# Mini project report: Movie reviews sentiment analysis with different approach

Wang Yangxuan (700025834)

May 5, 2021

**Abstract**

This project will use different feature extraction techniques and different machine learning classifiers to complete a simple binary sentiment analysis experiment. Through the analysis of movie reviews, judge whether the sentiment of the review is positive or negative.

## 1 Introduction: Contextualization and Motivation

### 1.1 Sentiment Analysis

Sentiment analysis is an pursued branch of Nature Language Processing (NLP)[1], it is a text mining process, which can analysis sentiment, opinion and subjective of text data[2]. With the quick development of internet, large amount of information exists in the form of text, such as shopping reviews, movie reviews, social media, forum website, etc. Sentiment analysis process these text data and extract useful information which reflect the emotional trend of internet users in certain period of time, and has been widely used in the field of business decision-making, opinion search, information prediction, emotion management, etc[3]. The methods of sentiment analysis usually include knowledge-based techniques, statistical methods, and hybrid approaches[4].

Knowledge-based techniques categorize text data according to some obvious words that can express emotions; Statistical methods combine machine learning and deep learning models, and the context and grammatical dependencies of words are also analyzed[5]; Hybrid approaches combines two methods mentioned. Similar to natural language processing, the basic steps of sentiment analysis include data preprocessing, feature engineering and classification.

- **Data preprocessing** mainly refers to the standardization of data, and its goal is to remove symbols, common words and other elements in the text that does not relate to the meaning of the text or have very small relationships, leaving only words that can reflect the semiotics of the text. The data standardization can reduce the error rate in NLP task, and it generally includes tokenizing word, normalizing word formats, stemming and segmenting sentences.
- **Feature engineering** is the process of turning raw data into features and it is the vital part of sentiment analysis. There are many different feature extraction techniques and the quality of text feature extraction directly related to the result of classification[6]. Common feature extraction techniques can be categorized as filtering method, like word frequency, mutual information, etc; fusion method, such as weighted KNN, the center vector weighted method; mapping method, clustering method and deep learning method[6]. Sentiment analysis can be regarded as a classification problem. Its basic idea is to find the polarity in the text and classify it into positive, negative or neutral[7].
- **Subjective Lexicon** and **Machine Learning** are two main method that used in sentiment classification. The former method base on a subjective lexicon which contain a score of each word, and this score represents sentiment of the word. Sum all subjective vocabulary scores in a text and use the highest score to represent the polarity of the text. Subjective lexicon includes dictionary based approach and corpus based approach[7].The text is transformed into feature vectors through feature extraction and used for machine learning model training. The trained model can make corresponding prediction classifications. Machine learning classification is divided into two categories: supervised learning and unsupervised learning[7].

### 1.2 Related Work

Sentiment analysis is a booming research field of natural language processing, many related studies have provided us with a large amount of literature materials.

Rudy et al. proposed a combined approach of sentiment analysis[8] which combines rule-based classification, supervised learning and machine learning, improved the classification effectiveness and achieved a good result on movie reviews. A new feature-based heuristic for aspect-level sentiment classification method is developed by Singh et al.[9], this aspect oriented approach analysis a movie from multiple reviews, which is more reasonable.

In the experiment of Ahuja et al.[10], two different text feature extraction technologies TF_IDF and N-Gram were used to extract features in text data, and six different classification algorithms (Decision Tree, Support Vector Machine, K-Nearest Neighbour, Random Forest, Logistic Regression, Naive Bayes) were used for classification, and the impact of different technologies on sentiment analysis was compared, and came to the conclusion that Logistic Regression is the best classification algorithm, and both feature extraction methods are very good.

# 2 Description of methods

## 2.1 Dataset

The data set used in this experiment provided by a Kaggle sentiment analysis competition "Bag of Words Meets Bags of Popcorn". This data set prepares 100,000 movie reviews from IMDB and 50000 reviews are labeled s positive and negative emotions. 25000 pieces of labeled data are used as the training set and the other 25000 labeled data as the test set. As for the extra data set of 50,000 pieces of data, we can use it for the training of the Word2Vec model.

## 2.2 Evaluation

Since the labels of the test set are not public, we can only upload the prediction results of the test set to the scoring page provided by the Kaggle website to obtain area under the ROC curve as the evaluation criterion.

