранее размеченных текстов, а затем использовать ее для анализа новых документов.

В статье [3] основным подходом является применение таких классификаторов, как наивный байесовский классификатор, метод опорных векторов (SVM) и MaxEntropy. Это позволяет добиться точности до 82.9% в определении эмоциональной окраски документов принадлежащим конкретной предметной области. В [4] была показана сильная зависимость точности классификации по тональности от тематики тренировочной и тестовой коллекции. Для оценки модели использовались наборы данных из различных предметных областей: новостные статьи на различные темы и отзывы на фильмы. Результаты тестирования модели, полученной на обучающем множестве из другой предметной области, оказываются гораздо хуже, чем на тестовом множестве той же тематики.

2.3. Машинное обучение без учителя

При использования методов машинного обучения без учителя не требуется заранее размеченных текстов, но данный подход значительно уступает в точности машинному обучению с учителем.

Автор [6] осуществлял классификацию с помощью среднего значения семантической направленности фраз, которые содержат в себе прилагательные и наречия. Нахождение значения семантической направленности базировалось на вычислении взаимной информации [11] фраз со словами "excelent" и "poor". Для вычисления взаимной информации использовались стастистики, полученные поисковой системой. В качестве наборов данных использовались отзывы о ресторанах, ноутбуках и отелях. Средняя точность классификации в данной работе составляет около 74%.

Таким образом, можно сделать вывод о том, что данная задача очень актуальна и существует много подходов для ее решения. Однако у каждого метода есть свои особенности, которые необходимо учитывать при использовании в решении реальных практических задач.

3. Используемые методы

3.1. Постановка задачи классификации

Задача анализа тональности текстов сводится к задаче классификации текстовых документов. В формальном виде ее можно описать следующим образом.

У нас имеется коллекция документов D – множество объектов и фиксированный набор классов C – множество ответов. Также имеется неизвестная функция $y : D \to C$, которая выражает зависимость между множеством объектов и множеством ответов. Требуется по обучающей выборке $\{d_1, d_2, ..., d_n\} \subset D$ для которой известны значения $c_i = y(d_i)$, $i = 1, .., n$ построить фунцию $a : D \to C$, которая будет апроксимировать неизвестную зависимость на всем множестве D.

В решаемой задаче определения тональности множество C состоит из двух классов: "отрицательный" и "положительный", а документы представленны в виде текста на естественном языке. При использовании методов машинного обучения объекты обычно задаются их признаковыми описаниями. В случае с текстовыми документами это может быть представление в виде вектора признаков.

3.2. Классификаторы

Использование метода опорных векторов и наивного байесовского классификатора является одним из самых распространненных подходов для решения задачи определения эмоциональной окраски текстов[7], они также показывают приемлемые результаты. [3] По этим причинам данные классификаторы были выбраны для экпериментов.

3.2.1. Метод опорных векторов (SVM)

Метод опорных векторов - набор схожих алгоритмов обучения с учителем, использующихся для задач классификации и регрессионного анализа.

В задаче классификации по набору тренировочных данных, принадлежащих одному из двух классов, алгоритм, основанный на методе опорных векторов, строит модель, которая по новым входным данным способна определять их класс. Такая модель предполагает представление объектов в виде точек в некотором пространстве размерности **n** и построения разделяющей гиперплоскости размерности **n-1** с максимальным зазором, которая будет разделять объекты из разных классов.

