

Федеральное государственное образовательное бюджетное учреждение  
высшего профессионального образования  
«Финансовый университет при Правительстве Российской Федерации»

Департамент анализа данных, принятия решения и финансовых  
технологий

**Курсовая работа**

по дисциплине "Технологии анализа данных и машинное обучение" на  
тему:

**"Классификация текстов методами машинного обучения"**

Вид исследуемых данных: Корпус новостей с сайта Lenta.Ru

Выполнила:  
студентка группы ПМ17-1

Баданина Н. Д.

Научный руководитель:  
доктор техн. наук, профессор  
Судаков Владимир Анатольевич

Москва 2020

# Содержание

Введение . . . . .	3
0.1 Сложности при обработке текстов на естественном языке . . . . .	4
0.2 Перспективы развития технологии . . . . .	4
1 Теоретическая часть . . . . .	6
1.1 Задачи классификации текстов . . . . .	6
1.2 Формализация процесса обработки текста . . . . .	7
1.2.1 Разметка текстовых данных . . . . .	7
1.2.2 Токенизация . . . . .	7
1.2.3 Исправление опечаток . . . . .	7
1.2.4 Сегментация предложений . . . . .	7
1.2.5 Лемматизация. Стемминг . . . . .	8
1.2.6 Извлечение ключевых слов . . . . .	9
1.2.7 Модель представления текста . . . . .	9
1.2.8 Модели классификации . . . . .	11
1.2.9 Оценка качества . . . . .	12
2 Практическая часть . . . . .	15
2.1 Инструменты разработки . . . . .	15
2.2 Анализ данных . . . . .	15
2.3 Работа с пропущенными значениями . . . . .	17
2.4 Предварительная обработка данных . . . . .	18
2.4.1 Лемматизация . . . . .	19
2.4.2 Кодирование категориальных данных . . . . .	21
2.4.3 TF-IDF преобразование . . . . .	21
2.5 Разбиение выборки на обучающую и тестовую . . . . .	22
2.6 Итоги работы классификаторов . . . . .	22
Заключение . . . . .	23
Список использованных источников . . . . .	24
Приложение . . . . .	25

# Введение

На сегодняшний день технологии развиваются с экспоненциальной скоростью. Это значит, что за следующий десяток лет появится больше инновационных продуктов, чем за предыдущий десяток. За короткий период времени в научном мире развилась целая новая область знаний, которая требует изучения — это Искусственный Интеллект (ИИ, artificial intelligence, AI). В AI, как подраздел, можно включить машинное обучение (machine learning, ML), алгоритмы машинного обучения. В свою очередь алгоритмы ML можно условно разделить на четыре направления. Первое направление — классическое обучение. В него входят такие методы, как классификация, регрессия, кластеризация, уменьшение размерности и другие. Часть перечисленных методов реализованы в данной работе. Оставшиеся направления включают в себя нейронные сети и глубокое обучение, обучение с подкреплением, ансамблевые методы.

Пик развития машинного обучения начался ориентировочно в 2015 году, когда бизнесы начали активно цифровизироваться, уходить в онлайн, для чего им понадобилось внедрение большого числа современных цифровых технологий, в том числе методов машинного обучения. Также, развитие алгоритмов обусловлено борьбой за клиентов, которым важна и безопасность продукта, и доступность, и персонализация. Это подтолкнуло компании к вложению средств в изучение области AI.

В данной работе речь пойдет об алгоритмах классификации и обработки текста на естественном языке. Эти разработки активно используются при создании голосовых помощников, чат-ботов, умных устройства для дома. Технологии, основанные на распознавании естественных языков создают новый интерфейс для взаимодействия с пользователем. К примеру, пользователю достаточно устно озвучить запрос, не вводя ничего привычным способом. В речевом анализе аудиоданные преобразуются в текст, к которому можно применить алгоритмы Natural Language Processing (NLP) — методы обработки текстов на естественном языке.

Также примером внедрения анализа естественного языка может служить поддержка "тегов рекомендаций", реализованная у Netflix, YouTube. Тег — это метоинформация о фрагменте контента, важная для поиска и рекомендации. Теги определяют свойства описываемого ими объекта и могут быть использованы для группировки схожих элементов и предложения описательных названий для таких групп. Широкое использование теги получили в электронных новостных изданиях, к примеру Я.Дзен, Lenta.ru, Тинькофф Журнал. Для выполнения практической части курсовой работы мною были взяты размеченные данные с сайта Lenta.ru. Данные содержат текст новости, заголовок, тег.

## 0.1 Сложности при обработке текстов на естественном языке

Основной сложностью при обработке текстов на естественном языке 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.

Следующая проблема связана со свойством слов менять свое значение с течением времени. Единицей анализа текста является лексема — слово как абстрактная единица морфологического анализа. Лексема "батарея" изменила свой смысл с течением времени. Так, в текстах 20 века и ранее можно увидеть это слово для обозначения артиллерийского подразделения из нескольких орудий. В современных публикациях лексема используется для обозначения хранилища, преобразующего химическую энергию в электронную.

Специалисты по обработке и анализу текстов также сталкиваются с проблемой высокой и низкой энтропии. Это означает, что в системе в большей или меньшей мере существует мера неопределенности, в частности непредсказуемость появления какого-либо символа алфавита. 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. Человек способен использовать не только контекст, но и предыдущие знания, чтобы ответить. Однако, компьютерные модели не обязательно обладают этой информацией.

## 0.2 Перспективы развития технологии

Развитие технологии классификации текстов началось с введения спам-фильтров для почты. В 20 веке электронные письма были не 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 актуальной 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 процессов. Рекомендательные системы позволяют им получать дополнительную прибыль и основывать бизнес стратегию на AI системах. В свою очередь, при построении рекомендаций фильмов, книг, новостей необходимо учитывать текстовые данные, указанные у объекта: заголовок, отзыв, рецензия, тег, категория. Правильно настроенная модель классификации текстов в перспективе улучшит работу рекомендательных движков, которые в свою очередь генерируют прибыль 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.

# 1 Теоретическая часть

## 1.1 Задачи классификации текстов

Основные задачи NLP:

1) Классификация текстов. Модели направлены на определение текста к тому или иному классу исходя из его признаков и особенностей.

— фильтрация спама. Целевой переменной выступает булева переменная, принимающая значение 1 или 0 в случае наличия или отсутствия в письме спама соответственно.

— анализ тональности. Применяется для классификации отзывов, сообщений, твитов по положительному, отрицательному или нейтральному отношению автора.

