

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

Стригулин Кирилл Анатольевич

Выпускная квалификационная работа бакалавра

Анализ тональности высказываний в Twitter

010400

Прикладная математика и информатика

Научный руководитель,
старший преподаватель,
Мозжерина Е. С.

Санкт-Петербург
2016

Содержание

Введение	3
Постановка задачи	4
Глава 1. Обзор предметной области	5
1.1. Интеллектуальный анализ данных	5
1.2. Предварительная обработка данных	9
1.2. Анализ тональности текстов	11
1.3. Некоторые методы машинного обучения	14
Глава 2. Исследования	18
Глава 3. Программная реализация	20
Заключение	23
Список литературы	24

Введение

За последнее десятилетие значительно возросло использование различных онлайн-ресурсов, в частности, социальных сетей, таких как Twitter. Многие компании и организации определяют эти ресурсы как значимые для маркетинговых исследований. Обычно, чтобы получить обратную связь и понимание того, как покупатели относятся к их продукции, компании проводят интервью, анкетирования и опросы. Эти стандартные методы часто требуют больших затрат времени и денег; более того, они не всегда приносят желаемый результат.

Каждый день в Интернет загружается огромное количество данных, содержащих потребительское мнение. Такие данные являются, в основном, неструктурированным текстом, из которого компьютеру сложно извлечь мнение потребителя. В прошлом было невозможно обрабатывать такой большой объём неструктурированных данных, но сегодня это не составляет большого труда. Таким образом обработка естественного языка и анализ тональности играют важнейшую роль в принятии обоснованных решений о маркетинговых стратегиях и дают полезную обратную связь о продуктах и услугах.

Целью данной работы является проведение анализа тональности отзывов в социальной сети Twitter для определения репутации компаний

Постановка задачи

Имеются:

пространство (векторов) наблюдений X , точками которого являются высказывания в Twitter;

пространство ответов Y , точками которого является оценки конкретных высказываний; каждому высказыванию приписана оценка $1, -1, 0$ — положительная, отрицательная, либо нейтральная соответственно;

пространство F классификаторов $f : X \rightarrow Y$

пространство \mathcal{P} распределений (вероятностных мер) на $X \times Y$;

функция штрафа $E : Y \times Y \times X \rightarrow \mathbb{R}$, принимающая значения 0 , либо 1 в зависимости от того, правильно ли отработал классификатор;

обучающий набор $L = (x_1, y_1), \dots, (x_N, y_N)$, состоящий из примеров вида $(x_j, y_j) \in X \times Y$, которые являются значениями независимых случайных величин с одним и тем же неизвестным распределением $\pi \in \mathcal{P}$.

Требуется по X, Y, F, \mathcal{P}, E и L построить классификатор $f \in F$, минимизирующий математическое ожидание штрафа по распределению π

$$\sum_{y \in Y} \sum_{x \in X} E(f(x), y) \cdot \pi(x, y) \rightarrow \min_{f \in F}$$

Глава 1. Обзор предметной области

1.1. Интеллектуальный анализ данных

Интеллектуальный анализ данных (далее ИАД) или как его еще называют «Data mining» — это вычислительный процесс нахождения каких-либо признаков в большом объеме данных. Его методы заимствованы из областей искусственного интеллекта, машинного обучения, компьютерных наук, технологий баз данных и статистики. Целью ИАД является извлечение полезной информации или знаний из какого-либо набора данных и приведение их к понятному виду.

ИАД решает множество задач, основными из которых являются:

- Классификация;
- Кластеризация;
- Ассоциация;
- Регрессия;
- Прогнозирование;
- Последовательность;
- Определение отклонений или выбросов.

Рассмотрим каждую задачу более подробно:

Классификация Является наиболее частой задачей ИАД. По сути своей, классификация представляет собой акт присвоения категории каждому объекту. Любой объект содержит набор признаков, которые характеризуют ту или иную категорию. Анализируя признаки объекта, классификатор определяет, к какой категории его отнести.

Кластеризация Данная задача возникает в том случае, когда данные нужно сгруппировать, т.е. найти естественные группы объектов на основе каких-либо признаков. Объекты, попадающие в одну группу имеют схожие признаки.

Рис. 1 демонстрирует набор данных клиентов, содержащий только два признака: возраст и доход. Алгоритм кластеризации разделил данные на три кластера, основываясь только на этих признаках. В первый кластер попали молодые клиенты с низким доходом, во второй попали кли-

енты среднего возраста с высоким доходом. В третий же кластер попали пожилые люди с относительно низким доходом. Кластеризация — это задача обучения без учителя. Большинство алгоритмов кластеризации строят модель посредством ряда итераций и останавливаются, когда модель сходится, т.е. когда границы сегментов стабилизируются.

