ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 50 за 2021 г.

ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

Н.Д. Баданина, В.А. Судаков

Модели машинного обучения для классификации отзывов о банках

Рекомендуемая форма библиографической ссылки: Баданина Н.Д., Судаков В.А. Модели машинного обучения для классификации отзывов о банках // Препринты ИПМ им. М.В.Келдыша. 2021. № 50. 14 с. https://doi.org/10.20948/prepr-2021-50
https://library.keldysh.ru/preprint.asp?id=2021-50

О р д е н а Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й а к а д е м и и н а у к

Н.Д. Баданина, В.А. Судаков

Модели машинного обучения для классификации отзывов о банках

Москва — 2021

Баданина Н.Д, Судаков В.А.

Модели машинного обучения для классификации отзывов о банках

На примере корпуса отзывов о банковских продуктах и услугах проводится анализ и составление моделей классификации текстов. В работе исследуются разные подходы к обработке неструктурированной текстовой информации. На основе выбранных подходов анализируется корпус отзывов о банковских продуктах и услугах, полученных в период пандемии COVID-19. Разработан автоматический парсер интернет-ресурсов для получения требуемой обучающей выборки. Разработано программное обеспечение, реализующее основные методы для построения моделей классификации. Данная модель может быть использована для создания систем мониторинга отношения населения к процессам в банковской сфере.

Ключевые слова: классификация, анализ данных, контекст документа, важность слов, лингвистика, машинное обучение

Natalya Dmitriyevna Badanina, Vladimir Anatolyevich Sudakov

Machine learning models for bank reviews classification

Using the banking products and services review corpus, analysis is conducted to establish different text classification models. The paper explores different approaches to the processing of unstructured textual information. Based on the selected approaches, the review corpus on banking products and services received during the COVID-19 pandemic is analyzed. An automatic Internet resources parser has been developed to obtain the required training sample. Software has been developed that implements basic methods for the classification models construction. This model can be used to create system for monitoring people’s attitudes to banking processes.

Key words: classification, data analysis, document context, words importance, linguistics, machine learning

Исследование выполнено при финансовой поддержке РФФИ и CNPq (Бразилия), Фонда содействия инновациям (Россия), DBT, DST (Индия), MOST, NSFC (Китай), SAMRC (ЮАР) в рамках научного проекта № 20-51-80002

Введение

Тема классификации текстов методами машинного обучения имеет широкие перспективы во многих сферах, в том числе в банковской сфере. Для банка, как для бизнеса, важно проводить мониторинг отзывов, доступных на специализированных интернет-площадках, в социальных сетях, при проведении опросов для поддержания высокого индекса потребительской лояльности, выявления случаев мошенничества как внутри, так и вне банковской структуры. Мониторинг вручную, с помощью специалистов, дорогостоящий, занимает большее количество времени, чем обработка текста с помощью моделей машинного обучения.

В данной работе исследуются алгоритмы классификации и обработки документов, написанных на естественных языках. Эти разработки активно используются при создании голосовых помощников, чат-ботов, умных устройства для дома. Цель работы — реализовать модели классификации отзывов, написанных на естественном языке, где целевой переменной выступает оценка, поставленная пользователем. Обученную модель можно применить для оценки комментариев в социальных сетях. Эта задача особенно актуальна в условиях распространения COVID-19, так как позволяет оперативно проводить мониторинг изменения общественного мнения, эмоциональной окраски сообщений. В работе проанализированы современные подходы, используемые для классификации текстов на естественном языке в соответствии с их тематикой. Выбранные методы работы с документами определяются сложной общей спецификацией задачи — зашумленные обучающие выборки, выборки недостаточного размера или искажение размера класса. Иными словами, алгоритмы были протестированы на реальных данных, имеющих свою специфику и особенности. В качестве данных для анализа и построения моделей обработки естественного языка (Natural Language Processing, NLP) был выбран корпус отзывов с сайта banki.ru. Банки.ру — независимый финансовый супермаркет. На сайте можно подобрать и оформить любой финансовый продукт (кредит, ипотека, вклад и т. д.) в различных банках. Также на сайте размещаются отзывы о банках, которые могут оставлять пользователи.

Основной сложностью при применении технологий NLP программными средствами является понимание алгоритмом контекста, в рамках которого идет обработка отдельного слова. Зачастую в тексте используются слова в переносном значении или в значении, которое установили собеседники между собой по договоренности. Особенно часто такое происходит в профессиональной литературе. При существовании множества смыслов язык становится избыточен. Избыточность является серьезной проблемой при построении алгоритмов NLP, так как разработчик такой системы не сможет и не будет указывать буквальный смысл каждого ассоциативного слова [1].

