УДК 004.021

Миляев Дмитрий Романович
аспирант
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»
Профессора Попова ул., 5, Санкт-Петербург, 197376, Россия
e-mail: milyaev.dmitry00@mail.ru

ФИЛЬТРАЦИЯ ТЕКСТОВЫХ ДАННЫХ С ПОМОЩЬЮ МЕТОДА TF-IDF

Статья посвящена проблеме поиска эффективного алгоритма фильтрации текстовых сообщений. В частности, рассматривается фильтрация сообщений из соцсетей. В первой части статьи приведено теоретическое описание алгоритма. Далее рассмотрен конкретный пример использования метода. Статья завершается описанием достоинств и недостатков технологии**.**

The article is devoted to the problem of finding an effective algorithm for filtering text messages. In particular, filtering messages from social networks is being considered. The first part of the article provides a theoretical description of the algorithm. The following is a specific example of using the method. The article concludes with a description of the advantages and disadvantages of the technology.

Ключевые слова: Неструктурированные данные, фильтрация данных, алгоритм обработки, информация, данный из соцсетей.

Keywords: Unstructured data, data filtering, processing algorithm, information given from social networks.

В современном мире объемы текстовых данных растут экспоненциально: социальные сети, электронная почта, чат-боты и форумы ежедневно генерируют огромное количество информации. В таких условиях автоматическая обработка и фильтрация текстов становятся критически важными задачами. Одним из эффективных методов анализа и ранжирования текстов является TF-IDF (Term Frequency-Inverse Document Frequency), который позволяет оценивать значимость слов в документе относительно всей коллекции текстов.

TF-IDF широко применяется в задачах классификации сообщений, поиска релевантных документов и даже в борьбе со спамом. Этот метод сочетает простоту реализации с высокой эффективностью, что делает его популярным инструментом в машинном обучении и обработке естественного языка. В данной работе мы рассмотрим принципы работы TF-IDF, его преимущества и применение для фильтрации текстовых сообщений. Для большей конкретики будет описан пример обработки массива сообщений с целью выявления жалоб.

Теоретически алгоритм состоит из следующих этапов:

1) Сбор и разметка данных

Требуется составить датасет сообщений, где каждое сообщение помечено как: Жалоба (1) или Не жалоба (0).

Примеры жалоб: "Ваш сервис ужасен, ничего не работает!", "Почему вы не отвечаете на мои обращения?"

Примеры не-жалоб: "Спасибо, всё отлично!", "Как пользоваться этой функцией?"

2) Создание эталонного вектора жалобы

Необходимо каждую размеченную жалобу векторизовать (класс 1) через TF-ID. Усредняем все векторы:

$$\text{Эталон} = \frac{1}{N}\sum_{i=1}^{N} \text{Вектор}(\text{Жалоба}_i)$$

3) Предобработка текста

Каждое сообщение проходит очистку:

- Токенизация (разбивка на слова)
- Лемматизация (приведение слов к начальной форме: "ужасен" → "ужасный")
- Удаление стоп-слов ("и", "но", "в")
- Очистка от спецсимволов (эмодзи, знаки препинания)

4) Векторизация текста

Чтобы сравнивать сообщения математически, переведём их в числа методом TF-IDF. Каждому слову присваивается вес, показывающий его важность. Формула для слова t в документе d:

TF-IDF(t,d) = TF(t.d)*IDF(t), Где:

TF(t,d) – частота слова t в документе d

$IDF(t)=\log(\frac{N}{\text{Число документов с t}})$ – обратная частота в корпусе

5) Фильтрация через косинусное сходство

Для сообщения msg вычисляем косинусное сходство с эталоном:

$$\cos = \frac{\text{Вектор}(\text{msg}) * \text{Эталон}}{\|\text{Вектор}(\text{msg})\| * \|\text{Эталон}\|}$$

Если $\cos \geq 0{,}7 \rightarrow$ это жалоба

6) Проверка по ключевым словам

Список триггерных слов: "жалоба", "претензия", "недоволен", "верните деньги", "ужасный сервис". Если есть хотя бы одно - повышаем вес.

Итоговый алгоритм

1. Предобработка сообщения (очистка, лемматизация).
2. Векторизация (TF-IDF).
3. Сравнение с эталоном (косинусное сходство).
4. Дополнительные проверки (ключевые слова).
5. Пороговая фильтрация

Оценка качества

Метрики для определения эффективности: Accuracy (общая точность), Precision (сколько найденных жалоб — реальные жалобы), Recall (сколько реальных жалоб найдено).

Для повышения эффективности можно использовать следующие методы: подбор порогов, увеличение датасета.

Далее разберём конкретный пошаговый пример TF-IDF для сообщения "У вас ужасный сервис".