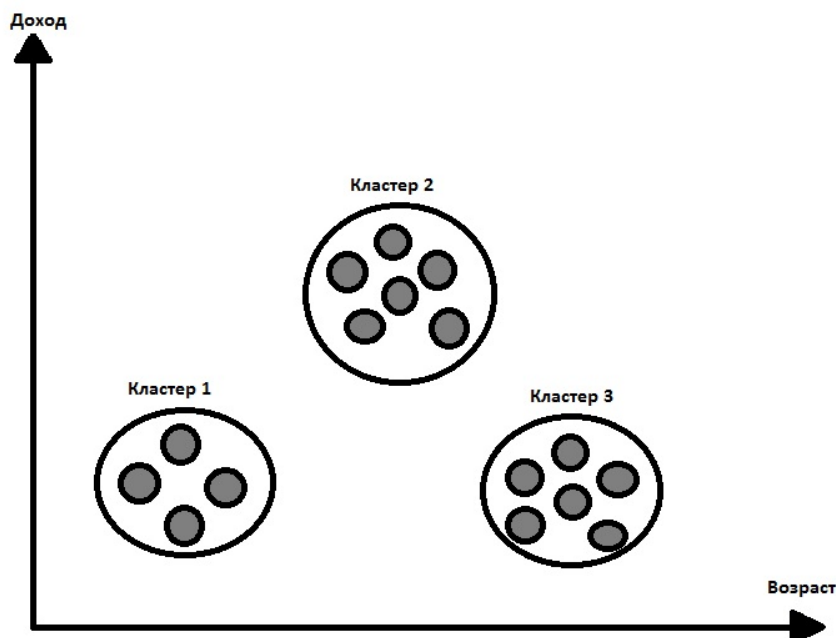


Рис. 1

Ассоциация

Данная задача преследует две цели: нахождение элементов, которые часто появляются вместе и, отталкиваясь от этого, определение ассоциативного правила, по которому это происходит.

Примером данной задачи может являться покупка сопутствующих товаров (*рис. 2*).

Регрессия

Данная задача схожа с задачей классификации, но вместо того, чтобы искать признаки, которые описывают ту или иную категорию, ищутся закономерности, определяющие численное значение (например, возраст, вес, расстояние и т.п).

Прогнозирование Является важной задачей ИАД. Цель прогнозирования — предсказание будущих событий. В качестве входных данных используются последовательность цифр, представляющую собой особенности исторических данных. Опираясь на эти данные, осуществляется прогноз.

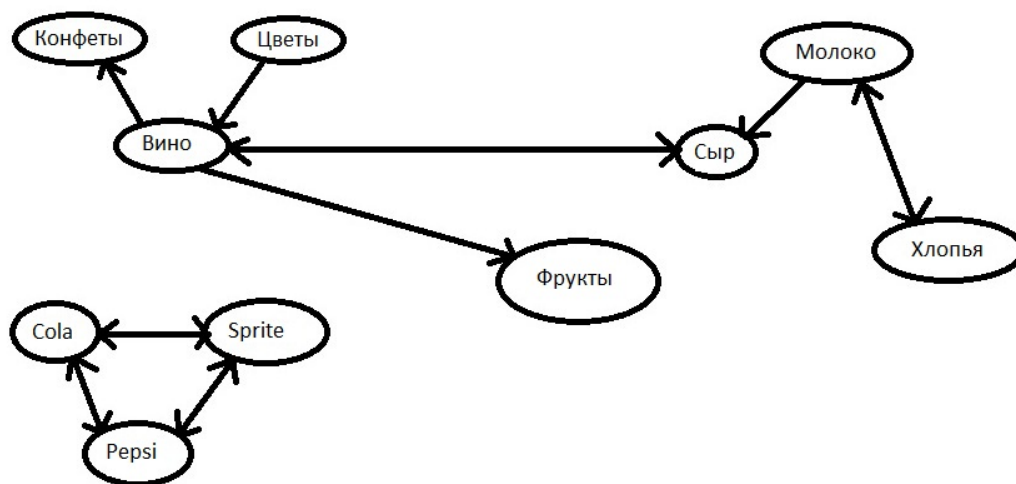


Рис. 2

Примером данной задачи может являться предсказание количества продаж продукта (рис. 3). Здесь сплошной линией показаны фактические данные о продажах, а пунктирной линией - данные, предсказанные на основе прошлый лет.