Следующая проблема связана со свойством слов менять свое значение с течением времени.

Специалисты по обработке и анализу текстов также сталкиваются с проблемой высокой и низкой энтропии. Это означает, что в системе существует мера неопределенности, в частности непредсказуемость появления какого-либо символа алфавита [1]. Рассмотрим задачу восстановления пропущенного слова в предложении. Фразеологизм "мастер на все ..." имеет низкую энтропию, то есть существует мало слов, которые подойдут для завершения предложения, тогда как "встретимся около ..." предполагает множество вариантов для завершения. Человек способен использовать не только контекст, но и предыдущие знания, чтобы ответить. Однако компьютерные модели необязательно обладают этой информацией.

Развитие технологии классификации текстов началось с введения спам-фильтров для почты. В 20 веке электронные письма были менее распространены, поэтому защитить почту от мошенников можно было точечной блокировкой адресов. С распространением технологии уже трудно представить ручную регулировку. Это подтолкнуло разработчиков к созданию первых алгоритмов классификации текстов. Развитие технологии классификации мошеннических писем остается актуальным и на сегодняшний день в связи с увеличением сервисов для коммуникации: Telegram, WhatsApp, mail и так далее. В перспективе с помощью этих же моделей можно классифицировать незаконные банковские транзакции, транзакции с участием криптовалют.

Технологические гиганты также ищут способ оптимизации бизнес-процессов. Рекомендательные системы позволяют им получать дополнительную прибыль и основывать бизнес-стратегию на системах искусственного интеллекта. В свою очередь, при построении рекомендаций фильмов, книг, новостей необходимо учитывать текстовые данные, указанные для объекта: заголовок, отзыв, рецензия, тег, категория. Правильно настроенная модель классификации текстов в перспективе улучшит работу рекомендательных движков, которые, в свою очередь, генерируют прибыль продуктов и сервисов.

Приложения, основанные на использовании естественного языка, только начинают распространяться, но в будущем могут взять на себя задачи, которые сейчас решаются стандартными формами и интерфейсами.

Предварительная обработка текста

В NLP токенизация — это особый тип фрагментации документа. Сегментация разделяет текст на небольшие части (сегменты) с более ограниченным информационным содержанием. Сегментация может включать разбиение документа на абзацы, предложения, фразы и на токены (слова), а также знаки препинания. Процесс сегментирования текста в токены называется токенизацией.

Чаще всего стоит задача разбивки текста на слова, однако к токенам относятся как слова, так и знаки пунктуации. Возможна разбивка на запятые, цифры, это зависит от задачи. Зачастую необходимо представить текст в виде

массива значимых слов. В этом случае после токенизации необходимо удалить знаки пунктуации и незначимые слова, например предлоги. Регулярные выражения используют специальный вид (класс) грамматики формального языка, называемый регулярной грамматикой. У нее предсказуемое, доказуемое поведение, и в то же время регулярные выражения достаточно гибкие для приведения в действие некоторых самых сложных диалоговых движков и чат-ботов на рынке. Amazon Alexa и Google Now — использующие регулярные грамматики движки, основанные главным образом на паттернах. Сложные правила регулярной грамматики часто можно выразить одной строкой кода, называемой регулярным выражением. Для языка Python существуют эффективные фреймворки для чатботов, такие как Will, полагающиеся исключительно на этот тип языка. Amazon Echo, Google Home и аналогичные помощники применяют регулярную грамматику с целью кодирования логики для большей части их взаимодействий с пользователем. Токенизаторы могут легко стать очень сложными. Например, программист может захотеть разделить текст на точки, но только если после точки не стоит число, чтобы не разбивать десятичные числа. Еще один пример, когда не нужно делить текст точкой, — точка является частью смайлика, как в сообщениях.

Большинство алгоритмов машинного обучения не могут работать с типом данных строка, поэтому для построения модели необходимо перевести текст в векторное пространство. В работе используется подход присваивания веса на основе TF-IDF показателя, где каждый признак получает вес, пропорциональный частоте появления признака в тексте (term frequency) и обратно пропорциональный количеству документов коллекции (inverse document frequency), в которых есть этот признак.