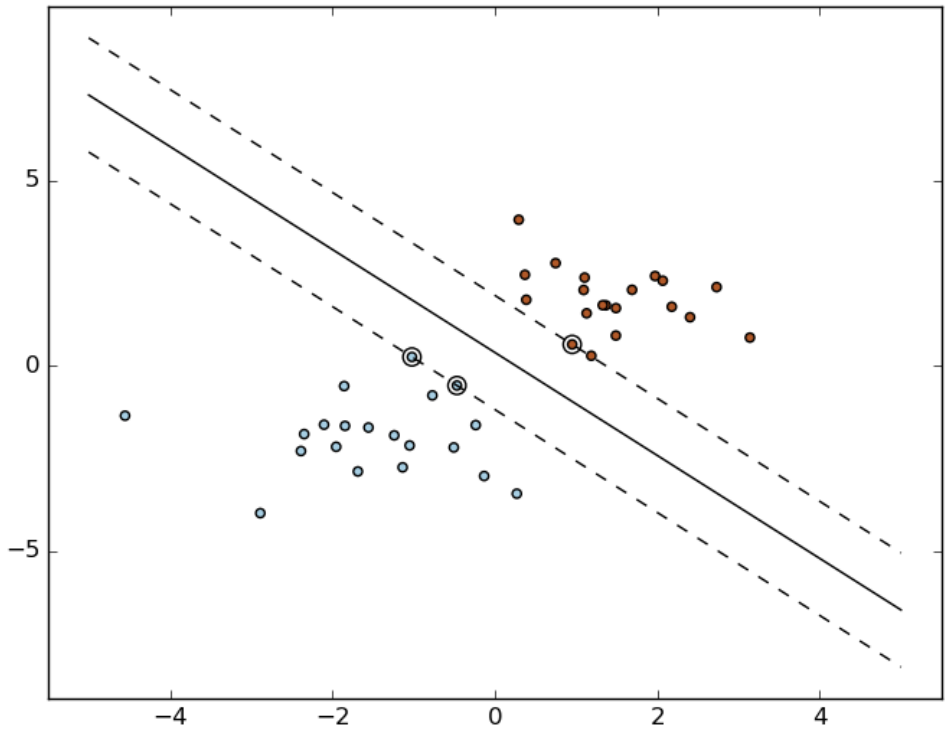

Рис. 1:Пример разделяющей гиперплоскости с максимальным зазором для объектов из двух классов.

Более формально, рассмотрим задачу классификации объектов, заданных признаками в $\mathbb{R}^n$. Будем рассматриватть простой двухклассовый случай, где метки классов принадлежат множеству $\{1, -1\}$. Цель - построить линейный классификатор, то есть такую функцию которая возвращает 1 или -1 и при этом будет зависеть от скалярного произведения вектора признаков x на вектор весов w той же размерности и свободного члена w_0. Рассмотрим функцию:

$$\hat{y} = sign(\langle w, x \rangle - w_0)$$

Знак такой функции будет определять класс объекта. Задача нахождения параметров w и w_0 сводится к оптимизацонной задаче поиска минимума функционала:

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(\langle w, x_i \rangle - w_0)\right)\right] + \frac{1}{2C}||w||^2,$$

где $\frac{1}{2C}||w||^2$ регуляризационное слагаемое, позволяющее избегать проблему переобучения в случае мультиколлинеарности признаков. С помощью константы C можно ослаблять или усиливать регуляризацию.

Задача минимизации функционала может быть эффективно решена с помощью различных численных методов. Например с помощью стохастического градиентного спуска[15].

В нелинейном случае, вместо скалярного произведения можно использовать другие функции, называемые ядром. В данной работе будет использоваться только линейный случай.

3.2.2. Наивный байесовский классификатор

Наивый байесовский метод - множество алгоритмов машинного обучения с учителем, которые основываются на применении Теоремы Байеса с "наивным" предположением о независимости между каждой парой признаков. Для данного класса **y** и зависимого вектора признаков $(x_1, .., x_n)$ Теорема Байеса выражает следующее:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots x_n \mid y)}{P(x_1, \ldots, x_n)}$$

Предположение о независимости признаков можно выразить как:

$$P(x_i|y, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = P(x_i|y),$$

Для всех индексов **i** получаем:

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Поскольку значение $P(x_1, \ldots, x_n)$ постоянно и известно заранее для входных данных, можно использовать следующее правило для классификации:

$$P(y \mid x_1, \ldots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

Помимо модели независимых признаков, наивный байесовский классификатор включает в себя правило решения, по которому будет определяться тот или иной класс для объекта.

$$\hat{y} = \arg\max_{y} P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

Разные наивные байесовские классификаторы отличаются предположениями о распределении $P(x_i \mid y)$. В данной работе используется мультиномиальное распределение, которое параметризуется векторами $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ для каждого класса **y**, где **n** - количество признаков. θ_{yi} - вероятность $P(x_i \mid y)$ того, что признак **i** встретится в примерах из класса **y**. Параметры θ_{yi} оцениваются сглаженной версией функции максимального правдоподобия[9]:

$$\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha n},$$

где $N_{yi} = \sum_{x \in T} x_i$ - количество появления i-го признака в документе класса **y** тренировочного множества **T**, а $N_y = \sum_{i=1}^{|T|} N_{yi}$ - общее количество всех признаков для класса **y**. α - сглаживающий параметр.