— определение интента, то есть намерения пользователя. К примеру, при вводе поискового запроса система "подсказывает" возможные продолжения.

— деление текстов по теме/жанру/тегу. Эта задача реализована в практической части данной работы.

2) Кластеризация текстов. Разбиение коллекции документов на кластеры, то есть на подмножества близких по темам документов.

— агрегация новостей. Агрегатор осуществляет сбор и систематизацию новостей так, что интернет-пользователь может искать информацию в создаваемом информационном массиве по ключевым словам, при этом исключаются ненужные ресурсы, которые также могут содержать ключевые слова.

— рекомендации. Кластеризация документов, построение рекомендаций.

3) Извлечение информации.

— именованные сущности (Named-entity recognition, NER). Задача NER — выделить спаны сущностей в тексте, где спан — непрерывный фрагмент текста. Заранее определяется некоторый фиксированный набор — например, люди, места, организации, время и так далее. Из текста необходимо выделить слова, которые принадлежат одному из наборов.

— выделение атрибутов объектов и семантических отношений между ними.

4) Машинный перевод. Является одной из самых ранних разработок в сфере NLP. Первые модели использовали стратегию пословного перевода, но быстро стало понятно, что необходимо учитывать и другие лингвистические модели. Несмотря на долгое время существования, качество машинного перевода все ещё далеко до отличного. Прорыв в этой области основан на возникновении нейронных сетей, на основе которых работают интернет переводчики от Google, Яндекс.

5) Вопросно-ответные системы. Одно из самых частых применений — чат-боты, которые используются в совершенно разных отраслях, включая даже банки.

6) Генерация текстов. Технологии машинного обучения позволяют создавать

оригинальные тексты, указав, к примеру, начальное слово.

7) Распознавание речи. Активно применяется при создании голосовых помощников.

8) Проверка правописания.

9) Реферирование текста (Summarization), то есть сокращение объема текста и получение краткого изложения содержания. Также саммаризацию можно выполнять по нескольким близким документам, к примеру по новостям. Близкой задачей является аннотирование текста, то есть выделение ключевых тем.

## **1.2 Формализация процесса обработки текста**

### **1.2.1 Разметка текстовых данных**

Размеченные данные, то есть такие, у которых известна целевая переменная, например категория, можно анализировать с помощью методов ML с учителем. "Учителем" выступают сами разметки, так как однозначно понятно какой ответ разработчикам хотелось бы получить от алгоритма. Данные имеют метки, то есть понятно, к каким категориям принадлежат записи. Обычно быстрее, проще и дешевле найти и разметить достаточно данных, на которых будет обучаться модель — вместо того, чтобы пытаться оптимизировать сложный метод обучения без учителя, когда разметка отсутствует.

### **1.2.2 Токенизация**

Токенизацией называют разбивку текстов на более мелкие части — токены. 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.

### **1.2.3 Исправление опечаток**

В редких случаях в процесс обработки текста включают исправление опечаток. Дело в том, что это очень трудоемкий и кропотливый процесс, которые скорей всего не увеличит точность работы модели. Однако, в отдельно взятых случаях он необходим.

### **1.2.4 Сегментация предложений**

В зависимости от задач, может потребоваться токенизация не по словам, а по предложениям. К примеру, для решения задачи саммаризации. В таком случае предстоит разделить текст на предложения. Сложность заключается в определении конца и начала, так как не всегда предложение заканчивается точкой, может стоять восклицательный, вопросительный знак. Также не всегда точка

значит конец, знак используется для сокращений, обозначений дат. Заглавная буква также может встречаться как в начале, так и в именах, названиях и не является неоспоримым признаком начала предложения.

### 1.2.5 Лемматизация. Стемминг

Лемматизация — приведение слов к начальной форме. Это необходимо, так как при представлении текста в векторном пространстве, слова с разными формами, но одинаковым значением будут считаться как разные признаки. Считается, что для каждого токена существует его нормальная форма или лемма. От этой начальной формы создаются все остальные формы слова путем флексии, то есть изменений этой начальной формы. Рассмотрим пример на слове "сделали": сделал — основа, частица -и — флексия.

Введем новый термин — словоформа. Под словоформой будем понимать группу, состоящую из токена, связанной с ним начальной формы и множества грамматических параметров. Тогда, исходя из данного определения, можно сказать, что задачей лемматизации является нахождение в словаре словоформы по её токену.

При анализе текстовых данных можно считать количество словоупотреблений токенов, однако, аналогично, лучше рассчитывать сколько раз встретилась каждая из лексем, чтобы избежать дублирования слов разной формы. Для этого необходимо предварительно провести лемматизацию. В итоге выполнения преобразований получится вектор. Вектор может быть построен как для лемм, так и для словоформ или токенов, в зависимости от того, какая задача стоит перед разработчиками.

Векторное представление позволяет перейти от текста к его описанию в некотором пространстве. Каждая лемма задаёт собственное направление в некоем многомерном пространстве, размерность которого будет равна количеству лемм в тексте. В таком случае, текст можно представить как точку или вектор в этом многомерном пространстве. Переход к многомерному пространству позволяет измерять расстояния между текстами, то есть степень, в которой они похожи или не похожи друг на друга. Если в двух текстах употребляются одни и те же слова, то эти тексты посвящены одной или сходной теме.

В зависимости от постановки задачи, морфологический анализатор может возвращать разную информацию. Если необходимо рассчитать вектор частотности употребления слов в тексте, то необходимо использовать только лемму или лемму с указанием части речи. Для языков с невысокой флексивностью применяется стемминг, то есть определение основы слова путём отбрасывания окончаний из известного набора. Отличие стемминга от лемматизации можно продемонстрировать на примере слова выходцы: процесс лемматизации определит лемму выходец, в то время как стемминг вернёт псевдооснову выход. При анализе корпусов на русском языке предпочтительнее



использовать лемматизация в виду разнообразия основ и превдооснов языка.

### 1.2.6 Извлечение ключевых слов

Ключевые слова — это лексические группы, отражающие содержание документа. Проблематичным является автоматическое извлечение многокомпонентных ключевых слов. При всех методиках, алгоритм извлечения ключевых слов универсален и включает этапы:

- а) формирования множества «кандидатов» в ключевые слова
- б) фильтрации этого множества для получения результирующего списка ключевых слов.