Рассмотрим Term Frequency. Количество вхождений слова в заданном документе называется частотностью терма (term frequency, TF) [2]. Иногда можно встретить нормализованные путем деления на общее число термов в документе количества слов. Это мера указывает, как часто слово t встречается в документе d из всего корпуса документов D. Обозначим её как $n_{t,d}$. Таким образом, каждое слово и документ будут иметь собственное значение TF.

$$tf_{t,d} = \frac{n_{t,d}}{\sum_k n_{k,d}}, \text{где } \sum_k n_{k,d} - \text{количество слов в документе.} \quad (1)$$

Inverse document frequency (IDF) — мера важности слова. Вычисляется как отношение количества документов в коллекции D к количеству документов в коллекции, в которых встречается заданное слово t.

$$idf_t = \log \frac{D}{\{d_i \in D | t \in d_i\}}. \quad (2)$$

Вычислив эти две меры, можно посчитать TF-IDF для каждого документа в корпусе. Слова с большим значением считаются более важными.

$$tf - idf_{t,d} = tf_{t,d} \times idf_t. \quad (3)$$

Таким образом, чем больше раз слово встречается в документе, тем выше будет значение TF (и, следовательно, TF-IDF). В то же время по мере увеличения количества документов, содержащих слово, его IDF (и, следовательно, TF-IDF) будет уменьшаться. Иногда все вычисления выполняются в логарифмическом пространстве, так что умножение становится сложением, а деление — вычитанием.

Одним из альтернативных способов использования непосредственно косинусного расстояния TF-IDF для ранжирования результатов запроса вариантов является Okapi BM25 или его самый последний вариант BM25F. При подсчете Okapi BM25 вместо простого вычисления косинусного подобия TF-IDF разработчики нормализовали и сгладили значения метрик, а также проигнорировали повторяющиеся термы в документе запроса. Скалярное произведение косинусного подобия нормализуется не векторной нормой TF-IDF (количеством слов в документе и запросе), а с помощью нелинейной функции от длины самого документа: у текстов, выражающих одно и то же разными словами или в различной орфографии, что дает различные представления векторов TF-IDF. Это затрудняет работу поисковых систем и средств сравнения документов, в основе которых лежит подсчет количества токенов.

В рамках поставленной задачи целью является выяснить список слов, играющих важную роль для указанного документа из корпуса всех документов. С помощью такого списка в дальнейшем можно искать релевантные документы в корпусе на основе коэффициента близости. Для обучения моделей необходимо получить численные значения из строковых данных. Был использован алгоритм TF-IDF. Такая модель является статистической, так как основывается на частотностях слов. Однако количество слов, даже при условии нормализации количества по длине документа, не дает достаточно информации.

После применения TF-IDF преобразования к лемматизированному датасету на выходе получается матрица. Строчки матрицы соответствуют конкретному отзыву. Столбцы соответствуют словам, встреченным в корпусе, состоящем из всех отзывов использованного датасета. Значения — коэффициенты TF-IDF. Матрица будет считаться разреженной, то есть матрицей с преимущественно нулевыми элементами.

Оценка качества

Качество результата — совокупность свойств объекта, обусловливающих его способность удовлетворять заданные потребности в соответствии с его назначением. Качество выступает обычно как ограничение, то есть при любой разработке продукт должен иметь качество не ниже заданного уровня [3]. Таким образом, при разработке программного обеспечения и моделей необходимо определиться с метрикой качества, чтобы понимать, удовлетворяют ли результаты поставленным целям.

Введем несколько определений: ошибка первого рода (Type one error, false positive, FP) — ситуация, когда модель бинарной классификации отнесла объект класса 0 к классу 1, ошибка второго рода (Type two error, false negative, FN) — ситуация, когда модель бинарной классификации отнесла объект класса 1 к классу 0.

Аналогично правильные предсказания можно разделить на true positives (TP) и true negatives (TN). Для реализации функционала оценки многокритериальных классификаторов существует другой набор метрик. При извлечении сущностей, атрибутов и отношений общепринятым способом рассчитываются оценки качества модели. Точность (Accuracy) — доля объектов с правильно предсказанным классом (accuracy и другая рассматриваемая метрика precision переводятся на русский язык одинаково). Точность (Precision) можно определить как суммарную частоту правильных ответов, поделенную на сумму всех найденных ответов. Полнота (Recall) — количество правильных ответов, деленное на общее число правильных ответов. Дополнительной метрикой оценки качества извлечения служит F-мера — соотношение между точностью и полнотой, чаще всего определяющееся как гармоническое среднее.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN},$$
$$Precision = \frac{TP}{TP + FP}, \quad (4)$$
$$F = \frac{2RP}{R + P}, \text{ где } R \text{ — полнота, } P \text{ — точность.}$$