Несмотря на достаточно сильные упрощения в предположениях, наивные байесовские классификаторы работают очень хорошо для многих практических задач. Они требуют небольшое количество тренировочных данных для того чтобы вычислить необходимые параметры. Еще одним преимуществом является скорость работы по сравнению с более сложными методами[14].

3.3. Извлечение признаков

В задачах классификации текстов документы задаются их признаковым описанием. Наиболее часто для этого используют модель N-грамм [10]. Документ представляется в виде вектора $d = (w_1, w_2, ..., w_n)$, где n - количество уникальных N-грамм, а w_i - числовой признак для каждой N-граммы, определяемый функцией взвешивания.

В качестве такой функции можно использовать число вхождений в документ соответсвующей N-граммы. Также можно использовать бинарную функцию взвешивания, которая определяется наличием или отсутствием признака: $w_i = 1$, если i-ая N-грамма содержится в документе и $w_i = 0$ в противном случае.

Одним из распростаненных походов является использование TF-IDF [13] в качестве фунции взвешивания. Это показатель, который равен произведению двух чисел: TF (term frequency) и IDF (inverse document frequency).

$$TF - IDF = TF * IDF$$

Значение TF равно отношению числа вхождений слова в документ к общей длине документа. IDF зависит от того, в скольких документах выборки встречается это слово. Чем больше таких документов, тем меньше IDF.

Таким образом, TF-IDF будет иметь высокое значение для тех слов, которые много раз встречаются в данном документе, и редко встречаются в остальных.

Для проведения экпериментов было решено использовать:

- Униграммная модель с функцией взвешивания TF-IDF.
- Униграммная модель с бинарной функцией взвешивания.
- Униграммная и биграммная модель с бинарной функцией взвешивания.

4. Описание набора данных

Для построения классификаторов и их сравнения использовались данные, полученные с платформы микроблоггинга Twitter. Данная платформа была выбрана в связи с её большой популярностью, удобными иструментами для получения информации, а также с важностью для социологических исследований[1].

4.1. Наборы данных

По причине отстутствия в открытом доступе больших, размеченных вручную коллекций твитов, было решено использовать данные, разметка которых была полученна автоматически.

Сбор и разметка TWITTER_DATASET производились автором [16] по принципу, предложенному в [4]. В результате был получен корпус коротких сообщений, состоящий из следующих коллекций:

- 114,911 твитов, размеченных как положительные.
- 111,923 твитов, размеченных как отрицательные.

Для оценки качества полученных классификаторов были найдены тестовые наборы данных, размеченные ассесорами вручную:

- TWITTER_3972 – 3972 твитов различной тематики. 1134 положительных, 1644 нейтральных, 1194 отрицательных
- BANK_TWEETS_TRAIN – 4999 твитов о банках для обучения. 356 положительных, 3421 нейтральных, 1222 отрицательных
- BANK_TWEETS_TEST – 4548 твитов о банках для оценки. 347 положительных, 3534 нейтральных, 687 отрицательных
- TTK_TWEETS_TRAIN – 4999 твитов о телекоммуникационных компаниях для обучения. 920 положительных, 2276 нейтральных, 1803 отрицательных

- TTK_TWEETS_TEST – 3844 размеченных твитов о телекоммуникационных компаниях для оценки. 345 положительных, 2633 нейтральных, 886 отрицательных

Коллекции BANK_TWEETS_TRAIN, BANK_TWEETS_TEST, TTK_TWEETS_TRAIN, TTK_TWEETS_TEST были использованы участниками SentiRuEval_2015 [5] для обучения и оценки их моделей.