Достаточно часто до извлечения ключевых слов из текста удаляются стоп-слова. Стоп-слова — это слова, которые не несут никакой смысловой нагрузки (артикли, предлоги, союзы, частицы, местоимения, вводные слова, междометия и так далее)

### 1.2.7 Модель представления текста

Модель представления текста в векторном пространстве. Label Encoding (LE) — простой подход, основанный на переводе каждого значения в цифру. Категориальные значения сортируются и им присваиваются упорядоченные порядковые номера. Используется для кодировки меток в датасете.

Animal Category	LE
Dogs	0
Cats	1
Birds	2

LE прост в применении, однако его минусом является то, что другие алгоритмы ML могут принять численные категории как иерархию, увидеть порядок в них. Эта проблема исправляется в алгоритме One-Hot Encoding (ONE). Каждый признак конвертируется в новую колонку. Признакам присваивается значение 1 или 0 в зависимости от соответствия новой колонки и первоначального признака.

Animal Category	Dogs	Cats	Birds
Dogs	1	0	0
Cats	0	1	0
Birds	0	0	1

Разновидностями признаковой модели являются модель bag of words (BOW — мешок слов), в которой текст характеризуется набором своих значимых

слов. Обычно это леммы всех ключевых слова, а также векторная модель текста, в которой указанный набор упорядочен. При применении BOW теряется информация о взаимном расположении слов внутри текста. При его использовании семантическая близость двух текстов оценивается по количеству совпадающих слов. Это означает, что два текста, в которых мало общих слов или вообще нет, считаются семантически и тематически неблизкими. Векторная модель применяется, например, в информационном поиске, при этом в качестве признаков чаще берутся не слова, а более сложные характеристики, такие как показатель TF-IDF для слов. При построении BOW строится корпус всех уникальных слов в документах и подсчитывается их количество в каждом документе. Недостатком является то, что при добавлении нового слова в корпус размерность матрицы увеличится. Более того, матрица будет содержать большое количество нулей. также не учитывается порядок слов.

Review	This	movie	is	very	scary	not	good
This movie is very very scary	1	1	1	2	1	0	0
This movie is not scary	1	1	1	0	1	1	0
This movie is good	1	1	1	0	0	0	1

Также часто используется известный подход взвешивания на основе веса TF-IDF, 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 Frequent (TF). Это мера указывает как часто слово  $t$  встречается в документе  $d$ . Обозначим как  $n_{t,d}$ . Таким образом, каждое слово и документ будут иметь собственное значение TF.

$$tf_{t,d} = \frac{n_{t,d}}{\text{Number of terms in the document}}$$

Inverse document frequency (IDF) — мера важности слова.

$$idf_t = \log \left( \frac{\text{number of documents}}{\text{Number of documents with the term } t} \right)$$

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

$$(tf - idf)_{t,d} = tf_{t,d} * idf_t$$

### 1.2.8 Модели классификации

Целью машинного обучения является подгонка существующих данных под некоторую модель, помогающую принимать решения или генерировать прогнозы на основе новых данных, путем поиска закономерностей в них. Затем обученной модели можно передавать новые данные, на основе которых она будет строить прогноз и возвращать метки, вероятности, признаки принадлежности или значения.

Среди алгоритмов ML, применяемых в рамках анализа текстовых данных, можно выделить методы обучения с учителем (supervised learning), методы обучения без учителя (unsupervised learning), методы частичного обучения с учителем (bootstrapping). Чаще всего применяется обучение с учителем, так как алгоритмы этого класса быстрее и качественнее работают с текстами. Строится машинный классификатор, который умеет распознавать различные классы текста. Построение классификатора происходит на предварительно размеченном текстовом корпусе (обучающей выборке), в котором данным присвоены метки, кодирующие их признаки. Обучение представляет собой, по сути, 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.

Наряду с традиционными методами машинного обучения для NLP, такими как наивный байесовский классификатор, деревья решений, метод опорных векторов, логистическая регрессия, все чаще применяются скрытые модели Маркова (Hidden Markov Models, HMM), метод условных случайных полей (condition random fields, CRF) и нейронные сети.

В то же время, поскольку при решении задачи извлечения сущностей активно используется локальный контекст классифицируемого токена, то эту задачу можно рассматривать как задачу предсказания последовательности. В этом случае логичнее использовать не классические методы обучения (байесовский классификатор, деревья решений), а скрытые марковские модели (HMM) и метод условных случайных полей (CRF), рассматривая категории именованных сущностей как скрытые состояния, а токены — как наблюдаемые.

В последние годы появились работы, в которых применяются нейронные сети и используются подходы на основе глубокого обучения (deep learning), например, технология Word2vec. Особенностью методов, использующих нейронные сети, является то, что они позволяют достичь качества, сравнимого с наилучшими современными методами (примерно 91% F-меры), но с минимальным набором дополнительной информации: признаков, токенов, словарных ресурсов.

### 1.2.9 Оценка качества

Рассмотрим несколько методов оценки результата работы классификатора. Для начала изучим вопрос на примере бинарных классов, то есть метка принимает значение 1 или 0. Допустим, что для класса 1 вероятность того, что модель угадала правильно равна  $p_x$ , для класса 0 эта вероятность равна  $1 - p_x$ . Можно подсчитать вероятность того, что классификатор "угадает" значение всех элементов:

$$P = \prod_{x \in 1} p_x \prod_{x \in 0} (1 - p_x)$$

Цель — максимизировать вероятность  $P$ . Работать с произведением неудобно, поэтому возьмем логарифм:

$$\ln(P) = \sum_{x \in 1} \ln(p_x) + \sum_{x \in 0} \ln(1 - p_x)$$

Обычно функцию потерь  $\ln(P)$  необходимо минимизировать, но, умножив эту функцию на минус один, получим функцию оценки качества Logloss.

$$\text{Logloss} = - \sum_{x \in 1} \ln(p_x) - \sum_{x \in 0} \ln(1 - p_x)$$

Можно заметить, что функцию оценки качества классификатора Logloss можно применять только при бинарной классификации. Также у наивных моделей часто возникают бесконечности, к тому же само число ничего конкретного не значит. Для реализации функционала оценки многокритериальных классификаторов существует другой набор метрик. При извлечении сущностей, атрибутов и отношений общепринятым способом рассчитываются оценки качества модели. Точность (Assurasy) — доля объектов с правильно предсказанным классом (ассурасу и другая рассматриваемая метрика precision переводятся на русский язык одинаково). Точность (Precision) — количество правильных ответов, делённое на количество всех найденных ответов. Полнота (Recall) — количество правильных ответов, делённое на общее число правильных ответов. Дополнительной метрикой оценки качества извлечения служит F-мера — соотношение между точностью и полнотой, чаще всего определяющееся как гармоническое среднее:

$$F = \frac{2RP}{R + P}$$

, где  $R$  — полнота,  $P$  — точность.