Последовательность

Данная задача представляет собой поиск закономерностей в цепочке событий, связанных во времени. В каком-то смысле, это обобщение задачи ассоциации, поскольку в данном случае ищется закономерность не между одновременно наступающими событиями, а событиями происходящими во времени.

Так, например, на рис. 4 показано, что при покупке дома в половине случаев в течение месяца приобретается новая кухонная плита, а в пределах двух недель 60% процентов новоселов обзаводятся холодильниками.

Определение отклонений или выбросов

Данная задача определяется как поиск и анализ данных, сильно отличающихся от общего множества данных.

В данной работе будут представлены задачи классификации и регрессии.

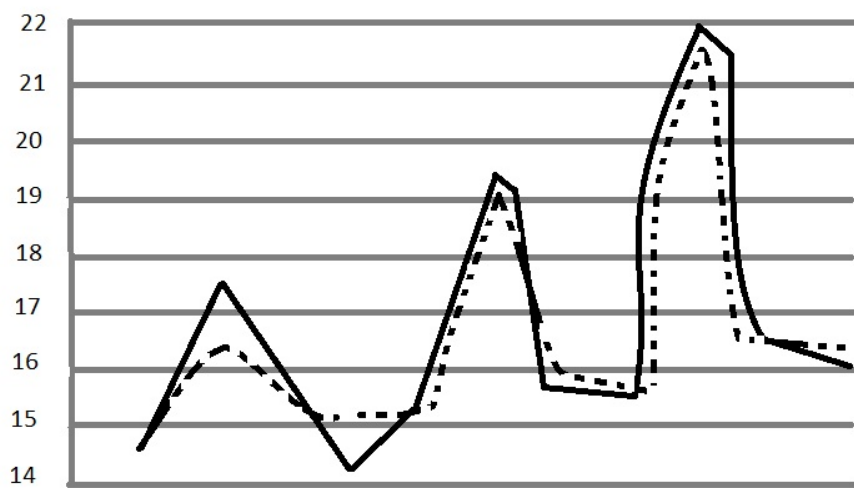


Рис. 3



Рис. 4

1.2. Предварительная обработка данных

Подготовка данных является важной частью ИАД, поскольку анализировать можно любые данные, но для получения качественного анализа, данные должны быть очищены от загрязненных данных. Чтобы правильно подготовить данные, очень важно знать и понимать область исследования, поскольку исследователь должен уметь распознать, какие данные важны, а какие можно исключить, поскольку они не несут весомой значимости. Примерами невесомых данных могут являться пропущенные значения, дубликаты данных, шумы и выбросы.

Примерами техники предварительной обработки данных являются:

- Очистка и преобразование данных;
- Сэмплинг;
- Нормализация;
- Извлечение признаков.

Очистка и преобразование данных

Данный процесс является наиболее ресурсоемким в ИАД. Целью очистки данных является удаление шума и несоответствующей информации из набора данных. Преобразование данных выполняется с целью изменить исходные данные таким образом, чтобы сделать их полезными для ИАД. В целом, очистка данных включает следующие этапы:

- Анализ данных: для того, чтобы определить, какие виды ошибок и несоответствий должны быть удалены
- Определение порядка и правил преобразования данных: в зависимости от числа источников данных, степени их неоднородности и загрязненности, данные могут требовать достаточно обширного преобразования и очистки
- Подтверждение: на этом этапе определяется правильность и эффективность процесса и определений преобразования
- Преобразования: на данном этапе осуществляется выполнение преобразований

- Противоток очищенных данных: после того, как ошибки были удалены, загрязненные данные исходные источников должны быть заменены очищенными

Сэмплинг

Данный процесс отбирает из общего объёма данных выборки, представляющие наибольший интерес для анализа. Такой подход позволяет работать с небольшим объемом данных. Благодаря этому, можно быстро и качественно строить аналитические модели, которые будут показывать высокую точность. Сэмплинг особенно полезен при работе с наборами данных, которые слишком велики, чтобы их можно было эффективно анализировать.

Нормализация

Суть данного метода заключается в приведении данных к одному виду для удобства последующей обработки. Основными методами нормализации являются: *Min – Max*-нормализация, *Z*-нормализация и масштабирование.

Извлечение признаков

Данный процесс нужен для того, чтобы избавиться от лишних данных, которые не повышают, а иногда даже уменьшают точность ИАД. Вообще говоря, извлечение признаков делится на две стадии: отбор признаков и извлечение признаков. На первой стадии в множестве данных ищется подмножество значимых признаков. На второй стадии из подмножества значимых признаков создаются новые признаки. Таким образом уменьшается количество значимых признаков. Данные задачи решаются различными методами, в том числе методом главных компонент, методом независимых компонент, методом жадного добавления, метода поиска в глубину.