Получение данных из веб-ресурса

На языке программирования Python3 был реализован парсер для сбора данных с веб-ресурса. Для каждой страницы с отзывами, сайтом генерируется индивидуальная ссылка, включающая в себя номер страницы на дату просмотра (при разных датах просмотра номер может меняться из-за добавления новых страниц с отзывами). Осуществляется запрос (request) по индивидуальной ссылке (url). Далее, получив содержание веб-страницы в формате HTML, по тегам извлекается информация, относящаяся к отзыву. Дело в том, что HTML-страница состоит из множества блоков: блок отзывов, рекламы, информационный блок и так далее. Для дальнейшего анализа нужен только блок отзывов. Из этого блока по HTML-тегам получаем необходимые поля: заголовок отзыва, оценка пользователя, ссылка на полный текст, название банка, дата и время публикации. После формирования датасета из блоков веб-страницы была проведена очистка от лишних символов табуляции, символов перехода строки. Интересно, что архитектура веб-сайта устроена таким образом, что полный текст отзыва хранится по отдельной уникальной ссылке. Для каждого отзыва был получен полный текст.

Данные, полученные после парсинга веб-ресурса, необходимо подвергнуть ряду преобразований, так как алгоритмы NLP не работают с чистыми языковыми конструкциями. Во-первых, текст отзыва и текст заголовка были разбиты на токены — слова, исходя из наличия пробела. Самый простой способ токенизации предложения — использовать пробелы в качестве разделителей слов в строках. В Python3 для этого подходит стандартный метод split из общедоступной библиотеки, который доступен для всех экземпляров объекта str. В данной работе рассматриваются регулярные выражения, как наиболее гибкий инструмент для разбиения текста на токены. Каждый токен был приведен к нижнему регистру. Case folding (выравнивание регистра) — это сочетание нескольких вариантов написания слова, которые различаются только регистром букв. Из текста с помощью регулярного выражения извлекаются все слова, при этом удаляются цифры, знаки пунктуации, иностранные слова и символы. Это обусловлено тем, что нерусскоязычные словесные конструкции помешают при дальнейшей разработке системы. Так как сайт ориентирован на пользователей из РФ и стран СНГ, то и текст отзывов в подавляющем большинстве содержит русскоязычные конструкции.

Важной составляющей предобработки текстовых данных является удаление стоп-слов. Стоп-слова — это часто используемые слова на любом языке, которые не несут в себе смысловой нагрузки. Корпус стоп-слов для русского языка можно получить из библиотеки NLTK. Пример таких слов: "а", "без", "более", "больше", "будет", "будто", "бы", "был", "была", "были", "было" и так далее. Однако библиотека содержит информацию о 151 слове и является неполной, поэтому из электронного ресурса был скачан дополненный список, который содержит 421 слово [4]. После всех вышеперечисленных преобразований с помощью библиотеки Pymorphy2 производится лемматизация оставшихся слов. По первоначальной словоформе и тегам осуществляется поиск нормальной формы слова. При предварительном анализе слов, неизвестных в Pymorphy2, последовательно применяется ряд методов. Сначала префикс отделяется от слова, которое содержится в списке префиксов, которые уже известны. Если остаток слова был найден в списке, удаленный префикс присваивается результатам сканирования. Кроме того, в случае неудачи в конце анализируется форма слова. В случае, когда слово имеет множество вариантов разбора — выбирается наиболее вероятный.

$$P(w|t) = \frac{F_r(w,t)+1}{F_r(w)+|R(w)|}, \quad (5)$$

$F_r(w)$ — количество раз, которое словоформа w встретилась в корпусе;
$F_r(w,t)$ — количество раз, которое словоформа встретилась с тегом t;
$|R(w)|$ — число разборов, полученных для словоформы w.

Анализ исходных данных

Получив лемматизированный корпус отзывов с оценками пользователей, проведем первичный анализ датасета с целью выявления закономерностей. Чаще всего публикуются отзывы с негативной оценкой 1, но это можно объяснить тем, что довольный клиент менее склонен к написанию положительного отзыва, чем недовольный – к написанию негативного. Поэтому компаниям так важно стимулировать удовлетворенных клиентов к написанию отзывов и рецензий.