Введем еще два определения: ошибка первого рода (Type I error, false

positive, FP) — ситуация, когда модель бинарной классификации отнесла объект класса 0 к классу 1, ошибка второго рода (Type II 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), тогда:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

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

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

Рассмотрим следующую метрику — ROC-кривая. Пусть существует отсортированный по вероятности массив объектов. Необходимо решить, где его разделить на две части — все левые будут определены как класс 0, а правые как класс 1. Для каждого возможного разреза можно посчитать две метрики и нарисовать график зависимости одной от другой:

$$False\ positive\ rate = \frac{FP}{FP + TN} = \frac{FP}{size(0)}$$

$$True\ positive\ rate = \frac{TP}{TP + FN} = \frac{TP}{size(1)}$$

ROC-кривая начинается в точке (0, 0) и заканчивается в точке (1, 1). Между ними она должна сделать несколько шагов вверх и вправо. Можно идти слева направо по отсортированному списку объектов и каждый раз, когда попался объект класса 0, делать шаг вверх на  $\frac{1}{size(0)}$ , а когда встретился объект класса 1, делать шаг вправо на  $\frac{1}{size(1)}$ . Тогда в итоге линия обязательно придет из (0, 0) в (1, 1). Действительно, каждый раз когда один объект перемещался из класса 0 в класс 1, если его реальный класс равен 0, то TPR не изменился, а FPR увеличился на  $\frac{1}{size(0)}$ . Если его реальный класс равен 1, то FPR не изменился, а TPR увеличился на  $\frac{1}{size(1)}$ .

Кривая выступает хорошей визуализацией, однако хотелось бы иметь числовую метрику, которая бы говорила, насколько хорош получившийся из вероятностей порядок. Для это обычно берут площадь под этой кривой (Areas

Under Curve - AUC), её ещё называют ROC-AUC. Эта метрика не больше, чем 1, и чем выше и левее эта кривая, тем лучше. Практически всегда ROC AUC  $>$  0.5, иначе это легко поправить - надо просто перевернуть все вероятности. ROC AUC позволяет глубоко оценивать качество предсказанных вероятностей. Accuracy, Precision и Recall работали только с самими предсказаниями классов, и никак вероятности не затрагивали.

Обобщим метрики для модели с множеством классов. Есть два способа это сделать:

1) macro — это аналог One vs The Rest. Метрика просто считается для каждого класса независимо, а потом усредняется.

2) micro — более сложная, необходимо рассмотреть каждую пару "объект, класс" как объекты, и считать метрики на бинарной классификации в этой задаче.

## 2 Практическая часть

### 2.1 Инструменты разработки

Методы анализа текста применяются в первую очередь в машинном обучении, поэтому был выбран язык программирования Python3, включающий в себя большой набор мощных библиотек.

Scikit-Learn — расширение для библиотеки SciPy, предоставляющее высокую производительность. Расширение распространяется с открытым исходным кодом и допускает коммерческое использование, предоставляет доступ для многих моделей регрессии, классификации, кластеризации и уменьшения размерности, а также утилиты для перекрестной проверки и настройки гиперпараметров.

NLTK (Natural Language Tool-Kit) — пакет инструментов для обработки естественного языка. Содержит корпуса, лексические ресурсы, грамматику, алгоритмы обработки языков и предварительно обученные модели.

Rymorphy2 — морфологический процессор с открытым исходным кодом, предоставляет все функции полного морфологического анализа и синтеза словоформ. Процессор базируется на словарной морфологии и использует словарные данные проекта OpenCorpora. Словарь содержит около 250 тысяч лемм, а также является полностью открытым и регулярно пополняемым. Итоговый размер словаря составляет около 7 МБ

### 2.2 Анализ данных

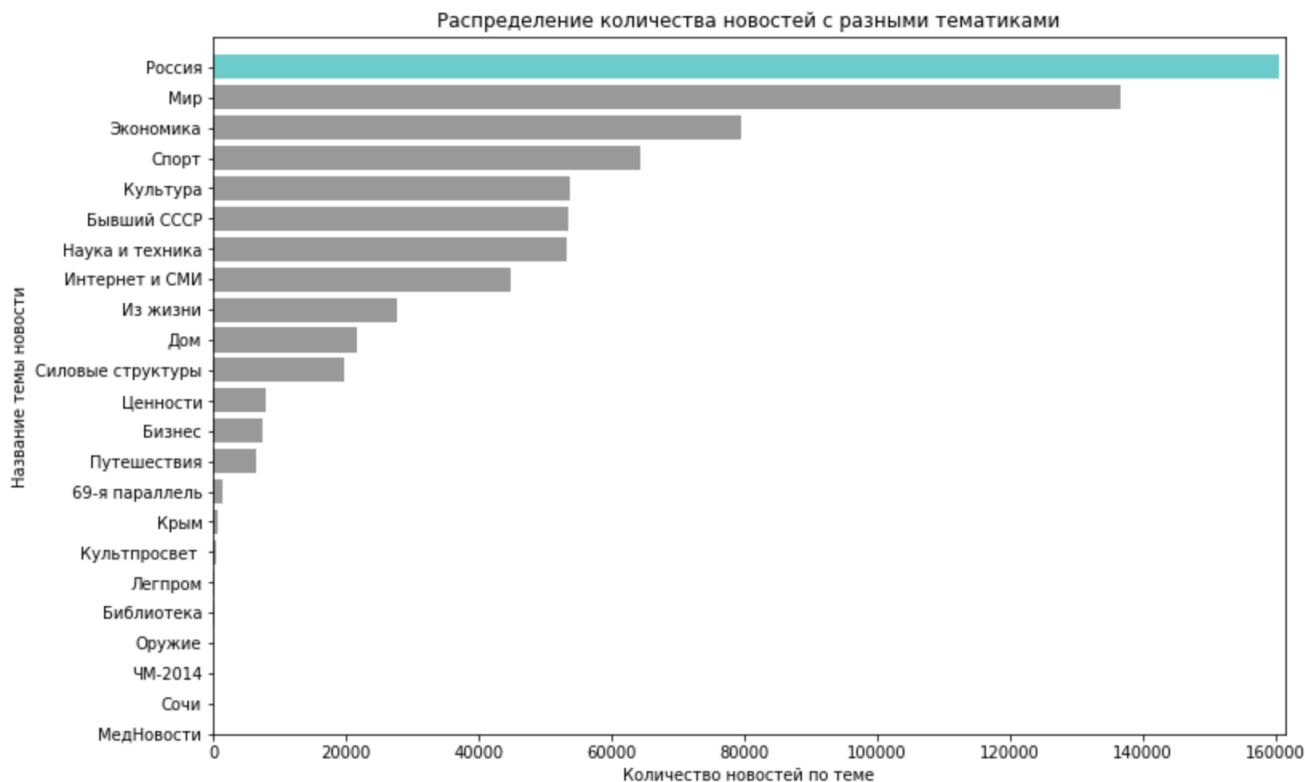
Для построения системы классификации текстов был выбран датасет с информацией о  $\approx 800$  тысяч новостей, взятых с сайта Lenta.ru. Первоначальные данные содержат ссылку на новость в исходном информационном источнике (url), заголовок с краткой информацией о содержании новости (title), полный текст (topic) и дату (date). Датасет размечен двумя способами: во-первых, указана тема (topic), а во-вторых тег (tag). Самая ранняя и поздняя дата публикации новости — 1914 год и декабрь 2019 года соответственно. Новости, датированные 1914 годом относятся к Первой мировой войне, дата остальных совпадает с датой публикации на информационном ресурсе.