1.2. Анализ тональности текстов

Интеллектуальный анализ данных направлен на выявление каких-либо закономерностей в данных. В это же время *анализ тональности текста* направлен на выявления закономерностей в текстах с целью классификации настроения в этом тексте.

В сфере обработки естественного языка существуют различные методы обработки данных (токенизация, токенизация предложения, стемминг).

Существует несколько способов произвести анализ тональности текстов:

- по правилам;
- по словарям;
- машинное обучение;
- обучение без учителя.

Основанный на правилах

Цель такого подхода – поверхностный синтаксический анализ на основе какого-то правила, показывает хорошую точность при большом количестве правил.

Пример такого подхода: если высказывание содержит прилагательные из набора ['хороший', 'добрый', 'новый'] и не содержит прилагательных из набора ['плохой', 'старый', 'ужасный'], то высказывание является положительным.

Основанный на словарях

Данный подход подразумевает составление, либо использование готовых словарей тональности. Словарь представляет собой список слов с приписанной им тональностью. Каждому слову документа ставится в соответствие тональность из словаря, а затем подсчитывается общая тональность документа.

Основанный на машинном обучении

Машинное обучение - это ветвь искусственного интеллекта, направленная на построение моделей, которые могут обучаться на данных, а затем использоваться для анализа. В данной работе используется именно этот подход (обучение с учителем), поэтому ниже он будет изложен подробнее.

Основанный на обучении без учителя

Данный способ не требует обучающих данных, работа идёт с данными, где ответы не известны. Идея данного подхода в том, что чем чаще слова встречаются в одном тексте и чем реже в другом, тем большую значимость они играют для первого. Таким образом, для того, чтобы узнать тональность текста, нужно исходить из тональности таких слов.

Говоря о подходе, основанном на машинном обучении, имеется ввиду обучение по прецедентам (с учителем).

Метод обучения по прецедентам представляет собой алгоритм с обучающими данными; эти данные состоят из обучающих примеров с предполагаемыми для них ответами. Алгоритм обучения по прецедентам использует эти данные, чтобы научиться сопоставлять подаваемые ему примеры с ожидаемыми ответами. Если процесс обучения проходит правильно, то алгоритм может обобщить обучающие данные так, что подаваемые ему на вход новые данные будут правильно сопоставлены с нужными ответами. Метод обучения по прецедентам делится на две категории: классификация и регрессия.

- В случае классификации, целью является присвоение наблюдению класса из конечного множества классов;
- В случае регрессии, целью является присвоение наблюдению вещественного числа.

Для того, чтобы решить задачу обучения по прецедентам, необходимо выполнить следующие шаги:

1. Определить типа обучающих примеров: прежде чем делать что-либо, исследователь должен решить, какого рода данные должны быть использованы в качестве примера. Например, это может быть один символ, слово или целая последовательность слов.
2. Собрать обучающие данные: обучающая выборка должна быть репрезентативной. Таким образом, с помощью экспертов или по каким-то измерениям должен быть найден набор входных значений и соответствующих ответов.
3. Определить представление входных признаков обучаемой функции. Точность обучаемой функции строго зависит от того, как представлены

входные объекты. Как правило, входной объект преобразуется в вектор признаков, который содержит ряд особенностей, описывающих этот объект. Количество признаков не должно быть слишком большим, но должен содержать достаточно информации, чтобы точно предсказать ответ.

4. Определить структуру обучаемой функции и соответствующий алгоритм обучения.
5. Запустить алгоритм обучения на собранных обучающих данных. Некоторые алгоритмы обучения по прецедентам могут потребовать от исследователя настройки некоторых параметров управления. Они могут быть отрегулированы с помощью кросс-валидации.
6. Оценить точность обучаемой функции. После настройки параметров и обучения, точность обучившейся функции должна быть проверена на текстовых данных.

1.3. Некоторые методы машинного обучения

Наивный Байесовский метод

Наивный Байесовский метод является простой вероятностной моделью, которая опирается на предположение о независимости признаков для классификации входных данных. Большую популярность он приобрел благодаря очень простой реализации и относительно высокой точности. Алгоритм предполагает, что признаки не зависят от отсутствия или присутствия других признаков во входных данных, именно поэтому он называется «наивным». В реальности слова в предложении тесно связаны, их позиции и присутствие в предложении сильно влияют на общий смысл и тональность. Несмотря на свою простоту, этот алгоритм часто используется в текстовой классификации и показывает высокую точность при хорошем обучении.