However, in order to obtain more specific results and facilitate analysis and comparison, we also designed another set of experiments: Divide 25,000 labeled data into training set and data set according to the ratio of 7:3. Perform the same experiment. This experiment can output the accuracy, precision, and recall of the model classification, and generate a confusion matrix and visualize ROC and PR curves which improved the comparative analysis process.

## 2.3 Feature extractor

This experiment performed three different feature extraction techniques, Bag of word model, TF_IDF model, and Word2Vec.

- **Bag of word:** The bag-of-words model is a simple feature representation method that without considering the order of words in a text sentence and the connection between words but only considers the frequency of occurrences of a certain word in the vocabulary in a specific text. For example, the vector S=[0,0,0,1,1,2] represents the feature vector of a sentence, and the number represents the number of times a specific word in the vocabulary appears in the sentence , and 6 elements indicate there are 6 words in the corresponding vocabulary.
- **TF_IDF:** TF_IDF (Term Frequency_Inverse Document Frequency) is often used as a weighting factor, it combines TF and IDF, the TF represents the number of occurrences of a term in a document, and a term is more important if it occurs more frequently in a document, the formula of TF:

  $$TF(w) = \frac{Number\ of\ times\ word\ w\ appears\ in\ text}{Total\ number\ of\ words\ in\ the\ text}$$

  ; The idea of IDF is a term occurs in fewer documents, it is more discriminative the formula of IDF is:

  $$IDF(w) = Log\frac{Total\ number\ of\ documents\ in\ the\ corpus}{Number\ of\ documents\ containing\ w\ +\ 1}$$

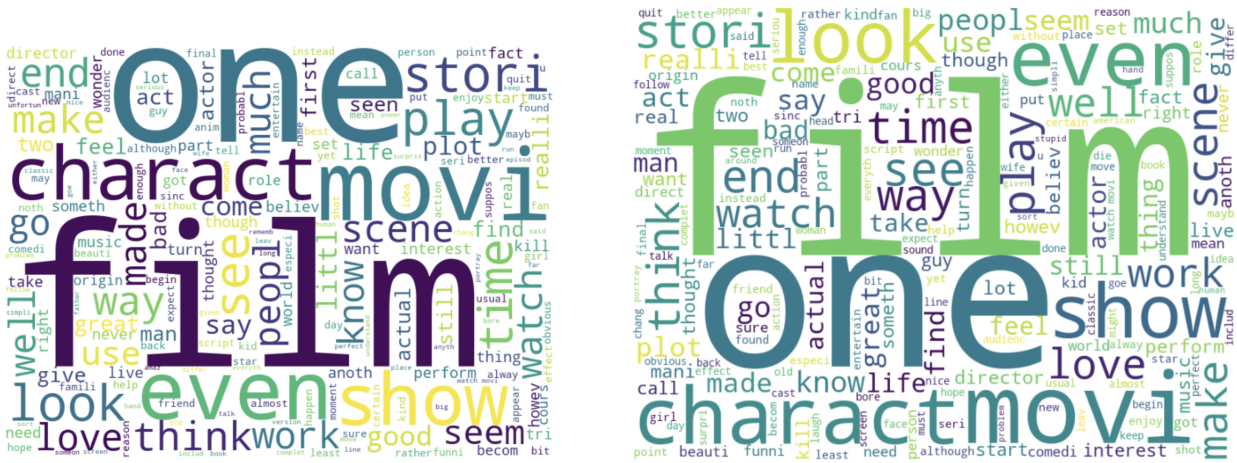
  . Consequently, the calculation formula of TF_IDF is::

  $$TF\_IDF(w)\ =\ TF(w) \times IDF(w)$$

  . The larger the TF-IDF value, the more important the word is, and it can also be said that the word is a keyword. The introduction of TF_IDF solves the problem that the bag-of-words model cannot reflect the keywords of a sentence.
- **Word2Vec:** The Word2Vec model was proposed by the Google research team led by Tomas Mikolov. It uses a trained single-layer neural network map a one-hot form of a sparse word vector to a dense vector of n dimensions. In order to speed up model training, the tricks include Hierarchical softmax, negative sampling, Huffman Tree, etc. Word2Vec includes two important models, CBOW (Continuous Bag-of-Word) and Skip-gram, Figure 1 is a schematic diagram of these two technologies. In sentiment analysis, the training of Word2Vec provides us with a set of weights, and this set of weights is the word feature vector.