Пример первичных данных:

url	title	text	topic	tags	date
<a href="https://lenta.ru/news/2018/01/26/ukranianpewds/">https://lenta.ru/news/2018/01/26/ukranianpewds/</a>	Популярный блогер перебрался в украинский инте...	Один из самых высокооплачиваемых YouTube-блогг...	Интернет и СМИ	Соцсети	2018/01/26
<a href="https://lenta.ru/news/2018/01/26/samolet/">https://lenta.ru/news/2018/01/26/samolet/</a>	Российский Су-34 перехватили над Балтикой	Истребители F-16 ВВС Дании, базирующиеся на ли...	Мир	Общество	2018/01/26
<a href="https://lenta.ru/news/2018/01/26/zolotoyorel/">https://lenta.ru/news/2018/01/26/zolotoyorel/</a>	Назван лучший российский фильм года	Лента Клима Шипенко «Салют 7» удостоена премии...	Культура	Кино	2018/01/26
<a href="https://lenta.ru/news/2018/01/26/what/">https://lenta.ru/news/2018/01/26/what/</a>	Российскую экономику назвали перспективнее аме...	Министр экономического развития Максим Орешкин...	Экономика	Госэкономика	2018/01/26
<a href="https://lenta.ru/news/2018/01/26/perepiska/">https://lenta.ru/news/2018/01/26/perepiska/</a>	В Германии рассказали о перехвате зашифрованных...	Федеральное ведомство по уголовным делам при М...	Интернет и СМИ	Киберпреступность	2018/01/26
<a href="https://lenta.ru/news/2018/01/26/stillalive/">https://lenta.ru/news/2018/01/26/stillalive/</a>	«Россия 24» по ошибке похоронила украинского п...	Телеканал «Россия 24» в эфире программы «Вести...	Интернет и СМИ	Пресса	2018/01/26

Целевой переменной для классификации могут выступать topic, tags. Заметим, что значений tags больше и они конкретнее, чем значения topic. Пример значений topic: Библиотека, Россия, Мир, Экономика, Спорт, тогда как в tags входит Музыка, Люди, Звери, Игры, Госэкономика, Гаджеты, Наука, Еда, Рынки, Деньги, Летние виды, Интернет и многое другое. Исходя из этого, целевой переменной для первичной модели можно взять tags, исключив topic из модели.

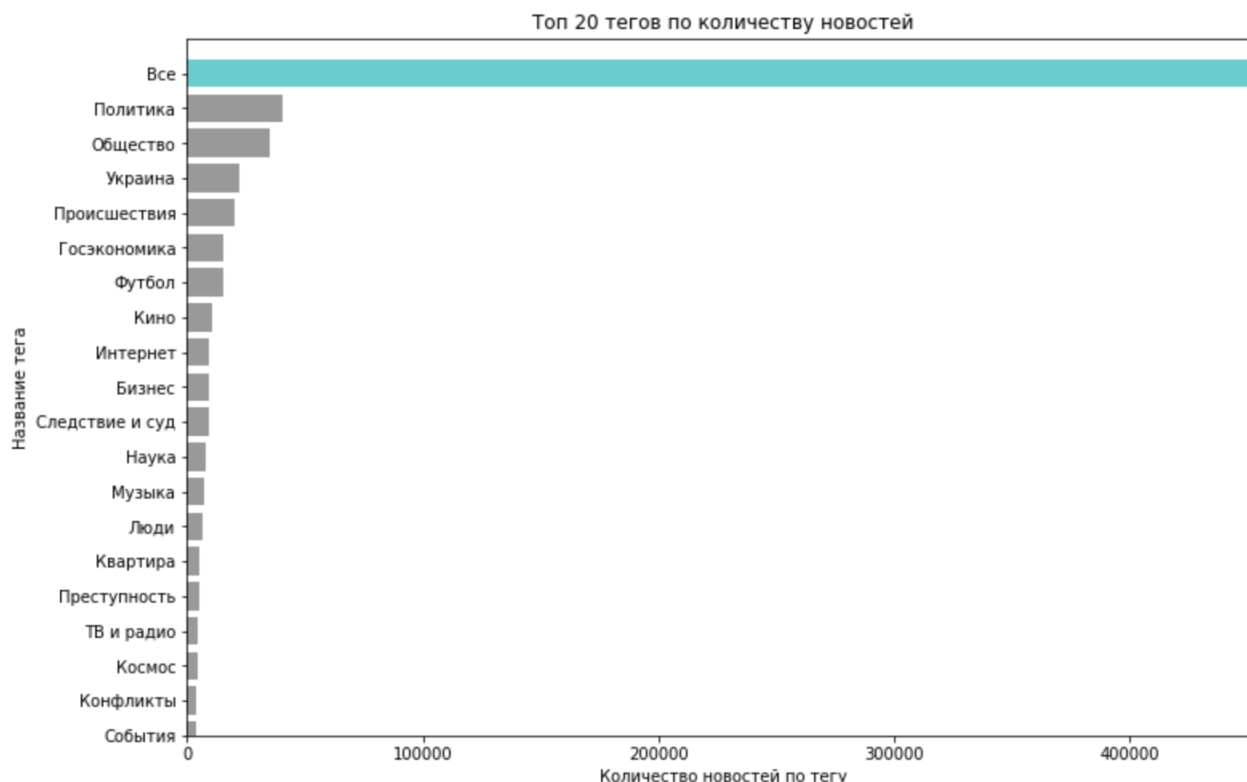
Из графика распределения количества новостей с различными темами видно, что лидирует тема "Россия", "Мир". Однако, новости с этими темами могут содержать информацию как о политике, так и о происшествиях, событиях. То есть информация, содержащаяся в topic недостоверна, поэтому её не стоит включать в модели.