Подход данного алгоритма основан на присвоении документу d класса $c = \arg \max_c P(c | d)$. Согласно правилу Байеса, вероятность принадлежности конкретного документа классу c вычисляется следующим образом

$$P(c | d) = \frac{P(c)P(d | c)}{P(d)},$$

где $P(d)$ не играет роли при выборе класса c , так как не зависит от c . Чтобы оценить вероятность $P(d | c)$, Наивный Байесовский метод раскладывает её на условные независимые величины f_i с учетом документа. Получаем Наивный Байесовский классификатор.

$$P_{NB}(c | d) := \frac{P(c) \left(\prod_{i=1}^m P(f_i | c)^{n_i(d)} \right)}{P(d)},$$

где $P(c)$ и $P(f_i | c)$ вычисляются путём оценки относительной частоты признаков f , которые извлекаются из обучающего корпуса данных; m - количество признаков во всём обучающем корпусе данных; $n_i(d)$ — число встретившихся признаков в документе d .

Проще говоря, алгоритм будет принимать на вход каждый признак (слово) из обучающей выборки и рассчитывать вероятность его попадания в каждый класс (положительный, отрицательный, нейтральный). После завершения подсчёта вероятностей слов-признаков, алгоритм готов к классификации новых данных. Когда будет классифицироваться новое предложение, алгоритм разобьёт его на слова-признаки и будет использовать вероятности отдельных признаков, подсчитанные в процессе обучения. Таким

образом, он найдет объединённую вероятность для предсказания класса данного предложения.

Многие алгоритмы машинного обучения игнорируют признаки, которые имеют слабое влияние на классификацию в целом, основным преимуществом Наивного Байесовского метода является то, что он использует все имеющиеся данные для классификации. Применяя этот подход, во внимание принимаются многие признаки, которые могут оказывать слабое влияние на классификацию по отдельности и огромное в совокупности.

Метод опорных векторов (Support Vector Machine)

Метод опорных векторов очень эффективен в текстовой классификации, как правило, опережая Наивного Байеса. SVM - это невероятный бинарный линейный классификатор. Суть этого метода заключается в том, что он отображает признаки в многомерном пространстве, а затем пытается разделить их на классы гиперплоскостью. Если классы нельзя сразу линейно разделить, то алгоритм будет добавлять новое измерение в попытке дальнейшего разделения. Он будет продолжать этот процесс, пока не будет способен разделить признаки на два отдельных класса. (Рис. 5)

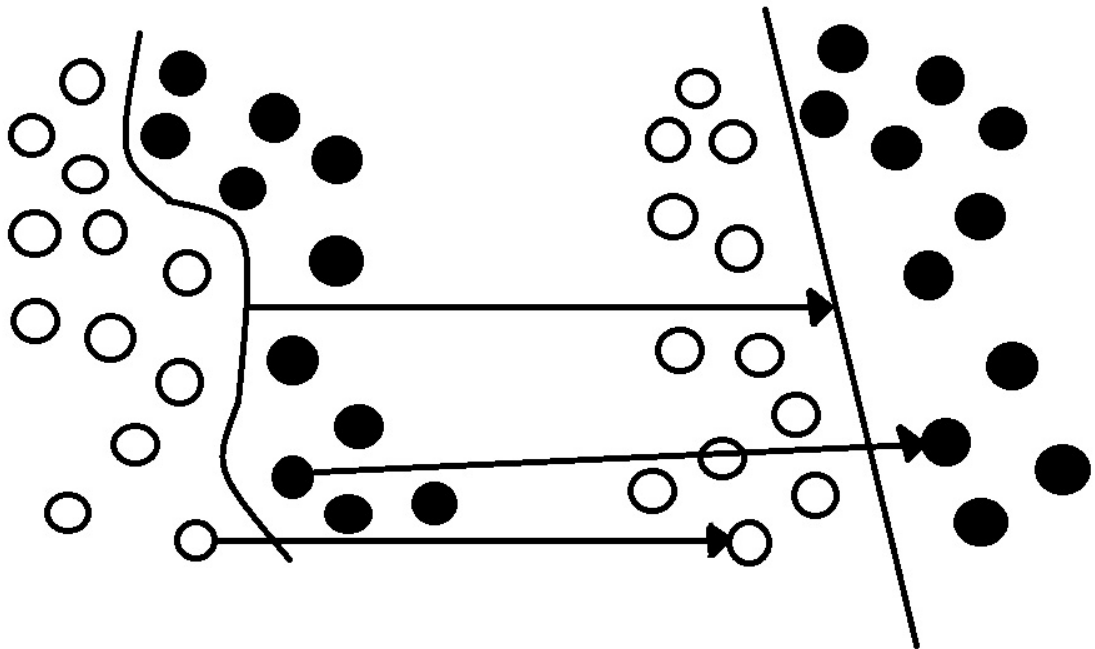


Рис. 5

Одним из основных отличий этого метода от других линейных классификаторов, таких как перцептрон [2], является то, как он выбирает гиперплоскость. В большинстве случаев может быть несколько гиперплоскостей

