МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Факультет информационных технологий

Кафедра систем информатики

(название кафедры)

Направление подготовки: 230100 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ БАКАЛАВРСКАЯ РАБОТА**

# Разработка и реализация алгоритма тоновой классификации коротких текстов

(тема работы)

Иванова Марина Николаевна

(фамилия, имя, отчество автора - студента –выпускника)

**«К защите допущена»**

Заведующий кафедрой

д.ф.-м.н., профессор

Лаврентьев М.М./…….....

(фамилия, И., О.) / (подпись, МП)

«……»……2014 г.

**Научный руководитель**

к.т.н., зав.лаб. ИСИ СО РАН

Загорулько Ю.А./……

(фамилия , И., О.) / (подпись, МП)

«……»……2014 г.

Дата защиты: «……»……2014 г.

Автор: Иванова М.Н./…….....

(фамилия, И., О.) / (подпись)

Новосибирск, 2014 г.

# Оглавление

Введение ..... 3

1. Постановка задачи ..... 4

2. Предметная область ..... 5

3. Метод построения корпуса текста ..... 6

4. Анализ существующих алгоритмов и методов. ..... 9

4.1. Метод опорных векторов. ..... 9

4.2. Метод k–ближайших соседей (k-Nearest Neighbors, k-NN) ..... 9

4.3. Модель наивного байесовского классификатора ..... 10

5. Построение классификатора ..... 13

5.1. Обучение классификатора ..... 14

5.2. Оценка классификатора (точность, полнота) ..... 15

5.3. Подключение классификатора ..... 16

6. Экспериментальное исследование классификатора ..... 17

6.1. Классификация корпуса текстов полученных из Facebook. ..... 17

6.2. Классификация корпуса текстов полученных из vkontakte. ..... 17

6.3. Результаты ..... 18

Выводы ..... 19

Список литературы ..... 20

# Введение

Всемирная паутина Интернет, так или иначе, затрагивает почти все сферы деятельности людей. Средствам обработки данных в сети все сложнее и сложнее справляться с большим потоком информации, которая уже существует и которая ежедневно добавляется в сеть. Кроме того, данные в Интернет организованы крайне стихийно и несистематично.

В наше время существует очень много систем, позволяющих людям общаться на разные темы и разными способами. Это - микроблоги, форумы. Ежедневно пользователи интернета оставляют тысячи сообщений на разных платформах Twitter, Facebook.

В сообщениях, которые люди пишут друг другу, нет определенных стандартов и шаблонов. Пользователи делятся своими эмоциями, впечатлениями, высказывают свое мнение или отношение к чему-либо. Всю эту информацию можно оценить при помощи оценочных слов, выражений или специальных символов, имеющих положительную, отрицательную или нейтральную окраску. Более того, пользователи в своих высказываниях предпочитают использовать определенные формы частей речи в зависимости от тональности сообщения.

В данной работе проводится исследование различных методов и подходов, для создания тонового классификатора текстовых сообщений полученных из различных социальных сетей.

Данная квалификационная работа посвящена разработке и реализации алгоритма тоновой классификации коротких текстов.

Для решения описанной проблемы была проведена работа:

- Исследована предметная область;
- Получен корпус текстовых сообщений для тренировки тонового классификатора;
- Проведен анализ существующих алгоритмов и методов;
- Выбран и реализован, метод позволяющий решить данную задачу;
- Проведены эксперименты.

# 1. Постановка задачи

Коммерческие компании, политики, лидеры мнений, артисты тратят по несколько часов в день на сбор и сортировку упоминаний о себе или своем продукте в Интернете. Зачастую, это монотонная и рутинная работа, которая является частью процесса формирования положительного имиджа. Задача формирования имиджа состоит из следующих подзадач:

1. Отслеживание новых отзывов;
2. Сортировки отзывов на негативные (надо сразу отреагировать), нейтральные и позитивные (можно ответить позже);
3. Реагирование на комментарий.

Чтобы облегчить задачу мониторинга упоминаний разрабатываются программные комплексы, которые состоят как минимум из двух модулей: модуль сбора сообщений по ключевым словам и модуль классификации сообщений по тональности.

Классификация отзывов по тональности может осуществляться на два класса (положительные, отрицательные), три класса (положительные, отрицательные, нейтральные), пять классов и более.

