Sentiment Analysis
(Feature engineering based && Word2Vec based)

Yanjian Zhang
16300200020@fudan.edu.cn

Abstract

In this project, we use feature engineering and word2vec based models for sentiment analysis. The word2vec based models and Bag of Words Based model reach 27.69% and 40.63% in accuracy respectively. Extra implemented TF-IDF based model reached 40.77% in accuracy.

1 Environment

Windows 10
Python 3.7.1
numpy 1.16.1

2 Feature engineering based sentiment analysis

2.1 Feature extraction

We use Bag of Words and TF-IDF to get features from all the sentence.

2.2 Sentiment analysis

We make a classification through Naive Bayes and Logistic regression. The result can be seen from the following table. Details can be seen from my code.

Method	Train Acc.	Dev Acc.	Test Acc.
Naive Bayes + Bag of Words	75.96	38.78	40.63
Logistic Regression + Bag of Words	95.47	37.42	40.54
Naive Bayes + TF-IDF	58.65	39.42	39.41
Logistic Regression + TF-IDF	77.73	39.42	40.77

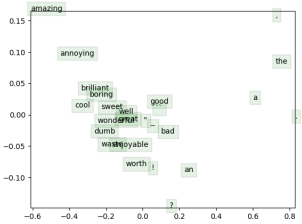
3 Word2vec based sentiment analysis

3.1 Word2vec training

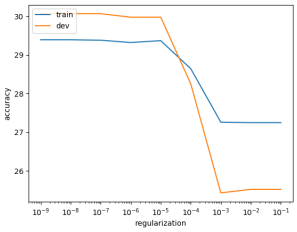
We use Skip-gram for Word2vec. After 40000 iterations, our word2vec visualization can be seen from the graph 1(a)

3.2 Sentiment analysis

In our regulation range, we get the results in graph 1(b). And the finial test result is 27.69%.



(a) Visualization



(b) Train-Dev result

```
=== Recap ===
Reg             Train           Dev
1.000000E-01    27.247191       25.522252
1.000000E-02    27.247191       25.522252
1.000000E-03    27.258895       25.431426
1.000000E-04    28.639981       28.247048
1.000000E-05    29.365637       29.972752
1.000000E-06    29.318820       29.972752
1.000000E-07    29.377341       30.063579
1.000000E-08    29.389045       30.063579
1.000000E-09    29.389045       30.063579

Best regularization value: 1.000000E-07
Test accuracy (%): 27.692308
```

(c) Numerical result