Предварительная работа - Подготовка обучающей выборки:

- "у вас ужасный сервис" (жалоба)
- "спасибо за помощь" (не жалоба)
- "всё работает отлично" (не жалоба)
- "претензия к качеству" (жалоба)
- "сервис стал хуже" (жалоба)

Уникальные слова (словарь):["вы", "ужасный", "сервис", "спасибо", "помощь", "всё", "работает", "отлично", "претензия", "качество", "стал", "хуже"]

Предобработка текста

Исходное сообщение: "У вас ужасный сервис"

Токенизация: ["у", "вас", "ужасный", "сервис"]

Удаление стоп-слов ("у" — стоп-слово): ["вас", "ужасный", "сервис"]

Лемматизация:

- "вас" → "вы"
- "ужасный" → "ужасный" (уже в начальной форме)
- "сервис" → "сервис"

Итоговые токены: ["вы", "ужасный", "сервис"]

Расчёт TF

Для нашего сообщения ["вы", "ужасный", "сервис"]:

- Количество слов: 3
- TF для каждого слова:
  - "вы": 1/3 ≈ 0.33
  - "ужасный": 1/3 ≈ 0.33
  - "сервис": 1/3 ≈ 0.33

Расчёт IDF

Считаем для каждого слова:

Слово "вы" встречается в документах: 1, 5 (например, "сервис стал хуже" → "стал хуже у вас")

IDF = log(5/2) ≈ 0.92

Слово "ужасный" встречается только в документе 1.

IDF = log(5/1) ≈ 1.61

Слово "сервис" встречается в документах: 1, 4, 5

IDF = log(5/3) ≈ 0.51

Итоговый TF-IDF вектор - умножаем TF на IDF для каждого слова:

"вы": 0.33 * 0.92 ≈ 0.30

"ужасный": 0.33 * 1.61 ≈ 0.53

"сервис": 0.33 * 0.51 ≈ 0.17

Вектор для сообщения:[0.30, 0.53, 0.17, 0, 0, 0, 0, 0, 0, 0, 0, 0] (Остальные слова словаря имеют вес 0, так как их нет в сообщении.)

Сравнение с эталоном жалобы

Допустим, эталонный вектор жалобы (усреднённый по размеченным данным) выглядит так: [0.25, 0.60, 0.40, 0, 0, 0, 0, 0, 0.30, 0.20, 0, 0.10]

Вычисляем косинусное сходство:

$$\cos = \frac{\text{Вектор}(msg) * \text{Эталон}}{\|\text{Вектор}(msg)\| * \|\text{Эталон}\|}$$

где:

- Вектор сообщения [0.30, 0.53, 0.17, ...],
- Эталон [0.25, 0.60, 0.40, ...].

Числитель:

0.30×0.25+0.53×0.60+0.17×0.40=0.075+0.318+0.068=0.4610.30×0.25+0.53×0.60+0.17×0.40=0.075+0.318+0.068=0.461

$$\|A\| = \sqrt{0.30^2 + 0.53^2 + 0.17^2} = 0.63$$

$$\|B\| = \sqrt{0.25^2 + 0.60^2 + 0.40^2 + 0.30^2 + 0.20^2 + 0.10^2} = 0.85$$

Итог

$$cos = \frac{0.461}{0.63 * 0.85} = 0.86$$

Порог: 0.7

Вывод: $0.86 > 0.7 \rightarrow$ это жалоба

Метод TF-IDF (Term Frequency-Inverse Document Frequency) является мощным инструментом для фильтрации и анализа текстовых сообщений. Благодаря своей способности оценивать значимость слов в контексте документа и всей коллекции текстов, он находит применение в задачах классификации, поиска релевантной информации, антиспам-фильтрации и других областях обработки естественного языка.

Ключевые преимущества TF-IDF включают простоту реализации, интерпретируемость результатов и эффективность работы даже на небольших наборах данных. Однако у метода есть и ограничения: он не учитывает семантические связи между словами, порядок их следования и контекстную зависимость. Тем не менее, в сочетании с другими алгоритмами машинного обучения (например, с методами векторизации Word2Vec или BERT) TF-IDF остается актуальным и полезным инструментом.

Таким образом, фильтрация сообщений с помощью TF-IDF – это эффективный подход для предварительной обработки текстов, который может значительно улучшить качество работы систем автоматической классификации и поиска информации. Дальнейшее развитие методов текстовой аналитики, вероятно, будет сочетать TF-IDF с более сложными моделями, что позволит добиться еще большей точности в обработке естественного языка.

СПИСОК ЛИТЕРАТУРЫ

1) Analyzing Documents with TF-IDF [электронный ресурс] URL: https://programminghistorian.org/en/lessons/analyzing-documents-with-tfidf (дата обращения 10.09.25).

2) Альберт Яковлев, Д. О. Соколова. Цифровая фильтрация и синтез цифровых фильтров. Новосибирский государственный технический университет, 2012. – 64 с.

3) TF-IDF в SEO: что это и как его использовать?[электронный ресурс] URL: https://journal.topvisor.com/ru/practice/tf-idf/ (дата обращения 11.09.25).

4) Что такое TF-IDF (частота термина — обратная частота документа)[электронный ресурс] URL: https://www.geeksforgeeks.org/machine-learning/understanding-tf-idf-term-frequency-inverse-document-frequency/ (дата обращения 12.09.25).