Целью данной квалификационной работы – является разработка и реализация алгоритма классификации коротких сообщений из социальных сетей на три класса: положительный, отрицательный, нейтральный, а также экспериментальная проверка этого алгоритма на «живых» данных.

Этапы работы:

1. Погружение в предметную область;
2. Сравнительный анализ существующих методов классификации. Выявление их слабых и сильных сторон;
3. Выбор наиболее подходящего метода для классификации коротких текстов и классификации длинных текстов на уровне предложений;
4. Программная реализация и практическая проверка работоспособности и применимости выбранного метода для задачи классификации коротких сообщений на три класса.

Полученный программный модуль можно использовать для определения эмоциональной окрашенности сообщения по отношению к объекту высказывания. Полученные результаты могут быть внедрены в систему мониторинга упоминаний товаров и личностей в Интернете, тем самым повысив ценность этой системы для специалиста.

# 2. Предметная область

В последние годы проводится много исследований в области определения тональности текста. Люди все больше опираются на мнение друг друга. В социальных сетях существуют различные способы выражения эмоций, это «смайлы», «лайки», «палец вверх или палец в низ». Самые распространённые исследования посвящены тоновой классификации отзывов на различные товары, отзывы о фильмах. Иначе говоря, изучалось достаточно большое количество текстовой информации, у которой заранее была определено, к какой предметной области она относится. Все это позволяет получать нужную информацию, опираясь на большинство мнение других.

На сегодняшний день не так много коллекций постов микроблогов на русском языке, было принято решение создать свою коллекцию текстов, с которыми в дальнейшем будет возможно работать. Коллекции состоят из коротких текстовых сообщений, разделенных на три класса «Positive», «Negative», «Neutral». В данной работе будут рассматриваться короткие тексты, полученные из социальной сети Twitter. На основе этих коллекции, будут классифицированы корпуса текстов из других социальных сетей.

# 3. Метод построения корпуса текста

С помощью API twitter, был собран корпус из 3 000 twitter постов. Корпус был разделен на три класса: позитивно окрашенные, негативно окрашенные и нейтральные. Так как люди не ограничены ни форматом, ни формой написания сообщений в микроблогах, нельзя выделить общий паттерн или построить единый словарь для определения эмоциональной принадлежности любого абстрактного сообщения. Поэтому для определения положительно окрашенных и отрицательно окрашенных сообщений использовался подход, предложенный в (Read, 2005 [1]). Для сбора корпуса был выполнен поиск по запросам, характерным для выражения эмоционального отношения к чему-либо. В пользовательских текстах с высокой точностью можно определить эмоцию, если автор указал символ обозначения эмоции на письме (смайлик).

Ключевые слова для поиска и формирования текстовых коллекций:

- положительные: :), =), =)), -), :-), 8-), (:, (=, :D, :-D, :))), )), ))) (более одной скобки), и пр.
- отрицательные: :(, =(, -(, :-(, (( (более одной скобки) и пр
- нейтральные: @lentaruofficial, @afisha_eda, @kommersant, @ura_ru, @snob_project, @ru_slon, @ru_rbc, @bbcrussian, @107_0, @tvrain, @infomoscow24, @rianru, @taygainfo, @EchoMskNews, @Vedomosti, @inosmi, @riabreakingnews, @sibfm b gh

В соответствии с письменным обозначением эмоций был произведен поиск позитивных, негативных и нейтральных сообщений и сформировано три выборки. Эти три выборки в дальнейшем использовались для последующего анализа позитивно, негативно и нейтрально окрашенных твитов [1].

В рамках дипломной работы, мною было написано программное средство, позволяющее, получать короткие текстовые сообщения из социальной сети.

Данное программное средство представляет собой небольшой модуль, снабженный пользовательским интерфейсом (см. рис.1). Процесс работы основан на следующем. Пользователь вручную вводит ключевое слово в поисковую строку – (в данном случае ключевым словом является смайл). Далее происходит подключение к социальной сети Twitter при помощи API Twitter, извлечение из нее сообщений, содержащих заданные

---

[1] Рубцова Ю. Автоматическое построение и анализ корпуса коротких текстов (постов микроблогов) для задачи разработки и тренировки тонового классификатора // Сборник трудов конференции «Инженерия знаний и технологии семантического веба – 2012». – СПб.: НИУ ИТМО, 2012. – С. 109–115.

ключевые слова, и загрузка их в БД данного программного модуля.