Изучив распределение тегов, можно сделать вывод о том, что они подробнее описывают содержание новости. Выделяется тег "Все", которым отмечены



новости совершенно разных направленностей, поэтому соответствующие данные были удалены из датасета.



## 2.3 Работа с пропущенными значениями

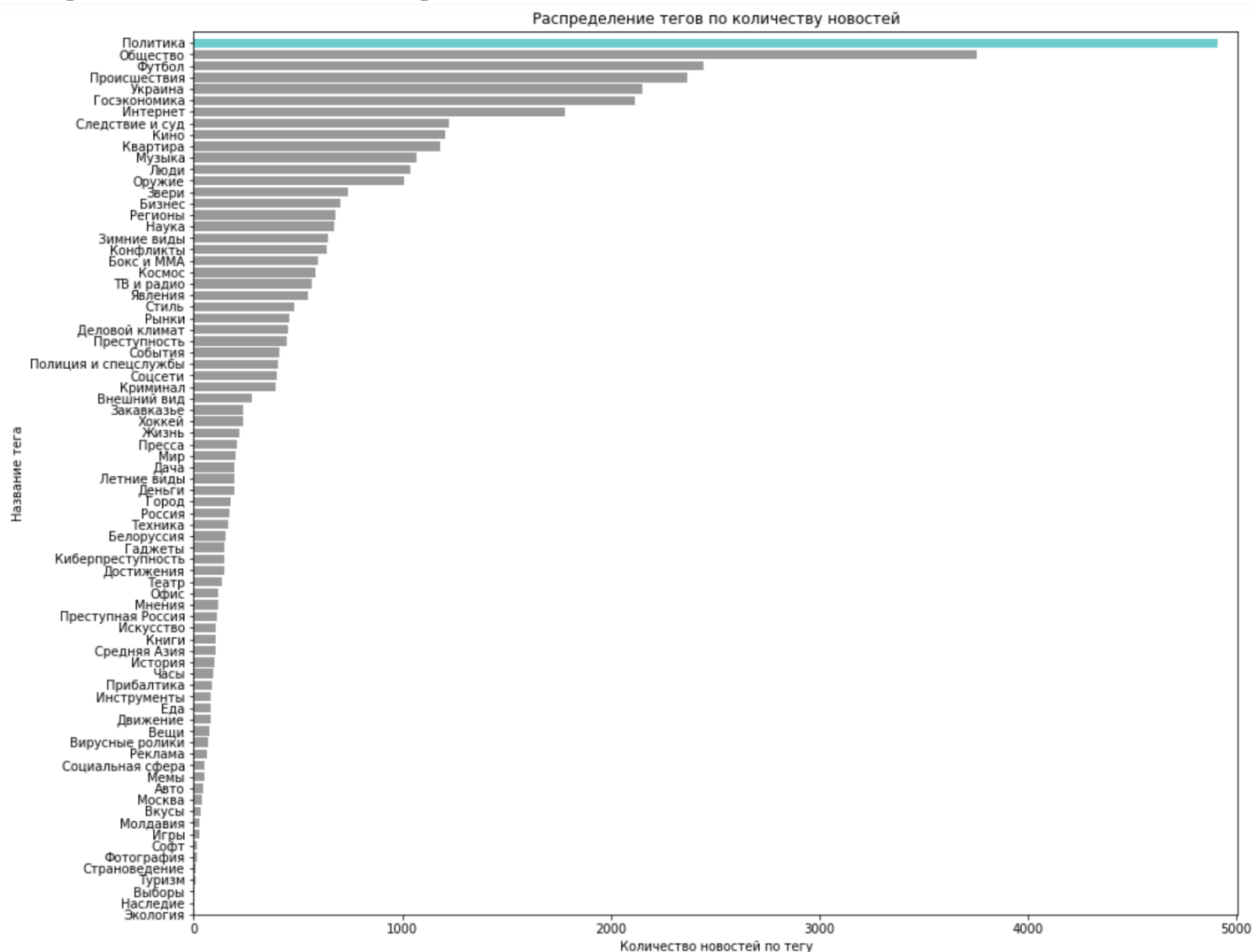
Проверим присутствие пропущенных значений в датасете, так как большинство алгоритмов NLP с ними не работают. Пропуски есть в колонках `topic`, `tags`, `text`. Эти колонки являются важными для моделей классификации, поэтому удалим новости, содержащие пропущенные значения.