4.2. Предобработка данных

Данные существенно отличаются от обычных текстов, используемых в задачах классификации. Для написания твитов пользователи используют неформальную речь, в которой могут присутствовать упоминания других пользователей, ссылки, хэштеги и эмотиконы. Также в Twitter существует строгое ограничение на длинну твита - 140 символов.

Для улучшения качества классификации и уменьшения размерности вектора признаков были произведены следующие действия:

– Удаление стоп-слов.

– Замена всех гиперссылок на ‘LINK’, а всех упоминаний пользователей на ‘USERNAME’.

– Удаление знаков пунктуации.

– Удаление спецсимволов, эмотиконов и эмодзи.

– Удаление хэштегов.

– Все заглавные буквы приведены к строчным.

5. Результаты экпериментов

Для проверки работы методов используются реализации из библиотеки Scikit-learn для языка Python.

5.1. Структура экспериментов

Корпус TWITTER_DATASET, состоящий из автоматически размеченных твитов делился на две части. 90% использовалась для обучения, 10% для тестирования. Далее оценивалось качество классификции на том же наборе данных.

Следующий этап - вычисление метрик качества классификации на тестовых наборах данных BANK_TWEETS_TRAIN, BANK_TWEETS_TEST, TTK_TWEETS_TRAIN и TTK_TWEETS_TEST.

5.2. Методы оценки

Для оценки качества классификации, использовались такие характеристики как: precision, recall [12] и F1 score [8]. Точность (precision) показывает, какая доля объектов попала в тот класс, которому они действительно принадлежат. Полнота (recall) показывает, какая часть объектов принадлежащих классу была выделена при классификации. Они могут быть вычисленны по формулам:

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

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

Точность и полнота не зависят от соотношения размеров классов. Даже если объектов одного класса на порядки меньше, чем объектов другого класса, данные показатели будут корректно отражать качество работы алгоритма.

F1-мера - гармоническое среднее точности и полноты:

$$F = \frac{2 * precision * recall}{precision + recall}$$

Данный показатель можно использовать как критерий качества на основе точности и полноты.

5.3. Корпус автоматически размеченных текстов

Для обучения классификаторов использовались 204,150 автоматические размеченных сообщений. Классификаторы сравнивались на 22,683 тестовых примерах.

В таблице 1 представленны результаты классификации коллекции автоматически размеченных текстов на два класса используя различные способы извлечения признаков с помощью линейного SVM и наивного байесовского классификатора.

Признаки	Количество признаков	Вес	Macro F1 score для NB	Macro F1 score для SVM
Униграммы	29731	TF-IDF	0.73	0.72
Униграммы	29731	Binary	0.74	0.74
Униграммы + биграммы	79481	Binary	0.75	0.75

Таблица 1: Macro F1 score для классификации TWITTER_DATASET используя SVM и наивный байесовский классификатор.

В данном случае наилучший результат получился при использовании униграмм и биграмм с бинарной функцией взвешивания для метода опорных векторов. Однако значение Macro F1 score незначительно

отличается для различных методов классификации и извлечения признаков.

5.4. Классификация тестовых коллекций

В таблице 2 представленны результаты классификации на два класса тестовых размеченных вручную коллекций текстов. В качестве признаков использовались униграммы с бинарной функцией взвешивания.

Набор данных	Macro F1 score для SVM	Macro F1 score для NB
TWITTER_3972	0.69	0.70
BANK_TWEETS_TRAIN	0.31	0.38
BANK_TWEETS_TEST	0.57	0.57
TTK_TRAIN	0.59	0.59
TTK_TEST	0.67	0.68

Таблица 2: Macro F1 score для классификации тестовых наборов данных используя SVM, наивный байесовский классификатор и униграммы с бинарной функцией взвешивания в качестве признаков.

В таблице 3 представленны результаты классификации на два класса тестовых размеченных вручную коллекций текстов. В качестве признаков использовались униграммы и биграммы с бинарной функцией взвешивания.

По получившимся результатам невозможно сделать однозначный вывод в пользу того или иного метода классификации. В некоторых случаях значения macro F1 score практически не отличаются друг от друга и для разных наборов данных качество классификации довольно сильно отличается.