или даже бесконечное их число. Алгоритм SVM выбирает гиперплоскость, которая обеспечивает максимальное разделение между классами, т.е. гиперплоскость с максимальным зазором, которая снижает верхнюю границу ошибки классификации. Стандартный метод нахождения оптимального способа разделения на классы - это построение двух гиперплоскостей таким образом, чтобы между ними не было признаков. Затем, с помощью этих двух плоскостей, вычисляется итоговая гиперплоскость. Признаки, которые попадают в промежуток между плоскостями называются опорными. (Рис. 6)

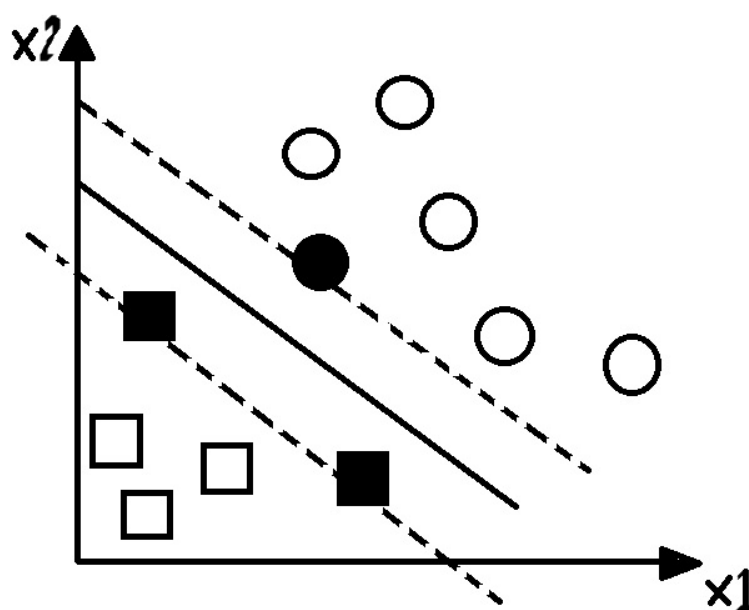


Рис. 6

После того как алгоритм нашел гиперплоскость, которая обеспечивает максимальный уровень разделения между классами, он может приступить к классификации новых данных. Новые экземпляры отображаются в пространстве признаков и классифицируются в зависимости от того, по какую сторону от гиперплоскости они оказываются.

Главной проблемой метода опорных векторов является то, что путем добавления дополнительных измерений, размерность признакового пространства увеличивается в геометрической прогрессии. С технической точки зрения, SVM нейтрализует этот недостаток использованием скалярного произведения в исходном пространстве. Этот метод значительно сокращает обработку, так как все расчёты производятся в исходном пространстве, а

затем отображаются в пространство признаков. С точки зрения классификации это увлечение размерности признакового пространства отрицательно влияет на точность классификации, это известно как эффект Хьюза. Негативное влияние происходит из-за того, что чем больше становится размерность признакового пространства, тем более разреженными становятся обучающие данные в этом пространстве. Для борьбы с этим явлением необходимо при каждом увеличении пространства признаков экспоненциально увеличивать обучающие данные, что не очень применимо на практике.

В заключении, можно сказать, что SVM имеет возможность производить очень высокоточную классификацию на текстовых данных. Главным недостатком является то, что сложность затрудняет четкое понимание, как именно он работает, по сравнению с более простыми алгоритмами.

Глава 2. Исследования

Поскольку в данной работе рассматривается анализ тональности русскоязычных высказываний в Twitter, то целесообразно будет рассмотреть особенности таких высказываний. Работа с русскоязычными твитами представляет некоторые проблемы для обработки естественного языка. Так как твиты ограничены длиной в 140 символов, то обычно они содержат в себе всего одну фразу, либо предложение. Как правило, такие высказывания содержат в себе массу орфографических ошибок, сленга, сокращений, угл-ссылок, хэш-тэгов и обращений типа «@имяпользователя». Поэтому возникает потребность в предварительной обработке данных.