## 2.4 Classifier:

In this experiment, we chose Random Forest (RF) and Support Vector Machine (SVM) as sentiment classifiers. They are commonly used classifiers in natural language processing. Random Forest (RF) is an ensemble algorithm. It uses decision tree as the base learner to build on the basis of bagging integration. Support Vector Machine (SVM) is a generalized linear classifier that performs binary classification of data according to supervised learning and it is widely used in pattern recognition problems such as portrait recognition and text classification.




(a) Train data

(b) Test data

Figure 1: Word Cloud

# 3 Discussion about experiments

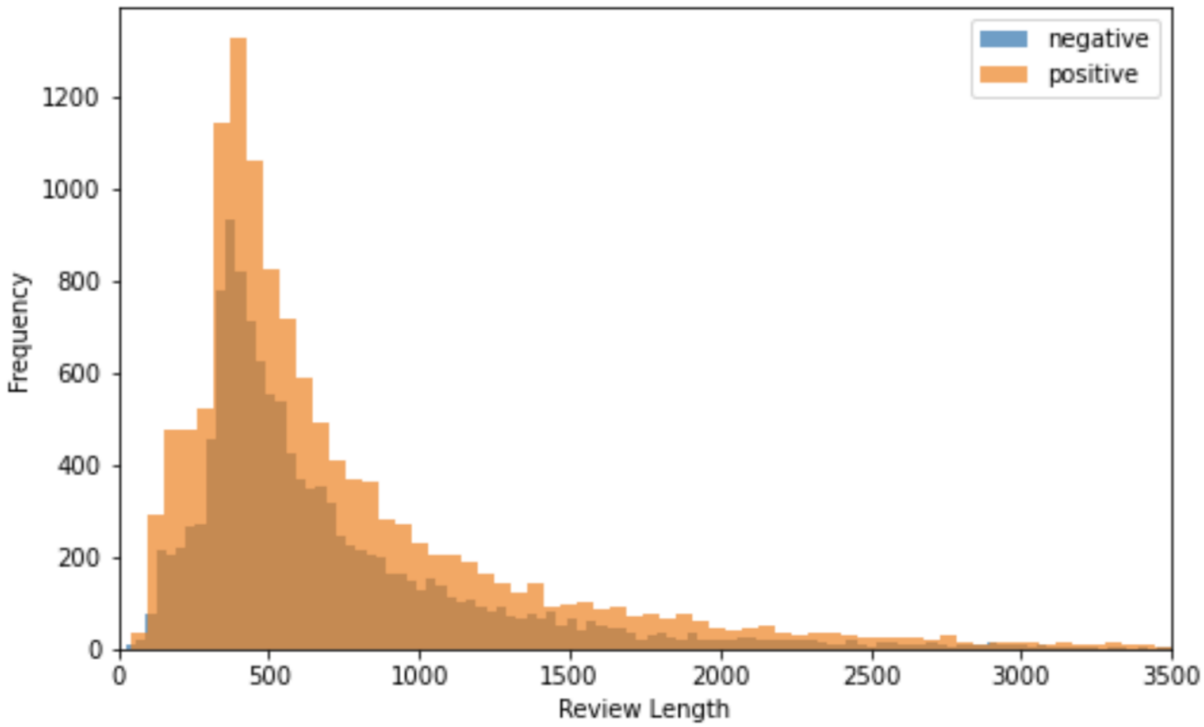
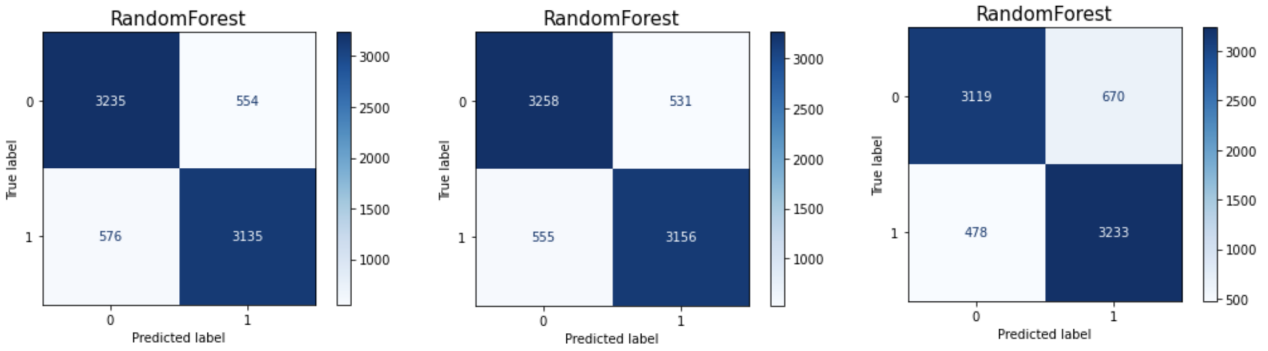



