Comparing Sentiment Analysis Methods

Alp Gökçek
Department of Computer Engineering
MEF University
Istanbul, Turkey
gokcekal@mef.edu.tr

Erdal Sidal Doğan
Department of Computer Engineering
MEF University
Istanbul, Turkey
doganer@mef.edu.tr

Abstract

In the recent years, *Sentiment Analysis* has gained significant popularity with the datasets of texts with opinions being available through the social media posts. (Liu, 2010) It is a powerful tool that can be used across many applications and industries. Nowadays, we see many organizations and companies already incorporated the automated sentiment analysis systems in order to monitor their reputability, opinion of the society about their products and services, conduct market research and so on.

There are already some number of methods for building a Sentiment Analysis System. While most of them are complex methods that include very sophisticated Neural Networks, there are simpler ways of accomplishing the task as well. In this paper, we implement Sentiment Lexicons, Word2Vec, LSTM methods of building Sentiment Analysis Systems and compare the results with regard to accuracy, precision, recall and f-score.

1 Introduction

Sentiment Analysis is an NLP technique to determine whether the given piece of text is positive, negative or neutral. Prior to World Wide Web and embracement of the social media platforms such as Twitter, Facebook, forums etc. by vast majority, studies on this area were very limited. Web has changed the way people express their opinions about an entity, hence, enabled researchers to collect the opinionated texts from Web and conduct their research.

In today's world, Sentiment Analysis has been incorporated by hundreds of companies and has a wide range of use in academia. For companies that want to monitor their reputation and opinions about their products/services, sentiment analysis

offer these companies to monitor social media in real time and assess their view in public in real time.

In Sentiment Analysis, it is common to classify sentences into two or three groups, Positive, Neutral, Negative.

As an example, text snippet from a hotel review in Manhattan can be seen below.

> *"The king suite was spacious, clean, and well appointed. The reception staff, bellmen, and housekeeping were very helpful. Requests for extras from the maid were always provided.... The neighborhood is the best for shopping, restaurants." (Feldman, 2013)*

Overall, the review presents a positive opinion about the hotel. A sentiment system accepts such review as an input and assigns a score to the positivity of the sentiment in the text.

In this paper, we implement Sentiment Lexicons, Word2Vec, LSTM methods of building Sentiment Analysis Systems and compare the results with regard to accuracy, precision, recall and f-score.

2 Motivation

Given that sentiment analysis has gained such popularity and being used for various of tasks, there are many individuals and organizations that implement such systems for their works. However, initial challenge they encounter is the selection of the solution approach. As in many other engineering problems, each solution method brings its unique trade-offs.

We believe it is valuable to provide a comparison between the state-of-the-art solution methods and ease the selection process of people who seek to implement their own customized sentiment analysis systems.

Method	Accuracy	Precision	Recall	F-Score
Sentiment Lexicons	x	x	x	x
LSTM	x	x	x	x
Word2Vec	x	x	x	x
Sentiment Lexicons	x	x	x	x

Table 1: Results of the various methods of Sentiment Analysis implementation

Testing of these methods has been conducted on *Amazon Customer Review Dataset* (He and McAuley, 2016; McAuley et al., 2015). In the dataset, reviews are not labeled as Positive or Negative, however, ratings of the users to the product on the scale of 5 is available. Hence, we assume that ratings that are greater or equal to 3/5 are Positive, less than 2/5 are to be Negative Reviews.

3 Problem Definition

In this paper we evaluate various methods of building a *Sentiment Analysis System*. Methods that compared are as follows.

3.1 Using the Sentiment Lexicons of Words

In this approach, we use a dataset which presents sentiment scores of 5000 English words. In the dataset, `mean_sentiment` score for words are in the range of -3.91 and $+2.76$. Running total of these scores for each word is assigned to be the `sentiment_score` for a review. If the `sentiment_score` is less than 0, we label the review as negative, if greater than 0 we label it as positive. Finally, after obtaining the `sentiment_score` for the review, we compare the labeling of the review with the the calculated `sentiment_score`. If the label is negative and `sentiment_score` is less than 0, or if the label is positive and score is greater than 0 we conclude that the classification is correct

3.2 Using LSTM Model

Another approach is to utilize the Long Short-Term Memory Models. This method is based on Neural Networks. LSTM is a special kind of Neural Network, they are assumed to excel at NLP tasks because of their ability to process long text inputs.

3.3 Using Word2Vec Model

Word2Vec is an algorithm that use Neural Networks to learn word associations from a corpus of text. Word2Vec represents each word with unique vector with regard to their meanings and associations with other words.

In this method, we use sklearn library to train a neural network with MLP Classifier. Given the rating in range 0-5 and the review text, we associate the word vectors from review to the rating. If the rating is greater than 3 we assume that is a positive review, if less it is a negative. After these association, we get a model that given a vector as input, it can give whether is likely to be associated to a positive rating review or a negative.

4 Evaluation

For the evaluation of the models, we use *f-score*, *precision*, *accuracy* and *recall*. Each metric is defined in terms of
TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

- **Accuracy** Calculated as the ratio of *True Predictions* to *Total Number of Predictions*.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- **Precision** Calculated as the ratio of *True Positives* to *All positive classifications*.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

- **Recall** Calculated as the ratio of *True Positives* to *Positive Labeled* reviews in the dataset.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

- **f-score**

$$2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

First method of implementation is the most straight forward method among the methods we compare. In the *Sentiment Lexicons* method, by

ussing the *Domain Specific Sentiment Lexicons Dataset* (Hamilton et al., 2016), we retrieve the `mean_sentiment` score for each of the words in the text.

Later on, we calculate the overall sentiment score as running total of the `mean_sentiment` of each words. Running this method on *Automotive Reviews* from the *Amazon Customer Review Dataset* (He and McAuley, 2016; McAuley et al., 2015), we obtain the following results.
- Accuracy: 75.6%
- Precision: 95.4%
- Recall: 77.8%
- F-Score: 85.7%

Another approach we evaluated was to implement the Sentiment Analysis System using LSTM Model

5 Conclusion

In this paper, we implemented the Sentiment Analysis by calculating a total sentiment score for the text by adding the *Sentiment Lexicon* of each word.

With the repetitive tries on different models, we observed that the *LSTM Method* always had the highest accuracy score.

References

Ronen Feldman. 2013. Techniques and applications for sentiment analysis. *Commun. ACM*, 56(4):82–89.

William L. Hamilton, Kevin Clark, Jure Leskovec, and Dan Jurafsky. 2016. Inducing domain-specific sentiment lexicons from unlabeled corpora.

Ruining He and Julian McAuley. 2016. Ups and downs. *Proceedings of the 25th International Conference on World Wide Web*.

Bing Liu. 2010. Sentiment analysis and subjectivity.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-based recommendations on styles and substitutes.