Рассмотрим статистические данные анализа тональности 45000 твитов методами опорных векторов и Наивного Байеса (табл. 1), полученные в работе [3].

Таблица 1

Алгоритм	униграммы	биграммы	триграммы
Наивный Байес	74.56	76.44	75.41
SVM	76.68	77.73	76.97

В работе [3] в качестве оценки точности работы алгоритмов используется следующая метрика:

$$\text{Точность} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (1)$$

TP – истинно-положительное решение;

TN – истинно-отрицательное решение;

FP – ложно-положительное решение;

FN – ложно-отрицательное решение.

Такая оценка (1) является очень простой, в связи с чем можно выделить ее слабые стороны. Она полагает, что все высказывания имеют одинаковый вес, что может не являться таковым, если распределение высказываний в обучающей выборке смещено в сторону какого-то одного или

нескольких классов. Решений у этой проблемы несколько: можно, например, обучать классификатор на специальном сбалансированном корпусе примеров, но за счёт этого, теряется информация об относительной частоте примеров. Другой подход заключается в том, чтобы использоваться для оценки не одну, а две метрики: точность (*precision*) и полноту (*recall*).

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (3)$$

В данном подходе точность показывает, сколько положительных ответов является правильными, а полнота – сколько положительных объектов было определено как правильные. Суть данного подхода заключается в постоянной балансировке между точностью и полнотой, т.к. попытка повысить точность приводит к уменьшению полноты, а увеличение полноты – к падению точности.

В данной работе, для простоты и наглядности, будет использоваться образующая от второго подхода F -мера, которая позволяет гармонизировать между точностью и полнотой, а также дать по ним оценку.

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (4)$$

Возвращаясь к результатам табл. 1, можно заметить, что SVM на всех n -граммах отработал лучше, чем Наивный Байес. Это может быть обусловлено как мощной теоретической основой метода опорных векторов, так и тем, что Байесовский подход считает признаки независимыми. Также стоит отметить, что наилучшие свои результаты методы показали при использовании модели биграмм.

Исходя из статистических и теоретических данных, можно предположить, что наиболее подходящими для задачи анализа тональности высказываний в Twitter является метод опорных векторов. Проверим данное предположение, проведя численный эксперимент на реальных данных.

Глава 3. Программная реализация

Для проведения экспериментов был реализован комплекс программ, позволяющих произвести анализ тональности высказываний в Twitter.

Общая схема работы программы представленная на рис.6

Корпус русскоязычных твитов был взят из конференции «Диалог-2016» [3] Корпус состоит из 8643 обучающих твитов и 19773 – тестовым. Твиты содержат в себе информацию об операторах сотовой связи в России.

Таблица 2

Корпус	положительные	отрицательные	нейтральные
Диалог	3326	3783	1534

На первом шаге происходит выгрузка твитов из базы данных и предварительная их обработка. Предобработка состоит из:

1. удаления знаков пунктуации
2. удаления цифр
3. удаления лишних пробелов
4. удаления ссылок
5. удаления хэштегов
6. удаления окончаний
7. удаления прочих символов
8. приведению всех слов к нижнему регистру

Таким образом «@МТС, подключите мне наконец вот эту опцию <http://t.co/fSRrIzOHQA> #МТС» превращается в «мтс подключит мне наконец вот эт опци мтс».

На втором шаге используется представление высказываний в качестве n-грамм, т.е. в качестве последовательности n-слов. В данной эксперименте рассматривается применение униграмм, двуграмм и триграмм. Все все слова высказываний объединяются по одному (в случае биграмм –

по два, триграм – по три) в один большой словарь. После этого каждый твит принимает вид бинарного вектора чисел размерности нашего словаря. В зависимости от того, встречается ли данное слово словаря в твите ставится 0, либо 1.

После этого, для уменьшения размерности матрицы признаков, используется пакет математических методов `numpy.linalg.svd` [ссылка], который производит сингулярного разложения матрицы, тем самым уменьшая размерность.