В БД короткие тексты хранятся в виде таблицы из трех столбцов:

1. Дата/время/автор;
2. Текстовое сообщение;
3. Класс, к которому относится сообщение.

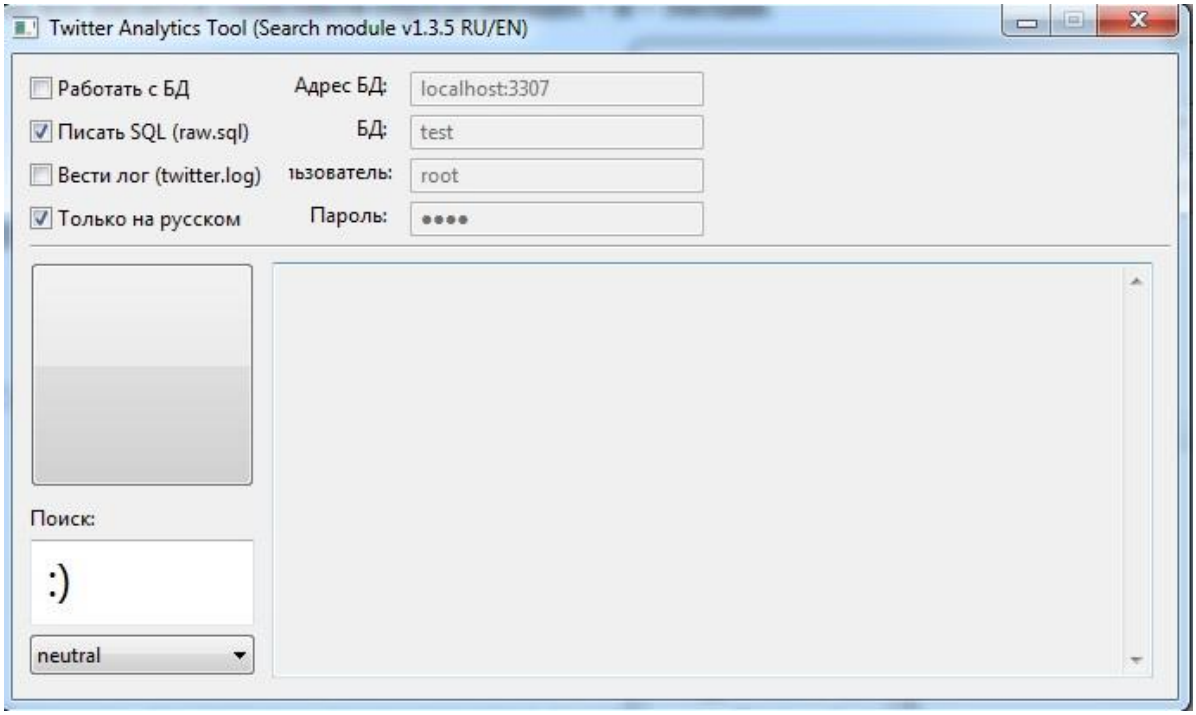


Рис.1. Пользовательский интерфейс модуля сбора коротких текстов.

Подробное описание функций:

1) «Работать с БД» – программа обрабатывает запрос, помещая результат ает в БД.
2) «Писать SQL»-выполняет две функции:
   - Записывает в БД все сообщения без повторений;
   - Удаляет все сообщения, которые были помеченные как эмоция.
3) «twitter.log» – записывает все, что было получено в результате обработки запроса (только сообщения), без помеченной эмоции.

Одинаковые твиты не добавляются в таблицу, т.к. все ее поля являются уникальными полями.

Заметим, что одинаковыми считаются сообщения, идентичные от первого до последнего символа. Так например, сообщения:

«Мама мыла раму» и «Мама мыла раму!» не считаются одинаковыми.

В результате исполнения программы был получен корпус текстов на русском языке автоматически размеченный на три класса в соответствии с тональностью, а именно «Positive», «Negative», «Neutral». Данные корпуса будут использоваться для обучения классификатора.

Следует отметить, что данный способ получения корпусов текста не единственный.

# 4. Анализ существующих алгоритмов и методов

В ходе выполнения работы было исследовано три метода классификации текстов: метод опорных векторов, метод k-ближайших соседей и наивный байесовский классификатор.