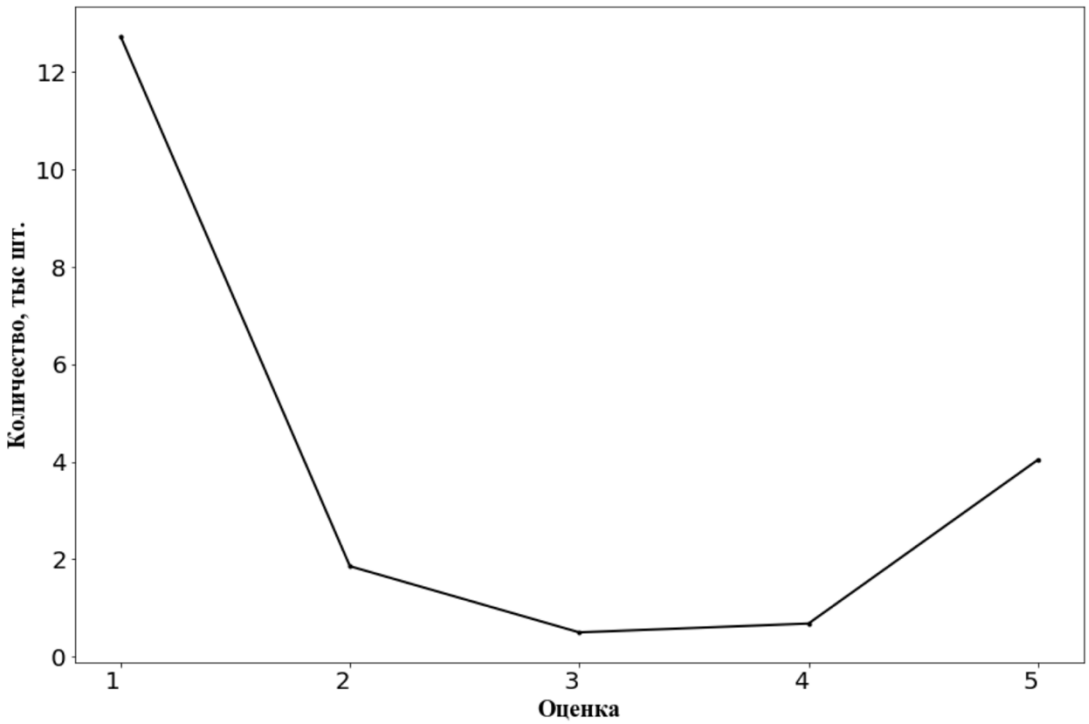

Рис. 1. Анализ количества отзывов для оценок

Рассмотрим длинные слова (длина которых больше 15 символов русского алфавита). Гипотеза заключается в том, что такие слова будут более информативны. Назовем такое свойство P, так, P(w) принимает значение истина, если w длиннее, чем 15 символов русского алфавита. Итак, получаем математическое описание условия: множество всех слов w, таких, что w является элементом пространства V (корпус слов) и w удовлетворяет свойству P.

