Анализ тональностей в русском языке

А. Лапидус, А. Кузнецова

31 марта 2018

Что у нас было

- Словарь тональностей Лукашевича и Четверкина
- Отзывы о книгах и фильмах с imhonet
- Предобученные эмбединги
- Пакет Socialsent с методами распространения тональностей по лексиконам.

Статья

Inducing Domain-Specific Sentiment Lexicons from Unlabeled Corpora (William L. Hamilton, Kevin Clark, Jure Leskovec, Dan Jurafsky, 2014)

Что мы хотели сделать?

- Проверить, работает ли алгоритм SentProp (sentiment propagation) описанный в статье для русского языка.
- Вывести словарь тонально-окрашенных слов для специфических корпусов текстов (отзывы на фильмы, отзывы на книги);
- Построить классификатор текстов с тональными лексиконами, выделенными с помощью SentProp и сравнить его с классификаторами, основанными на других методах.

Как работает SentProp?

Выделение тональных слов по контекстам

```
FILMS_POSITIVE = ['душевный', 'замечательный', 'достойный', 'добрый', 'веселый', 'неординарны
                  'красивый', 'гениальный', 'суперский', 'офигенный']
FILMS_NEGATIVE = ['скучный', 'скучища', 'посредственный', 'третьесортный', 'испортить'
                  'омерзительный', 'стрёмный', 'разочаровать', 'неинтересный',
                  'отвратительный']
```

Распространение тональностей

Для распространения тональностей используются небольшие словари окрашенных слов (seed sets). Мы вывели тональные слова из наших корпусов, выделяя конектсы.

Распространение тональностей

Построение лексического графа на основе косинусной близости слов, где слово соединяется ребрами с k ближайшими словами корпуса.

Random Walk

Тональности слов из seed sets переносятся по графу с использованием значений косинусной меры близости в качестве переходных вероятностей. По мере распространения по графу значения тональнотси постепенно затухают. Таким образом, мы получаем положительные и отрицательные оценки (scores) для слов в лексическом графе.

Алгоритм распространения

Составление матрицы эмбедингов по корпусу текстов.

```
]: #какая строчка из матрицы соответствует каждому слову из словаря
   embedding_matrix = np.zeros((len(word_index), 300))
   for word, i in word_index.items():
       embedding_vector = embeddings_index.get(word)
       if embedding_vector is not None:
           # words not found in embedding index will be all-zeros.
           embedding_matrix[i-1] = embedding_vector
```

```
]: embeddings = embedding.Embedding(embedding_matrix, vocabular, normalize=True)
```

Распространение полярностей методами из пакета Socialsent

```
polarities = polarity_induction_methods.random_walk(embeddings, POSITIVE_PLOT, NEGATIVE_PLOT)
```

Результаты распространения

5247	0.976134	ржавый
2464	0.976232	берри
2154	0.979769	ширвиндт
7794	0.983591	замечательный
736	0.987786	достойный
6190	0.988380	душевный
7817	0.993000	добрый
4804	0.994345	суперский
3346	0.998497	гениальный

MAX: гениальный 0.998497 MIN: скучный 0.000067

Результаты распространения

6059	0.000382	представитель
6396	0.000362	скотт
6462	0.000492	отодрать
6493	0.000458	выхватывать
6581	0.000487	свалиться
6606	0.000067	скучный
6621	0.000348	плагиатор

Алгоритм плохо выделяет негативные слова.