## 4.1. Метод опорных векторов

Метод опорных векторов (англ. *SVM, support vector machine*) - набор схожих алгоритмов обучения с учителем, использующихся для задач классификации и регрессионного анализа. Принадлежит к семейству линейных классификаторов. Особым свойством метода опорных векторов, является непрерывное уменьшение эмпирической ошибки классификации и увеличение зазора, поэтому метод также известен как *метод классификатора с максимальным зазором*.

Основная идея метода — перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве. Две параллельных гиперплоскости строятся по обеим сторонам гиперплоскости, разделяющей наши классы. *Разделяющей гиперплоскостью* будет гиперплоскость, максимизирующая расстояние до двух параллельных гиперплоскостей.

Алгоритм работает в предположении, что чем больше разница или расстояние между этими параллельными гиперплоскостями, тем меньше будет средняя ошибка классификатора.

Данный метод было решено не использовать, так было сложно определить параметры по которым будет происходить классификация, а также происходит медленное обучение.

## 4.2. Метод k–ближайших соседей (k-Nearest Neighbors, k-NN)

Метод k-ближайших соседей является одним из самых изученных алгоритмов, используемых при создании автоматических классификаторов. Впервые он был предложен еще в 1952 году[2] для решения задач дискриминантного анализа.

В основе метода лежит очень простая идея: находить в классифицированной коллекции самые похожие на анализируемый текст документы и на основе знаний об их категориальной принадлежности классифицировать неизвестный документ.

---

[2]С. А. Лазарев, Р. В. Шатеев, Построение подсистемы контентной фильтрации интернет-трафика на базе свободного программного обеспечения [Электронный ресурс] / Информационные ресурсы, системы и технологии - Режим доступа: http://irsit.ru/article124, свободный – Яз. рус.

Рассмотрим алгоритм подробнее. При классификации неизвестного документа находится заранее заданное число k текстов из обучающей выборки, которые в пространстве признаков расположены ближе всего. Иными словами находятся k-ближайших соседей. Принадлежность текстов к распознаваемым классам считается известной. Параметр k обычно выбирают от 1 до 100. Близость классифицируемого документа и документа, принадлежащего категории, определяется как косинус угла между их векторами признаков. Чем значение ближе к 1, тем документы больше друг на друга похожи.

Решение об отнесении документа к тому или иному классу принимается на основе анализа информации о принадлежности k его ближайших соседей. Например, коэффициент соответствия рубрики анализируемому документу, можно выяснить путем сложения для этой рубрики значений.

Главной особенностью, выделяющей метод k-NN среди остальных, является отсутствие у этого алгоритма стадии обучения. Иными словами, принадлежность документа рубрикам определяется без построения классифицирующей функции.

Основным преимуществом такого подхода является возможность обновлять обучающую выборку без переобучения классификатора. Это свойство может быть полезно, например, в случаях, когда обучающая коллекция часто пополняется новыми документами, а переобучение занимает слишком много времени.

Классический алгоритм предлагает сравнивать анализируемый документ со всеми документами из обучающей выборки и поэтому главный недостаток метода k-ближайших соседей заключается в длительности времени работы рубрикатора на этапе классификации.

### 4.3. Модель наивного байесовского классификатора

Вероятностная модель для классификатора — это условная модель $p(C|F_1,\dots,F_n)$ над зависимой переменной класса $C$ с малым количеством результатов или *классов*, зависимая от нескольких переменных $F_1 \dots F_n$. Проблема заключается в том, что когда количество свойств $n$ очень велико или когда свойство может принимать большое количество значений, тогда строить такую модель, на вероятностных таблицах становится невозможно. Поэтому мы переформулируем модель, чтобы сделать её легко поддающейся обработке.

Используя теорему Байеса, запишем

$$p(C|F_1,\dots,F_n) = \frac{p(C)\ p(F_1,\dots,F_n|C)}{p(F_1,\dots,F_n)}.$$

На практике интересен лишь числитель этой дроби, так как знаменатель не зависит от $C$ и значения свойств $F_i$ даны, так что знаменатель — константа.