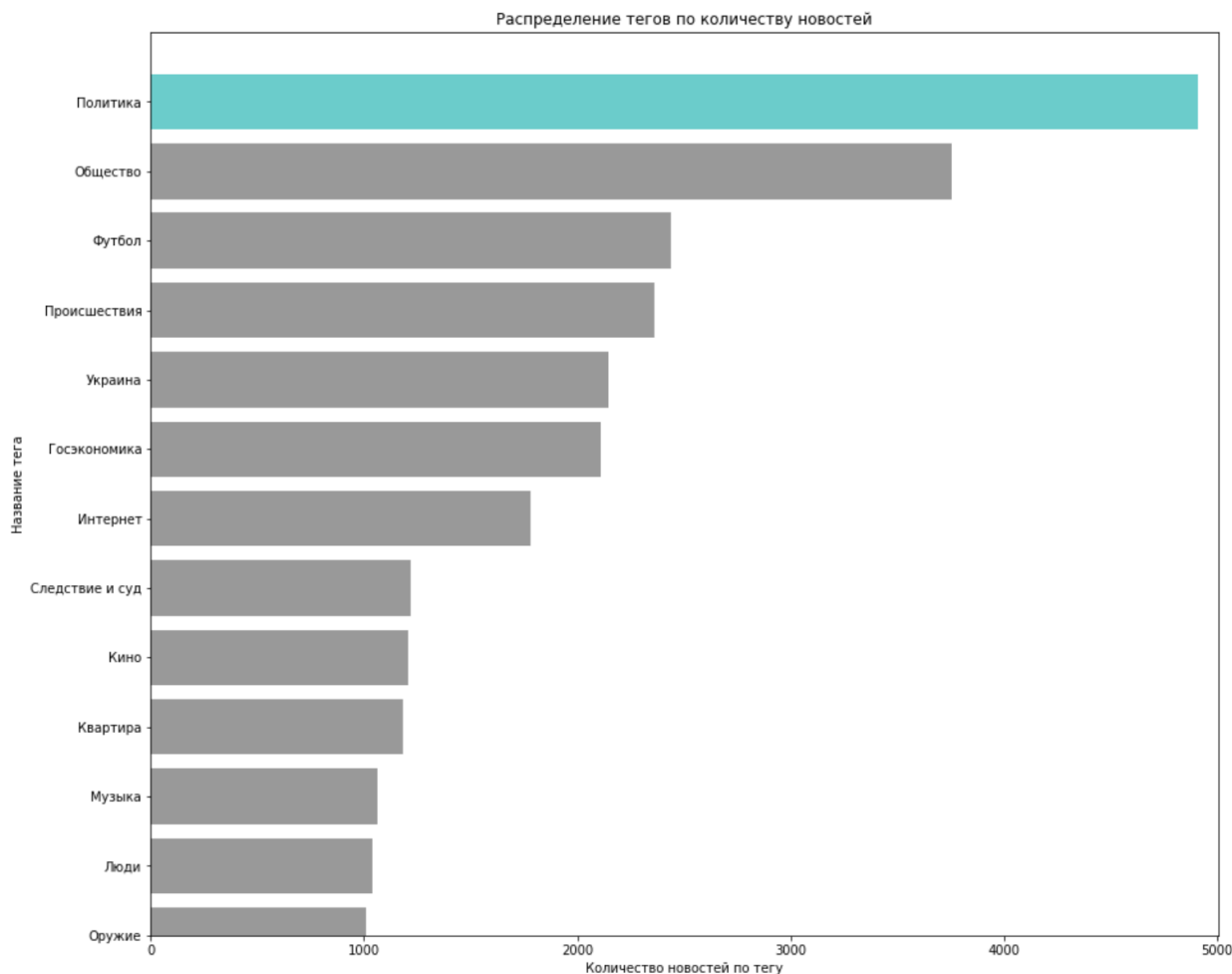
Количество пропущенных значений	
url	0
title	0
text	5
topic	62002
tags	27219
date	0

Также удалим колонку `url` с информацией о ссылке на новость, так как она не несет смысловой нагрузки для дальнейшей работы.

## 2.4 Предварительная обработка данных

Возьмем данные за 2018 год. Проверим распределение количества новостей по тегам. Из графика видны частотные теги: Политика, Общество, Футбол и мало-частотные: Экология, Наследие, Выборы. распределение получилось с тяжелым хвостом, поэтому для улучшения качества работы модели из данных были исключены новости с тегами, количество которых не превышало 1000. Распределение до и после обработки:





Разобьем заголовки новости и текст на токены, то есть на слова. В данной работе рассматривается наиболее гибкий инструмент для токенизации текста – регулярные выражения. Слова были приведены к нижнему регистру. Далее, из текста с помощью регулярного выражения извлекаются все слова, при этом удаляются цифры, знаки пунктуации, иностранные слова и символы. Из полученного массива удаляются стоп-слова. 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, содержащий 421 слово.

### 2.4.1 Лемматизация

После всех вышеперечисленных преобразований с помощью библиотеки `ru-morphy2` производится лемматизация оставшихся слов. По исходной словоформе и тегам выполняется поиск нормальной формы слова. Для анализа неизвестных слов в `RuMorphy2` используются несколько методов, которые применяются последовательно. Изначально от слова отсекается префикс из набора известных префиксов и если остаток слова был найден в словаре, то отсеченный префикс

приписывается к результатам разбора. Если этот метод не сработал, то аналогичные действия выполняются для префикса слова длиной от 1 до 5, даже если такой префикс является неизвестным. Затем, в случае неудачи, словоформа разбирается по окончанию. Для этого используется дополнительный автомат всех окончаний, встречающихся в словаре с имеющимися разборами. В процессе построения из автомата удаляются редкие окончания и разборы. Если слово имеет несколько вариантов разбора, то среди всех выбирается наиболее вероятный. Вероятности определяются по следующей формуле:

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

, где  $F_r(w)$  — количество раз, которое словоформа  $w$  встретила в корпусе  
 $F_r(w, t)$  — количество раз, которое эта словоформа встретила с тегом  $t$   
 $|R(w)|$  — число разборов, полученных от анализатора для словоформы  $w$

Продemonстрируем пример работы функции. Первоначальная новость:

' Фото опубликовано @chaif\_band Лидер группы «Чайф» Владимир Шахрин переделал припев своей песни «Аргентина — Ямайка 5:0». Свое творчество он опубликовал на странице музыкального коллектива в Instagram. «Уходит боль, уходит боль. Россия — Саудовская Аравия 5:0», — спел он, после чего назвал отечественную сборную «молодцами, порадовавшими старца». «Аргентина — Ямайка 5:0» — песня российской группы «Чайф» в стиле регги. Она посвящена разгромному поражению сборной Ямайки на чемпионате мира по футболу 1998 года. В 1999 году композиция получила премию «Золотой граммофон» как лучшая рок-песня. 14 июня сборная России обыграла команду Саудовской Аравии в матче-открытии чемпионата мира по футболу 2018 года. Встреча завершилась со счетом 5:0.'