$$\{w|w \in V \& P(w)\} \quad (6)$$

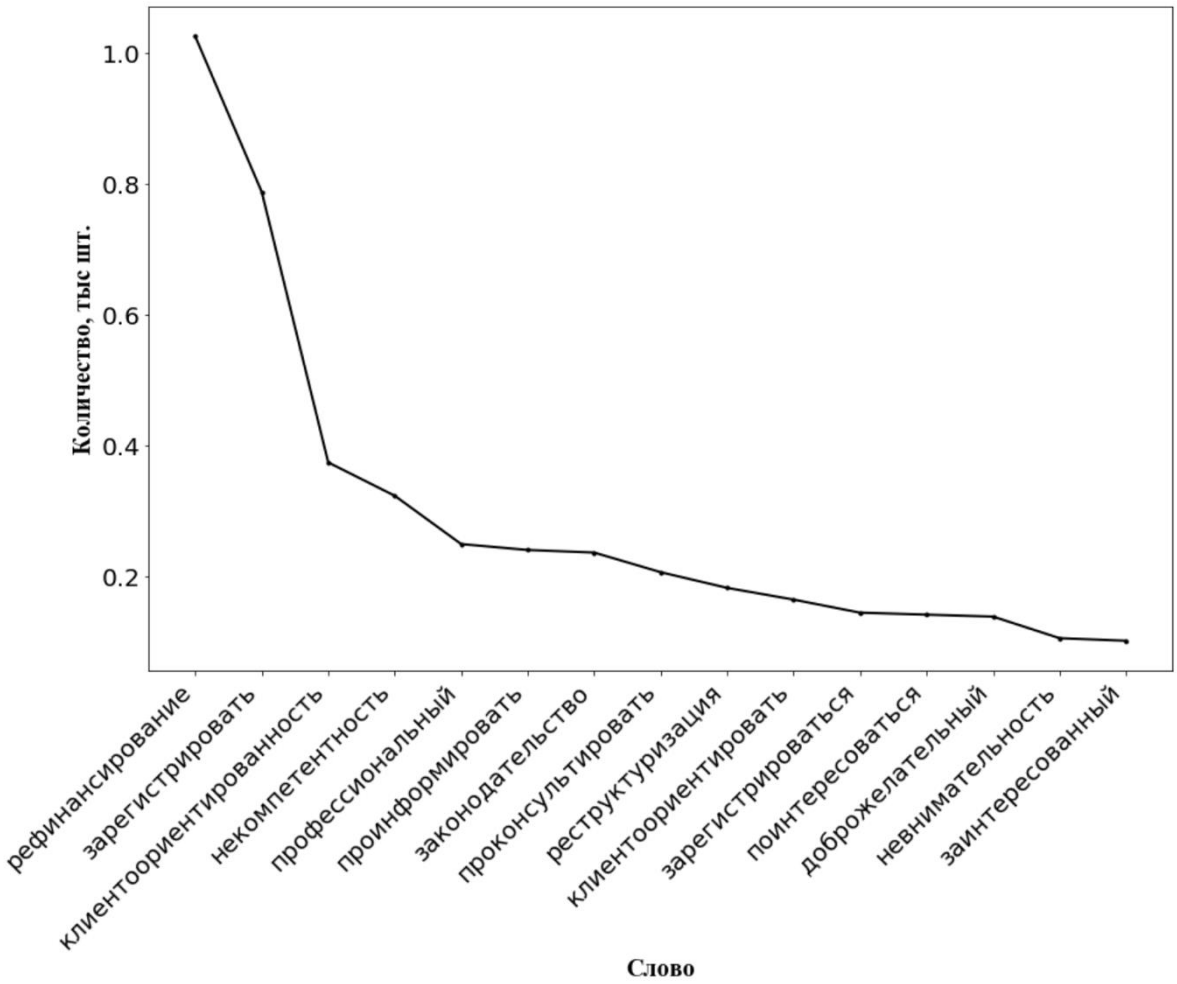

Рис. 2. Анализ слов, длина которых больше 15 символов

Выделяются два продукта: рефинансирование и реструктуризация. Рефинансирование — это оформление нового кредита для погашения уже имеющегося. Рефинансирование позволяет заемщику улучшить условия по кредиту: получить лучшую процентную ставку, продлить срок и уменьшить ежемесячную выплату. Кроме этого, при рефинансировании можно объединить два-три кредита в один, что на практике означает погашение средствами нового кредита долгов, например, по кредитным картам. Реструктуризация долга — это мера, которая применяется в отношении заемщика, который пребывает в состоянии дефолта, иными словами, не способен выплатить свой долг. Реструктуризация может относиться как к физическим, так и юридическим лицам. Также можно выделить отрицательные характеристики, которые беспокоят пользователей: некомпетентность, невнимательность — и положительные: клиентоориентированность, профессиональный, доброжелательный, заинтересованный.

Проведя анализ частоты появления слов в заголовках и в полном тексте отзывов, можно проверить выполнение закона Ципфа. Закон Ципфа был вдвинут стенографистом Жан-Батистом Эсту в его научной работе. Закон гласит, что если упорядочить все слова исследуемого языкового корпуса по

убыванию частоты их использования, то частота слова с порядковым номером n окажется приблизительно обратно пропорциональной его порядковому номеру, то есть рангу слова. К примеру, пятое по частоте слово будет встречаться в пять раз реже, чем первое. Закон Ципфа можно применить к различным сферам исследования. Проверим выполнение закона на словах, встречающихся в полном тексте отзывов. Для графика возьмем первые 100 слов по частоте появления.