Числитель эквивалентен совместной вероятности модели

$$p(C, F_1, \dots, F_n)$$

которая, может быть переписана следующим образом, используя повторные приложения определений условной вероятности:

$$\begin{aligned}
&p(C, F_1, \dots, F_n) \\
&= p(C)\ p(F_1, \dots, F_n|C) \\
&= p(C)\ p(F_1|C)\ p(F_2, \dots, F_n|C, F_1) \\
&= p(C)\ p(F_1|C)\ p(F_2|C, F_1)\ p(F_3, \dots, F_n|C, F_1, F_2) \\
&= p(C)\ p(F_1|C)\ p(F_2|C, F_1)\ p(F_3|C, F_1, F_2)\ p(F_4, \dots, F_n|C, F_1, F_2, F_3)
\end{aligned}$$

и т. д. Теперь можно использовать «наивные» предположения условной независимости: предположим, что каждое свойство $F_i$ условно независимо от любого другого свойства $F_j$ при $j \neq i$. Это означает:

$$p(F_i|C, F_j) = p(F_i|C)$$

таким образом, совместная модель может быть выражена как:

$$\begin{aligned}
&p(C, F_1, \dots, F_n) = p(C)\ p(F_1|C)\ p(F_2|C)\ p(F_3|C)\ \cdots \\
&= p(C) \prod_{i=1}^{n} p(F_i|C).
\end{aligned}$$

Это означает, что из предположения о независимости, условное распределение по классовой переменной $C$ может быть выражено так:

$$p(C|F_1, \dots, F_n) = \frac{1}{Z} p(C) \prod_{i=1}^{n} p(F_i|C)$$

где $Z$ — это масштабный множитель, зависящий только от $F_1, \dots, F_n$, то есть константа, если значения переменных известны.

Оценка параметров

Все параметры модели могут быть из набора данных обучения. Это оценки максимального правдоподобия вероятностей. Непрерывные свойства, как правило, оцениваются через нормальное распределение. В качестве математического ожидания и дисперсии вычисляются статистики — среднее арифметическое и среднеквадратическое отклонение соответственно.

Если данный класс и значение свойства никогда не встречаются вместе в наборе обучения, тогда оценка, основанная на вероятностях, будет равна нулю. Это проблема, так

как при перемножении нулевая оценка приведет к потере информации о других вероятностях. Поэтому предпочтительно проводить небольшие поправки во все оценки вероятностей так, чтобы никакая вероятность не была строго равна нулю.

Было принято решение использовать данный метод, а именно метод Naive Bayes в построении классификатора.

# 5. Построение классификатора

После изучения известных алгоритмов и методов классификации, было принято решение воспользоваться универсальным методом Naïve Bayes. Наивный Байесовский классификатор — простой вероятностный классификатор, основанный на применении Теоремы Байеса со строгими (наивными) предположениями о независимости.

Во многих практических приложениях, для оценки параметров для наивных Байесовых моделей используют метод максимального правдоподобия; другими словами, можно работать с наивной байесовской моделью, не веря в байесовскую вероятность и не используя байесовские методы.

Как известно, Байесовский классификатор, как и многие другие, обучаем и по этому, перед тем как приступить к реализации алгоритма, следует выделить свод правил по которым, будет обучаться классификатор.

Программная реализация осуществлена при помощи языка программирования Java, с использованием библиотеке Weka. Weka представляет собой набор алгоритмов машинного обучения для задач интеллектуального анализа данных. Алгоритмы могут применяться непосредственно к любой текстовой коллекции. Weka содержит инструменты для предварительной обработки данных, классификации, регрессии, кластеризации, ассоциативных правил и визуализации. Это также хорошо подходит для разработки новых схем машинного обучения.

Ранее полученные корпуса текстов «Positive», «Negative», «Neutral» были объединены в один большой корпус, где текст делился на три класса в соответствии с эмоцией, которая была присвоена каждому текстовому сообщению.

Дальнейший план действий таков:

1) Подготовка тренировочного или обучающего корпуса коротких текстов.

В состав обучающего корпуса входит текстовый файл, содержащий 2.700 коротких предложений, это 2/3 от основного корпуса текстов. Каждому предложения присвоен свой класс тональности, то есть, 900-предложений класса Positive, 900-предложений класса Negative, 900-предложений класса Neutral

2) Подготовка корпуса текста, который будет классифицирован после обучения классификатора.

Корпус текстов, предназначенный для проверки работы классификатора состоит из 1.200 - коротких сообщений(предложений). Это 1/3 от основного корпуса текстов.

Каждому предложению присвоен свой класс тональности. Всего было 400 - предложений класса Positive, 400 - предложений класса Negative, 400 - предложений класса Neutral.