После обработки:

'назвать рок-песня сборная разгромный ямайка старец аравия лидер поражение золотой хороший встреча опубликовать футбол завершиться счёт июнь граммофон саудовский спеть творчество группа регги страница чайф порадовать шахрина молодец обыграть владимир стиль команда матч-открытие аргентина коллектив премия уходит фото боль российский музыкальный россия отечественный чемпионат переделать получить посвятить композиция припев'

Итоговый датасет после лемматизации принял следующий вид:

	title	text	tags
0	{глаз, -летний, ровесница-немка, беженец, заре...	{афганец, изнасиловать, произойти, общаться,...	Общество
1	{кадыров, прочитывать, новогодний, стихотворение}	{чеченский, напомнить, великий, присвоить, дел...	Политика
2	{поддержать, иран, протестовать, белый, дом}	{общаться, демонстрант, популярный, решение, ц...	Политика
3	{пожар, возможный, причина, южно-сахалинск, ёл...	{прибыть, пресс-служба, пожарный, место, южно-...	Общество
4	{минимальный, труд, россиянин, размер, оплата,...	{закон, минимальный, путин, процент, минтруд, ...	Госэкономика
...	...	...	...
26224	{раскольник, глава, митрополит, украинский, уп...	{раскольник, пётр, общаться, решение, проход...	Украина
26225	{полиция, кордон, марсельеза, жилет, жёлтый, п...	{столкновение, гарни, мусорный, отставка, эмм...	Происшествия
26226	{украина, разбиться, су-}	{лётчик, пресс-служба, сша, произойти, погибну...	Украина
26227	{петербург, квартира, комната, насрать}	{квадратный, место, жизнь, выставить, россия, ...	Квартира
26228	{рэп, протест, путин, наркотик, секс, порассуж...	{отмена, искусство, матвиенко, россия, секс, п...	Политика

26229 rows × 3 columns

## 2.4.2 Кодирование категориальных данных

Целевой переменной для моделей классификации был выбран столбец `tags`. Можно заметить, что в нём содержатся данные строкового типа, что не подходит для реализации алгоритмов. С помощью функции `LabelEncoder` из библиотеки `sklearn` было проведено кодирование признаков, то есть перевод их в числовые значения. Результат преобразования представлен ниже:

	title	text	tags
0	{ 'глаз', '-летний', 'ровесница-немка', 'бежене...	{ 'афганец', 'изнасиловать', 'произойти', 'сооб...	6
1	{ 'кадыров', 'прочитать', 'новогодний', 'стихот...	{ 'чеченский', 'напомнить', 'великий', 'присвои...	8
2	{ 'поддержать', 'иран', 'протестовать', 'белый'...	{ 'общаться', 'демонстрант', 'популярный', 'реш...	8
3	{ 'пожар', 'возможный', 'причина', 'южно-сахали...	{ 'прибыть', 'пресс-служба', 'пожарный', 'место...	6
4	{ 'минимальный', 'труд', 'россиянин', 'размер', '...	{ 'закон', 'минимальный', 'путин', 'процент', '...	0
...	...	...	...
26224	{ 'раскольник', 'глава', 'митрополит', 'украинс...	{ 'раскольник', 'пётр', 'сообщаться', 'решение'...	11
26225	{ 'полиция', 'кордон', 'марсельеза', 'жилет', '...	{ 'столкновение', 'гарни', 'мусорный', 'отстав...	9
26226	{ 'украина', 'разбиться', 'су-' }	{ 'лётчик', 'пресс-служба', 'сша', 'произойти', '...	11
26227	{ 'петербург', 'квартира', 'комната', 'наслать' }	{ 'квадратный', 'место', 'жизнь', 'выставить', '...	2
26228	{ 'рэп', 'протест', 'путин', 'наркотик', 'секс'...	{ 'отмена', 'искусство', 'матвиенко', 'россия', '...	8

26229 rows × 3 columns

## 2.4.3 TF-IDF преобразование

Для обучения моделей необходимо получить численные значения из строковых данных. Был использован алгоритм TF-IDF.

	абашидзе	аббас	абдулатипов	абитуриент	абнуть	абонемент	абордаж	абориген	аборт
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

5 rows × 18704 columns

Строчки матрицы соответствуют конкретной новости, индекс которой указан в начале строки. Столбцы соответствуют словам, встреченным в корпусе, состоящем из всех новостей использованного датасета. Значения — коэффициенты TF-IDF. Матрица будет считаться разреженной, то есть матрицей с преимущественно нулевыми элементами. Стоит отметить, что первые три столбца в матрице — вовсе не ошибка, а фамилии политических деятелей, фигурирующих в тексте.

## 2.5 Разбиение выборки на обучающую и тестовую

В любой задаче машинного обучения набор данных разбивается на две выборки: обучающую (train) и тестовую (test). Модель обучают на train выборке, а на test выборке модель проверяют, то есть получают некоторое предсказание. Это необходимо для того, чтобы оценить качество работы алгоритма на уже имеющихся данных и избежать переобучения/недообучения модели.

Существуют несколько способов разбиения данных на выборки. В работе используется разбиение функцией `train_test_split` из библиотеки Sklearn. Функция разбивает данные рандомным способом, что не всегда бывает корректным методом разбиения, но на данном этапе разработки этот метод подходит.

## 2.6 Итоги работы классификаторов

Algorithm name	Accuracy	Time
SGDClassifier()	71%	1 min.
LinearSVC	71%	2 sec.
LogisticRegression()	69%	3 min.
PassiveAggressiveClassifier()	67 %	3 min.
RandomForestClassifier()	65%	8 min.
DecisionTreeClassifier()	57%	8 min.
MultinomialNB()	55%	1 sec.

## Заключение

## Список использованных источников

- [1] Applied Text Analysis with Python by Benjamin Bengfort, Rebecca Bilbro, and Tony Ojeda (O'Reilly). 9781491963043
- [2] Автоматическая обработка текстов на естественном языке и анализ данных : учеб. пособие / Большакова Е.И., Воронцов К.В., Ефремова Н.Э., Клышинский Э.С., Лукашевич Н.В., Сапин А.С. — М.: Изд-во НИУ ВШЭ, 2017. — 269 с. ISBN 978-5-9909752-1-7
- [3] МЕТОДЫ И МОДЕЛИ АВТОМАТИЧЕСКОГО ИЗВЛЕЧЕНИЯ КЛЮЧЕВЫХ СЛОВ С.О. Шереметьева, П.Г. Осминин Вестник ЮУрГУ. Серия «Лингвистика». 2015. Т. 12, No 1. С. 76–81
- [4] <https://countwordsfree.com/stopwords/russian>
- [5] Automated News Categorization using Machine Learning methods U Suleymanov1 and S Rustamov2,3 Applying Machine Learning Algorithms for News Articles Categorization: Using SVM and kNN with TF-IDF Approach A Comparative Analysis Of News Categorization Using Machine Learning Approaches Nabamita Deb, Vishesh Jha, Alok K Panjiyar, Roshan Kr Gupta



# Приложение

Исходный код можно найти по [ссылке](#).