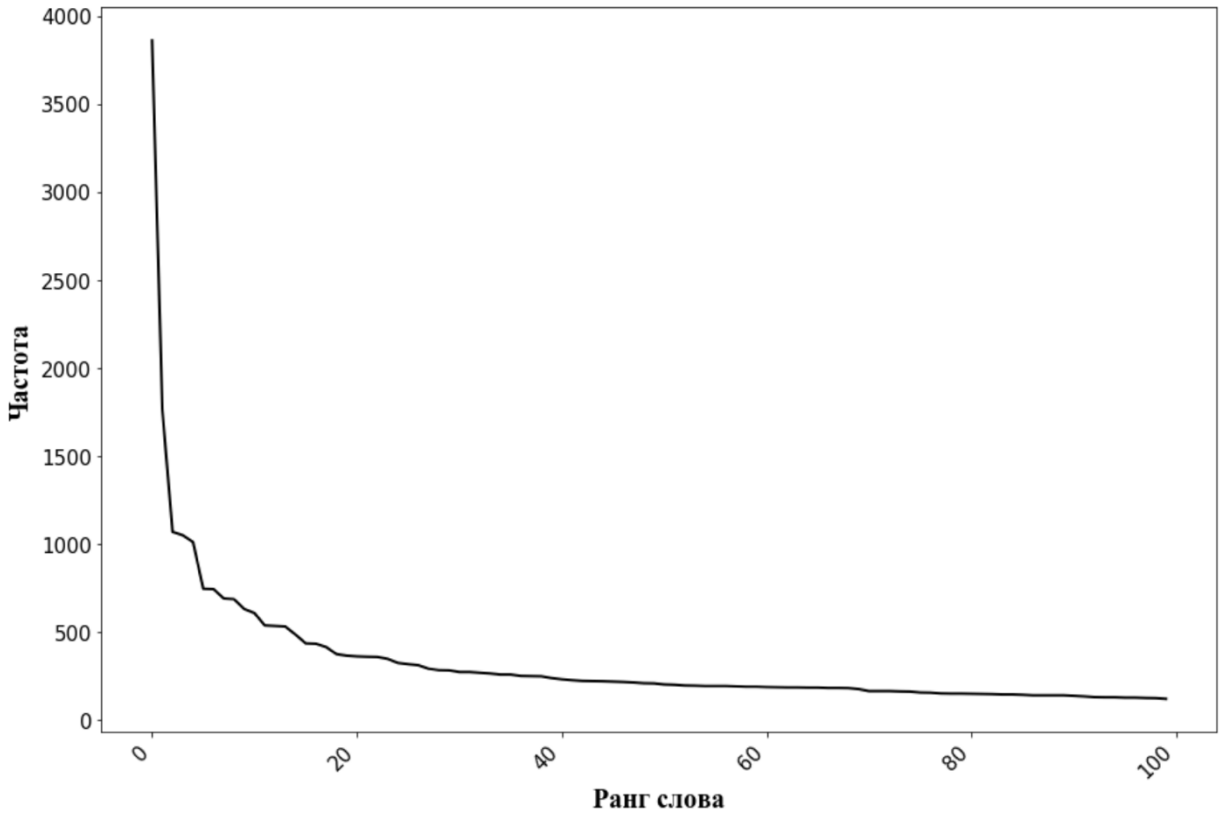

Рис. 3. Закон Ципфа, проверенный на тексте отзывов

Алгоритмы классификации

Целью алгоритмов машинного обучения (Machine learning, ML) является подгонка существующих данных под некую модель, которая помогает принимать решения, предоставлять прогнозы на основе новых данных, путем поиска взаимосвязей и закономерностей в данных. Далее обученной модели передаются новые данные, на основе которых модель строит прогноз, возвращает вероятности, метки, признаки принадлежности.

Среди алгоритмов ML, применяемых в рамках анализа текстовых данных, можно выделить методы обучения с учителем (supervised learning, SL), методы обучения без учителя (unsupervised learning, UL), методы частичного обучения с учителем (bootstrapping) [5]. Чаще всего применяется обучение с учителем, так как алгоритмы этого класса быстрее и качественнее работают с текстами. С помощью ML может быть построен машинный классификатор, который может распознавать разные классы текста. Построение классификатора происходит на предварительно размеченном

текстовом корпусе (обучающей выборке), в котором данным присваиваются метки, кодирующие их признаки. Обучение можно определить как выявление общих закономерностей на основе данных обучающей выборки. Первичной задачей является выявление характерных признаков в данных, которые способны предсказать целевую переменную (метку). Однако классификаторы непрозрачны для понимания и интерпретации.

Вместе с общераспространенными методами машинного обучения для NLP, такими как деревья решений, наивный байесовский классификатор, логистическая регрессия, метод опорных векторов, исследователи начали применять метод условных случайных полей (condition random fields, CRF), скрытые модели Маркова (Hidden Markov Models, HMM) и нейронные сети [6].

Поскольку локальный контекст классифицированного токена активно используется для решения проблемы извлечения объекта, проблему классификации также можно рассматривать как проблему прогнозирования последовательности. В этом случае логичнее использовать не классические методы обучения (байесовский классификатор, деревья решений), а скорее скрытые марковские модели (HMM) и метод условных случайных полей (CRF), с категориями именованных сущностей как скрытых состояний и токенов как наблюдаемых.

В последние годы появились работы с использованием нейронных сетей и подходов глубокого обучения, например, технологии Word2vec. Особенностью методов с использованием нейронных сетей является то, что они позволяют достичь качества, сопоставимого с лучшими современными методами (примерно 91% F-мер), но с минимальным набором дополнительной информации: ресурсы, токены, ресурсы словаря.