### 5.1. Обучение классификатора

Обучение классификатора проводилось на корпусе текстовых сообщений полученных из twitter. Создано два файла pred.arff и pred2.arff, разница между ними в количестве записей, pred2.arff – корпус, на котором выполнялось обучение классификатора, (составляет 2/3 от всего корпуса), pred.arff – корпус, на котором проверялась работа алгоритма (1/3от всего корпуса текстов).

Пример:

@relation "myrelation"

@attribute txt string

@attribute classatr {positive,negative,neutral}

@data

‘txt’, positive // одно текстовое сообщение принадлежащее классу «positive»

‘txt’, negative

‘txt’, neutral

и.т.д.

После запуска программы классификатор обучился на файле pred2.arff затем определил, на сколько точно классифицирован файл pred.arff.

```
Results
Correctly Classified Instances         749               68.0541 %
Incorrectly Classified Instances       454               31.9459 %
Total Number of Instances              1203
```

Классификатор обучился на 2700 экземплярах коротких сообщений и проверил 1203 сообщений, из которых оказалось 454 сообщения, для которых была не правильно определен тип эмоции. Таким образом, точность классификации составила 68,05%.

## 5.2. Оценка классификатора (точность, полнота)

Точность (precision) и полнота (recall) являются метриками, которые используются при оценке классификации информации. Иногда они используются сами по себе, иногда в качестве базиса для производных метрик, таких как F-мера или R-Precision

Точность системы в пределах класса – это доля документов действительно принадлежащих данному классу относительно всех документов, которые система отнесла к этому классу. Полнота системы – это доля найденных классификатором документов принадлежащих классу относительно всех документов этого класса в тестовой выборке.

Эти значения легко рассчитать на основании таблицы контингентности, которая составляется для каждого класса отдельно.

| Категория i | | Экспертная оценка | |
|---|---|---|---|
| | | Положительная | Отрицательная |
| Оценка системы | Положительная | TP | FP |
| | Отрицательная | FN | TN |

В таблице содержится информация, сколько раз система приняла верное и сколько раз неверное решение по документам заданного класса. А именно:

- *TP* — истино-положительное решение;
- *TN* — истино-отрицательное решение;
- *FP* — ложно-положительное решение;
- *FN* — ложно-отрицательное решение.

Тогда, точность и полнота определяются следующим образом:

*Precision=TP/(TP+FP), Точность= 749/(749+362)=67,42%*

*Recall=TP/(TP+FN), Полнота=749/(749+454)=62,26%*

Вывод, данный классификатор точен на 67,42 %.

## 5.3. Подключение классификатора

Данная работа, сама по себе является частью большой задачи. Существует приложение, которое получает коллекции текстов из различных социальных сетей. Эта коллекция текстов хранится в базе данных. Тексты коллекции не помечены эмоциями и не определена их тональность. Классификатор подключается к данной системе, и пользователь может выбрать любой текстовый корпус из БД или же получить новый, а после определить тональность сообщений.

# 6. Экспериментальное исследование классификатора

Были проведены эксперименты с корпусами текстов полученных из различных социальных сетей. Каждый корпус состоял из около 1,000 сообщений, всего было два корпуса, данные сообщения не имели эмоционального окраса, то есть не было определено, к какому классу относится каждое отдельное сообщение. Все корпуса вручную были размечены на три класса тональности. Это необходимо для оценки классификатора и для его тренировки: точность, полнота.

## 6.1. Классификация корпуса текстов полученных из Facebook

Коллекция текстов из социальной сети Facebook, состояла из 961 сообщения. Для определения правильности работы классификатора, корпус был размечен вручную, на три класса, примерно по равному количеству, то есть 320 сообщениям была присвоена эмоция positive, 320 – negative, 321 – neutral. Далее был обработан классификатором текстов и получен следующий результат.

```
Results
Correctly Classified Instances         307               31.9469 %
Incorrectly Classified Instances       654               68.0531 %
Total Number of Instances              961
```

Результат 31,94% ниже, чем при работе классификатора с текстами полученными из Twitter, это можно объяснить следующим. Текстовые коллекции, полученные из Facebook, отличаются от коллекций Twitter: длиной сообщения, количеством часто встречаемых слов. Все это потому что, в каждой социальной сети преобладают определенные темы общения. На Facebook большая часть тем это политика, новости, то есть сообщения, которые заведомо считаются neutral.

Полнота и точность

Оценка классификатора (точность, полнота)

Точность и полнота.

