(cc) BY

Шицелов Анатолий Вячеславович
магистр, инженер
Центр прогнозирования и развития
ФГБОУ ВО «Югорский государственный университет»
г. Ханты-Мансийск, ХМАО – Югра

ОБЩИЙ АЛГОРИТМ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ЕСТЕСТВЕННОГО ЯЗЫКА

Аннотация*: в статье представлен общий подход к базовой обработке текстов на естественном языке. Рассмотрен сценарий обработки исходно текста на русском языке для получения массива токенов, с которыми можно в дальнейшем работать как с классами.*

Ключевые слова*: естественный язык, машинное обучение, алгоритм.*

Введение

Существует множество задач, связанных с обработкой и в частности пониманием текстов, написанных на естественном языке, но эти тексты в исходном виде мало пригодны для применения на них алгоритмов машинного обучения, для решения этой проблемы необходимо привести текст на естественном языке в некий стандартный вид с ограниченным набором слов. Такое преобразование позволит применить статистические модели и модели из машинного обучения и нейронных сетей для обработки исходных тестов.

В задаче предварительной обработки текста есть несколько проблем которые необходимо решить:

1. В русском языке одно и тоже слово может писаться по-разному в зависимости от контекста.

2. В тексте могут встречаться орфографические ошибки.

3. В естественном языке есть служебные слова, которые не несут смысла.

4. В тестах есть редко встречаемые слова и часто встречаемые слова и они создают проблемы: в первом случае – с ними будет мало примеров для работы алгоритмов машинного обучения, а во втором случае они будут создавать шум.

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

Общий подход к обработке теста на естественном языке

Весь алгоритм можно разбить на несколько этапов:

1. Выделение слов в тесте. В исходном тесте помимо слов могут быть и другие объекты, например, числа и знаки препинания. Так же стоит отметить, что иногда стоит сохранять структуру текста (разбиение на предложения и абзацы). На этом этапе целесообразно применять регулярные выражения для разбиения теста на части.

2. Удаление ненужных слов. В естественном языке существует набор служебных слов, называемых стоп-словами, их необходимо удалять и полученного на шаге 1 массива.

3. Создание словаря. К каждому слову в тесте необходимо пременить следующие шаги:

– нормализовать слово (привести в нормальную форму);

– используя алгоритм нечеткого сравнения строк проверить наличие текущего слова в словаре;

– если слова нету в словаре, то добавить его и установить количество встреч этого слова в 0;

– увеличить счетчик количества встреч текущего слова на один.

4. Удалить из словаря все часто встречаемые слова и все редко встречаемые слова. Границы для удаления слов необходимо подбирать опытным путем.

После этих шагов на выходе будет два объекта: массив токенов (нормализованых слов) это бывшей исходный текст и словарь, который можно использовать для новых текстов без необходимости повторно проходить шаг 2, 3 и частично 1.

Отдельно стоит отметить алгоритм нечеткого сравнения строк. Его можно реализовать, объединив 2 алгоритма: триграммный поиск и расстояние Левенштейна. Триграммный поиск стоит первым и выбирает из всего словаря, некоторую малую часть похожих слов, для уточнения результата можно использовать расстояние Левенштейна [1].

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)

Далее в зависимости от задачи, может потребоваться представить набор слов как токенов в разном формате:

5. В виде самих токенов – это в основном алгоритмы LDA, LSA и им подобные.

6. В виде числового вектора – это когда каждому слову сопоставлен некоторый вектор, полученный с помощью таких алгоритмов как Word2Vec. При таком подходе сохраняется семантика слова, что позволяет выявлять синонимы, антонимы и другие подобные свойства слов. Однако у такого подхода есть и несколько минусов: для сопоставления слов и вектором необходим очень большой набор примеров для обучения, в силу особенностей работы алгоритма Word2Vec при каждом обучении вектора будут случайными из этого вытекает еще одна особенность – в данном подходе имеют значение только расстояние между словами (их векторами), а не сами позиции векторов, так как было сказано ранее вектора изначально создаются случайно, но в ходе обучения стягиваются друг к другу, поэтому расположение векторов при каждом обучении будет новым, но отношения между словами сохраняются.

Список литературы

1. Шицелов А.В. Система автоматической кластеризации текстов с применением искусственных нейронных сетей / А.В. Шицелов, В.В. Славский, В.В. Бурлуцкий // МАК: Математики – Алтайскому краю: Сборник трудов всероссийской конференции по математике. – 2017. – С. 136–140.

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)