В работе были реализованы распространенные алгоритмы машинного обучения. В них включены как алгоритмы, основанные на линейном подходе, так и деревья решений. Также для алгоритма LinearSVC был реализован поиск гиперпараметров с целью улучшения точности на тестовой выборке. Время выполнения может варьироваться в зависимости от характеристик машины, на которой производятся вычисления. Выведем дополнительные метрики для классификатора с наилучшим показателем точности: SGDClassifier. Precision: 0.81, Recall: 0.81, F1-measure: 0.81.

Таблица 1

Результаты работы алгоритмов

Название алгоритма	Точность на тренировочной выборке	Точность на тестовой выборке	Время выполнения
SGDClassifier	84.7%	81.5%	1 мин.
LinearSVC + GridSearchCV	86.0%	81.3%	18 мин.

Название алгоритма	Точность на тренировочной выборке	Точность на тестовой выборке	Время выполнения
LogisticRegression	83.7%	81.2%	11 мин.
LinearSVC .	95.3%	80.4%	2 сек.
RandomForestClassifier	99.9%	78.9%	3 мин.
MLPClassifier	99.9%	78.4%	58 мин.
PassiveAggressiveClassifier	99.9%	75.7%	2 мин.
MultinomialNB	71.9%	70.3%	2 сек.
DecisionTreeClassifier	99.9%	66.8%	12 мин.
Sequential	64.3%	64.2%	2 мин.

Заключение

Целью работы являлась реализация модели классификации отзывов, написанных на естественном языке, где целевой переменной выступает оценка, поставленная пользователем. В качестве данных был получен и использован корпус отзывов на банковские продукты и услуги. Данные были выгружены с сайта banki.ru с помощью реализации парсера страницы. Такое решение обусловлено отсутствием необходимых данных в открытых источниках. Это также указывает на уникальность решенной задачи.

В ходе выполнения работы было обучено десять моделей классификации, в том числе модель нейросети. Наилучшие результаты показала модель SGDClassifier с точностью 81.5%. Обученную модель можно применить для оценки комментариев в социальных сетях, так как комментарии и отзывы имеют схожую структуру. Потенциально пользователям социальных сетей, оставившим положительный комментарий, можно предлагать оставить отзыв на специализированном форуме с целью повысить рейтинг банка или банковского продукта.

При работе с языковыми конструкциями, написанными на естественном языке, возникает множество сложностей, связанных с уникальностью русского языка, а также с пониманием программы контекста. Это делает каждую задачу сферы NLP уникальной и нетривиальной. С развитием чат-ботов и голосовых помощников необходимость в изучении алгоритмов по обработке текстов и звуковых файлов с речью увеличивается. Такие задачи актуальны на сегодняшний день и будут продолжать развиваться в будущем.

Использование предложенных моделей позволит получить обобщённые индикативные оценки отношения населения к банкам, банковским продуктам, финансовым инструментам, валютам и криптовалютам. Что

особенно актуально в современных условиях, когда пандемия приводит к быстрому изменению информационной среды.

Библиографический список

1. Большакова Е.И., Воронцов К.В., Ефремова Н.Э., Клышинский Э.С., Лукашевич Н.В., Сапин А.С. Автоматическая обработка текстов на естественном языке и анализ данных: учеб. Пособие. М.: Изд-во НИУ ВШЭ. 2017. 269 с. URL: https://www.hse.ru/data/2017/08/12/1174382135/NLP_and_DA.pdf
2. Белова К.М., Судаков В.А. Исследование эффективности методов оценки релевантности текстов // Препринты ИПМ им. М.В.Келдыша. 2020. № 68. 16 с. http://doi.org/10.20948/prepr-2020-68 URL: http://library.keldysh.ru/preprint.asp?id=2020-68
3. Судаков В.А. Автоматизация процесса управления разработкой корпоративной информационной системы // Вестник Московского авиационного института. 2010. Т. 17. № 1. с. 149-153 URL: https://mai.ru/upload/iblock/162/1620fcb930b966087b2a6777160a660f.pdf
4. Список стоп-слов для русского языка [Электронный ресурс]: https://countwordsfree.com/stopwords/russian
5. Bengfort B., Bilbro R., Ojeda T. Applied Text Analysis with Python. США, Калифорния: O'Reilly Media. 2018. 332 с.
6. Machine Learning Algorithm Classification for Beginners [Электронный ресурс]: https://serokell.io/blog/machine-learning-algorithm-classification-overview.

Оглавление