*Precision=TP/(TP+FP), Полнота = 403/(403+683)=38,71%*

*Recall=TP/(TP+FN), Точность=403/(403+736)=35,38%*

## 6.2. Классификация корпуса текстов полученных из vkontakte

Коллекция текстов из социальной сети Vkontakte, состояла из 920 сообщений. Для определений правильности работы классификатора, корпус был размечен вручную, на три класс, примерно по равному количеству, то есть 300 сообщениям была присвоена эмоция positive, 316 – negative, 304 – neutral. Далее был обработан классификатором текстов и получен следующий результат.

```
Results
Correctly Classified Instances         370               40.2174 %
Incorrectly Classified Instances       550               59.7826 %
Total Number of Instances              920
```

Результат 40,21% ниже, чем при работе классификатора с текстами, полученными из Twitter, но выше чем результат, полученный Facebook, это можно объяснить следующим. Текстовые коллекции, полученные из Facebook, отличаются от коллекций Twitter и Vkontakte. Текст из Vkontakte простые, в основном это статусы, пожелания. Эти сообщения относятся к классам Positive и Negative. Результат классификации оказался ниже, потому что тексты, плаченные из Twitter при помощи инструментария являются короткими, их длина не превышает 20 слов, а тексты взяты для экспериментов имели длину более 50 слов.

Оценка классификатора (точность, полнота)

Точность и полнота.

*Precision=TP/(TP+FP), Полнота = 359/(359+370)=49,24%*

*Recall=TP/(TP+FN), Точность=359/(359+563)=38,93%*

### 6.3. Результаты

После проведения экспериментов были получены результаты классификации текстов полученных из Vkontakte и Facebook. Данные результаты говорят о том, что если брать тексты для обучения и тренировки из одной социальной сети, процент точности будет выше.

При классификации коротких текстов из различных сетей, следует учитывать следующие параметры, при сборе корпусов текстов:

1. Длина сообщения не должна превышать 15-20 слов;
2. Не допускать повторений одного и того же предложения;
3. После сбора корпуса, следует провести фильтрацию на «стоп - слова»

# Выводы

В ходе данной квалификационной работы были выполнены следующие задачи:

- Проведен анализ существующих алгоритмов и методов, а именно: метод опорных векторов, Метод k–ближайших соседей (k-Nearest Neighbors, k-NN), метод Naive Bayes.
- Выбран и реализован.
- Проведены эксперименты.

Анализ существующих методов построения тонового классификатора показал, что, хотя в данном направлении существует достаточно много исследований, но все же одного алгоритма, пригодного для всех приложений, нет. В данной работе рассмотрен один из методов построения тонового классификатора, который по-своему является уникальным.

Практическим результатом данной работы является программа, позволяющая определять тональность коротких текстов, разбивая их на три класса: позитивный, негативный и нейтральны.

В дальнейшем планируется исследование и усовершенствование предложенного алгоритма на других языках.

# Список литературы

1. Read, J. Using Emoticons to Reduce Dependency in Machine Learning Techniques for Sentiment Classification // In: Proceedings of the Student Research Workshop at the 2005 Annual Meeting of the Association for Computational Linguistics. Ann Arbor, Michigan. 2005. - C .43-48.

2. Логашенко И.Б., Современные методы обработки экспериментальных данных [Электронный ресурс] - http://www.inp.nsk.su/chairs/fti/download/Logashenko/Estimators.pdf

3. Четверкин И.И., Лукашевич Н.В. Автоматическая классификация отзывов на основе оценочных слов // 12-я Национальная конференция по искусственному интеллекту с международным участием (КИИ-2010). Москва: Физматлит, 2010.- С. 299–307.

5. Bo Pang and Lillian Lee, Opinion mining and sentiment analysis [Электронный ресурс]- http://books.google.ru/books

6. Peter D. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification Turney // Institute for nformation Technology National Research Council of Canada Ottawa, Ontario, Canada, K1A 0R6 (2002) Canada [Электронный ресурс] - http://aclweb.org/anthology//P/P02/P02-1053.pdf

7. Рубцова Ю. Автоматическое построение и анализ корпуса коротких текстов (постов микроблогов) для задачи разработки и тренировки тонового классификатора // Сборник трудов конференции «Инженерия знаний и технологии семантического веба – 2012». – СПб.: НИУ ИТМО, 2012. – С. 109–115.