Использование векторного представления текста для решения задачи определения категории товара.

Александр Широков

ПМ-1701

Задачи обработки текстов

- Информационный поиск (Informational retrieval): найти релевантные документы
- Анализ тональности (Sentiment analysis): определить позитивное или негативное отношение несёт текст;
- Языковые модели (Language models): по заданному отрывку текста предсказать следующее слово или символ;
- Распознавание именованных сущностей (Named entity recognition): имена, географические объекты и.т.д;
- Морфологическая сегментация (Morphological segmentation) - разделить слова на морфемы (приставки, суффиксы);

Предобработка текста

Первичная

- Токенизация
- Удаление лишних символов
  - большие буквы, слова-архаизмы
- Удаление стоп-слов
  - “будто”, “наконец”, ...
- Лемматизация

```
>> стали -> [(стать, 0.97),
             (сталь, 0.03)]
```

- Стэмминг

```
>> просвещения -> просвещён
```

Интеллектуальная

- Разбиение слова на сегменты
- Исправление опечаток
- Исправление сокращений
- Исправление одинаково звучащих слов

Разбиение слова на сегменты

- Алгоритм максимального соответствия
  ```
  >> input: новаявеличинарусскойпоэтическойреальности
  >> output: новая величина русской поэтической реальности
  ```

- Алгоритм обратного максимального соответствия
  ```
  >> input: новаявеличинарусскойпоэтическойреальности
  >> output: новая величина русской по эти чес кой реальности
  ```

- Двунаправленный алгоритм максимального соответствия

- Рекуррентная максимизация вероятности первого слова

- Реализация алгоритма выбора наиболее вероятной подпоследовательности с помощью перемножения вероятностей биграмм

Исправление опечаток

- Алгоритм **Питера Норвига** + N-gram модель
  - Для слова w необходимо найти наиболее вероятную правку $c = correct(w)$.
    - Находим всех кандидатов c, которые достаточно близки к w
    - Выбираем *наиболее вероятный* из них
  - Расстояние Левенштейна – минимальное необходимое количество удалений, перестановок, вставок и замен символов, необходимых, чтобы одно слово превратить в другое

Пары:	ø+камн	к+амн	ка+мн	кам+н	камн+ø	(a, b) пары
Удаления:	ø+амн	к+мн	ка+н	кам+ø		Удаление первой буквы в b
Перемена мест:	ø+акмн	к+ман	ка+нм			Перемена мест двух первых букв b
Замена:	ø+?амн	к+?мн	ка+?н	кам+?		замена буквы в начале b
Вставка:	ø+?+камн	к+?+амн	ка+?+мн	кам+?+н	камн+?+ø	Вставка буквы между a и b

```
>> input: Я бы не хоетл задаваться вопрсом. Чот ты здес длаешь?
>> output: Я бы не хотел задаваться вопросом. Что ты здесь делаешь
```

Исправление сокращений

- Алгоритм
  - Найдём словаре (либо префиксном дереве) N-граммы, начинающиеся с данного сокращения слова
  - Возьмём наиболее вероятную

```
>> input: салфетки бумажные в/уп 5 шт.
>> output: салфетки бумажные (вакуумная упаковка) 5 штук

>> input: Подуш"ШокоZAVR"молоч. Шок240гр
>> output: подушка шоколад ZAVR молочный шоколад 240 грамм
```

Data Fusion Contest - Goodsification

- **Задача:** по текстовому описанию чека определить категорию товара

- **Исходные данные:**
  - Более 8 миллионов уникальных чеков
  - 96 категорий товара
  - Метрика: WEIGHTED F1-SCORE

- 0 - алкогольная продукция
- 1 - презервативы
- 2 - сигареты
- 3 - стики, нагреваемые табачные палочки
- 4 - автомобильная лампа
- 6 - различные приборы чистки для авто: щётки, стеклоочистители
- 7 - газовые колонки, топливо, заправка
- 9 - дизельное топливо
- 11 - щётка, услуги автомойки
- 12 - масла, смазки, ароматизаторы, антифризы для автомобилей
- 13 - ремонт машин
- 19 - канцелярские принадлежности, бумага
- 20 - газеты, журналы
- 24 - детская литература, книжки для малышки
- 26 - органайзеры, ножницы, листки, закладки, корректоры

Векторное представление текста: подходы

1. TF-IDF + SVM
   - чем чаще слово встречается в документе, тем оно важнее;
   - чем реже слово встречается в других документах, тем оно важнее;
2. Word2Vec + KNN
   - Получение эмбеддингов слова – векторного представления слова
   - Каждый чек – среднее эмбеддингов слов, в него входящих
3. FastText + KNN
   - Попытка выучить не эмбединги слов, а эмбединги предложения

Сравнение результатов с предобработкой

- Исходный словарь встречаемости слов – обучен на миллионе статей Wikipedia

F1 - SCORE	TF-IDF + SVM Classifier	Word2Vec + k-Nearest Neighbors	FastText + k-Nearest Neighbors
Без предобработки	0.817	0.778	0.834
Разбиение на сегменты	0.824	0.811	0.8455
Исправление опечаток	0.834	0.815	0.8478
Поиск сокращений	0.825	0.808	0.8433
Полная предобработка	0.844	0.8203	0.8501

Выводы

- Перед обучением текстовых моделей весьма желательно тщательно обработать текст
- 3-е место среди публичных решений / бронзовая медалька на boosters

Источники:

- сравнение результатов предобработки текста
- решение соревнования с описанием методов

Спасибо за внимание!

github.com/aptmess