На третьем шаге происходит классификация высказываний посредством Наивного Байесовского метода и метода опорных векторов без ядра. В качестве оценки алгоритмов использована F_1 –мера

Таблица 3

Алгоритм	униграммы	биграммы	триграммы
Наивный Байес	0,36	0,475	0,37
SVM	0,42	0,53	0,44

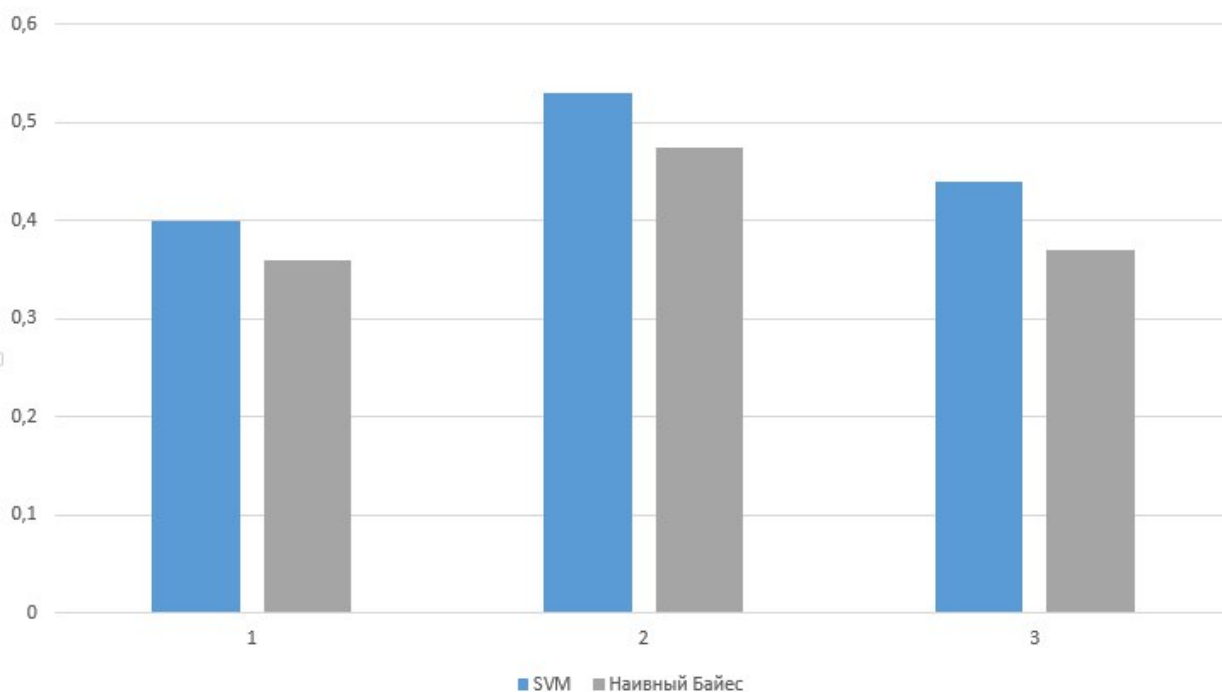


Рис. 7

Выводы

Из результатов видно, что метод опорных векторов отработал лучше, чем Наивный Байес. Также, можно заметить, что общий результат не так велик, как хотелось бы. Результаты можно попытаться улучшить путём выбора других признаков, также можно попробовать удалять стоп-слова, проводить лемматизацию, попробовать svm с линейным ядром, возможно, стоит оставить некоторые знаки препинания.

Заключение

В данной работе рассмотрены различные методы интеллектуального анализа данных, которые применены к анализу тональности русскоязычных высказываний в социальной сети в Twitter. Подробно разобраны методы предварительной обработки текста, описаны способы анализа тональности данных, а также разобраны методы машинного обучения, которые могут применяться для поставленной задачи. Основываясь на теоретических исследованиях социальной сети Twitter, а также особенностях машинного обучения, были проведены оценки различных подходов к решению задачи. Написан комплекс программ, который реализует рассмотренные подходы и на реальных данных проведены численные эксперименты. Проведено сравнение результатов теоретических исследований и экспериментальных результатов.

Список литературы

- [1] The perceptron: A probabilistic model for information storage and organization in the brain. Rosenblatt, F. Psychological Review. Vol 65(6), Nov 1958, 386-408.
- [2] [http://docs.scipy.org/doc/numpy-1.10.0/reference/generated/numpy.linalg.svd.ht](http://docs.scipy.org/doc/numpy-1.10.0/reference/generated/numpy.linalg.svd.html)
- [3] <http://www.dialog-21.ru/>
- [4] . Domingos. A Few Useful Things to Know about Machine Learning
- [5] Kiritchenko S., Xiaodan Z., Mohammad S. Sentiment Analysis of Short Informal 4 Texts
- [6] ishal A. Kharde, S.S. Sonawane Sentiment Analysis of Twitter Data: A Survey of Techniques
- [7] . Pang, Lillian L. LeeThumbs up? Sentiment Classification using Machine Learning Techniques