Классификаторы

- SVM на бинарных векторах BOW
- SVM на delta tfidf
- SVM на тональных словах

SVM на бинарных векторах

- бинарная классификация (positive/negative)
- positive/negative/neutral

SVM на delta tfidf

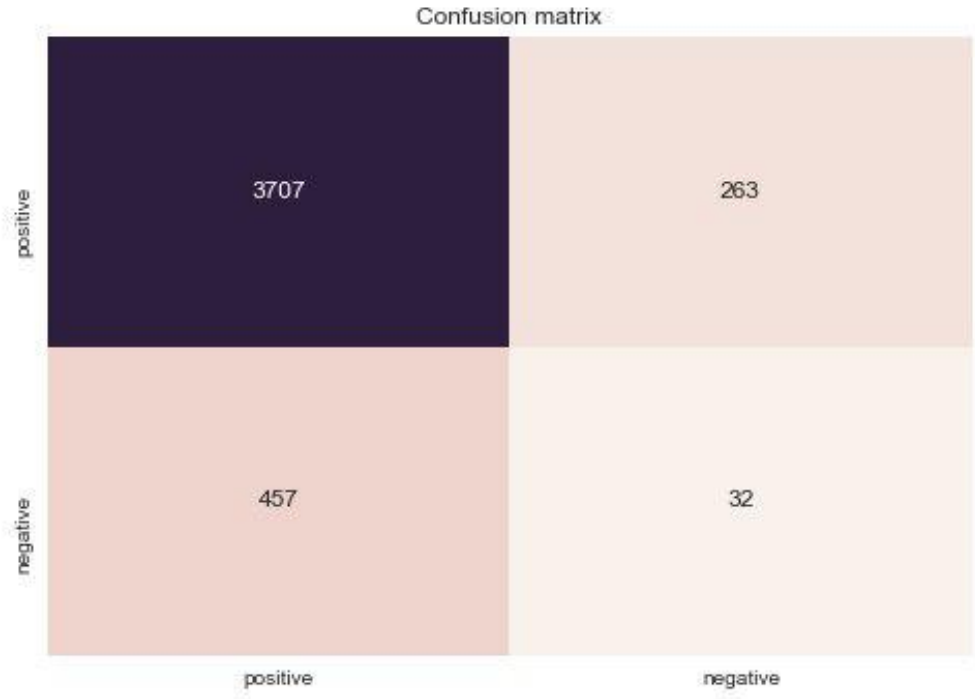
delta tfidf

$$V_{t,d} = C_{t,d} \log(\frac{|N||P_t|}{|P||N_t|}) \quad (1)$$

$V_{t,d}$ - вес слова t в документе d
$C_{t,d}$ - кол-во вхождений слова t в документ d
$|P|$ - кол-во документов с положительной тональностью
$|N|$ - кол-во документов с отрицательной тональностью
$|P_t|$ - кол-во положительных документов со словом t
$|N_t|$ - кол-во отрицательных документов со словом t

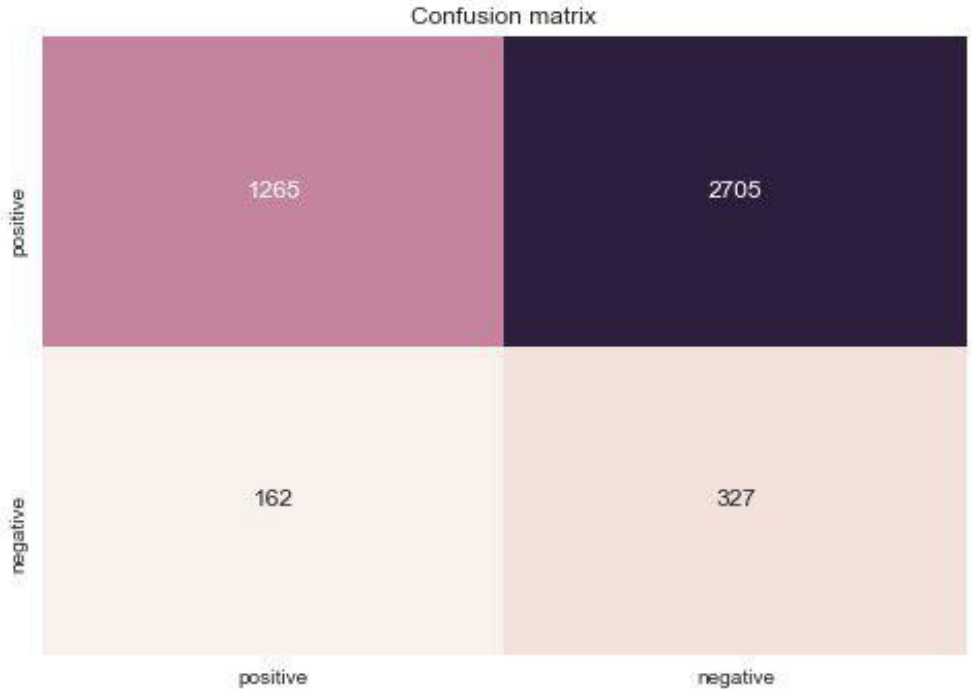
SVM на delta tfidf

	precision	recall	f1-score	support
-1.0	0.11	0.07	0.08	489
1.0	0.89	0.93	0.91	3970
avg / total	0.80	0.84	0.82	4459



SVD на тональных словах

	precision	recall	f1-score	support
-1.0	0.11	0.67	0.19	489
1.0	0.89	0.32	0.47	3970
avg / total	0.80	0.36	0.44	4459



SVM на бинарных векторах

-1.0	0.11	0.52	0.18	489
1.0	0.89	0.47	0.62	3970
avg / total	0.80	0.48	0.57	4459

Примеры

'книга супер моё мнение лучшая у пелевина если кто то только собирается познакомиться с этим автором есть ещё не читавшие рекомендую начать именно с этого произведения'
'это отличный автор матерящийся плющийся но при этом верящий в чудо'
'стар я наверно и было мне скучно и муторно уж очень я не люблю бросать книгу не закончив но грешен сломался и не смог дочитать'