Набор данных	Macro F1 score для SVM	Macro F1 score для NB
TWITTER_3972	0.70	0.70
BANK_TWEETS_TRAIN	0.31	0.36
BANK_TWEETS_TEST	0.59	0.60
TTK_TRAIN	0.58	0.58
TTK_TEST	0.63	0.65

Таблица 3: Macro F1 score для классификации тестовых наборов данных используя SVM, наивный байесовский классификатор и униграммы + биграммы с бинарной функцией взвешивания в качестве признаков.

5.5. Вывод

На автоматически размеченном корпусе твитов модели, построенные по обучающей выборке, достигают качества классификации macro f1 score 0.75. Такой показатель можно считать приемлемым, однако на узкотематических тестовых данных результат в некоторых случаях совсем немного лучше, чем случайный выбор класса.

Заключение

В ходе данной работы были решены поставленные задачи и достигнуты следующие результаты:

- Изучены основные методы для автоматическое определения тональности текстов.
- Собраны данные для обучения и тестирования классификаторов, а также проведена их предобработка.
- Проведено сравнение методов обучения с учителем для решения задачи классификации по тональности текстов на русском языке. Анализировались результаты применения наивный байесовского классификатора, метода опорных векторов при использовании различных способов извлечения признаков.
- Получен вывод о сложности задачи и зависимости результатов от конкретной предметной области.

Список литературы

[1] Big data: methodological challenges and approaches for sociological analysis / Ramine Tinati, Susan Halford, Leslie Carr, Catherine Pope // Sociology. — 2014. — P. 0038038513511561.

[2] Hutto Clayton J, Gilbert Eric. Vader: A parsimonious rule-based model for sentiment analysis of social media text // Eighth International AAAI Conference on Weblogs and Social Media. — 2014.

[3] Pang Bo, Lee Lillian, Vaithyanathan Shivakumar. Thumbs up?: sentiment classification using machine learning techniques // Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10 / Association for Computational Linguistics. — 2002. — P. 79–86.

[4] Read Jonathon. Using emoticons to reduce dependency in machine learning techniques for sentiment classification // Proceedings of the ACL student research workshop / Association for Computational Linguistics. — 2005. — P. 43–48.

[5] SentiRuEval: testing object-oriented sentiment analysis systems in russian / NV Loukachevitch, PD Blinov, EV Kotelnikov et al. // Proceedings of International Conference Dialog. — Vol. 2. — 2015. — P. 12–24.

[6] Turney Peter D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews // Proceedings of the 40th annual meeting on association for computational linguistics / Association for Computational Linguistics. — 2002. — P. 417–424.

[7] Wang Sida, Manning Christopher D. Baselines and bigrams: Simple, good sentiment and topic classification // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2 / Association for Computational Linguistics. — 2012. — P. 90–94.

[8] Wikipedia. F1 score // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/F1_score (online; accessed: 17.04.2016).

[9] Wikipedia. Maximum likelihood // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/Maximum_likelihood (online; accessed: 17.04.2016).

[10] Wikipedia. N-gram // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/N-gram (online; accessed: 17.04.2016).

[11] Wikipedia. Pointwise mutual information // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/Pointwise_mutual_information (online; accessed: 17.04.2016).

[12] Wikipedia. Precision and recall // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/Precision_and_recall (online; accessed: 17.04.2016).

[13] Wikipedia. tf-idf // Википедия, свободная энциклопедия. — 2016. — URL: https://en.wikipedia.org/wiki/Tf-idf (online; accessed: 17.04.2016).

[14] Zhang Harry. The optimality of naive Bayes // AA. — 2004. — Vol. 1, no. 2. — P. 3.

[15] Zhang Tong. Solving large scale linear prediction problems using stochastic gradient descent algorithms // Proceedings of the twenty-first international conference on Machine learning / ACM. — 2004. — P. 116.

[16] Рубцова Ю.В. ПОСТРОЕНИЕ КОРПУСА ТЕКСТОВ ДЛЯ НАСТРОЙКИ ТОНОВОГО КЛАССИФИКАТОРА // Программные продукты и системы. — 2015. — no. 109.