Figure 2: Distribution of review length

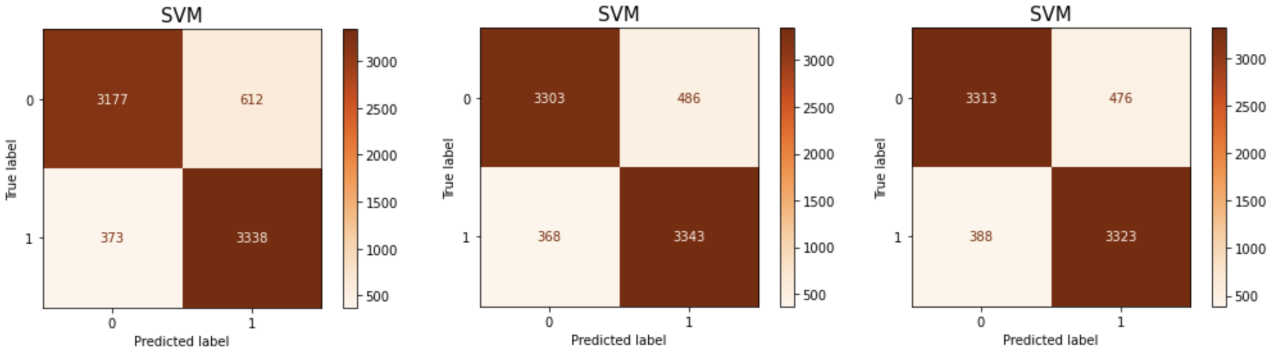
## 3.1 Data preprocessing

In this part, we use BeautifulSoup pakage to remove the HTML Markup in the text data, and then use the built-in regularization pakage .re of python to remove symbols and numbers; then the letters in the text are converted to lowercase and tokenized; finally, according to the English stopword provides by nltk library removes meaningless words in the text, and finally gets cleaned data. It is worth noting that in the Word2Vec model, we need to convert clear and good data into list format. On the other hand, in order to train the Word2Vec model, we need to prepare a sentences list as the corpus, which is obtained by combining the reviews column in the training data and the reviews in the unlabeled extra data and splitting them into a single sentence. We analyzed the cleaned data set. Fig 1(a) and Fig 1(b) are the word cloud of train data and test data, respectively, and Fig 2 describes the distribution of length (word count) of reviews presented according to different labels It is not difficult to find that the length




(a) Bag-of-word (b) TF_IDF (c) Word2Vec

Figure 3: Confusion Matrix of Random Forest




(a) Bag-of-word (b) TF_IDF (c) Word2Vec

Figure 4: Confusion Matrix of Support Vector Machine

of most reviews is between 100[U+FF5E]200, and the length of positive reviews generally exceeds that of negative reviews.

## 3.2 Feature extracting

The scikit-learn library provides us with Bag-of-word model and TF_IDF model, call CountVectorizer fit BOW model and convert the training data into vectors, and then use the trained model on the test data to get the test feature vector. Similarly, call TfidfVectorizer and perform the same operation to output the feature vector extracted by the TF_IDF method. In addition, the Word2Vec model comes from Gensim library. Call word2vec.Word2Vec in the prepared corpus to train our Word2Vec model and use the trained model to get the feature vector. Finally, we need to use the average of all word vectors as the final feature value for classification.

## 3.3 Classification

Similarly, based on the random forest model and support vector machine model provided by the scikit-learn library, we have built two functions to classify the data. For experiment 1 (split labeled data as train and test set), these functions can output the accuracy, precision, racall score of classification, plot confusion matrix(Fig 3,Fig 4),and based on these scores, we draw the comparison of the roc curve and pr curve of different methods(Fig 5,Fig 6).

| | BOW | TF_IDF | W2V |
|---|---|---|---|
| RF | 0.84788 | 0.84728 | 0.79100 |
| SVM | 0.84800 | 0.84964 | 0.84832 |

Table 1: Kaggle score

For experiment 2, we output the prediction results of the test data and score them on the Kaggle webpage, and the results are presented in Table 1.

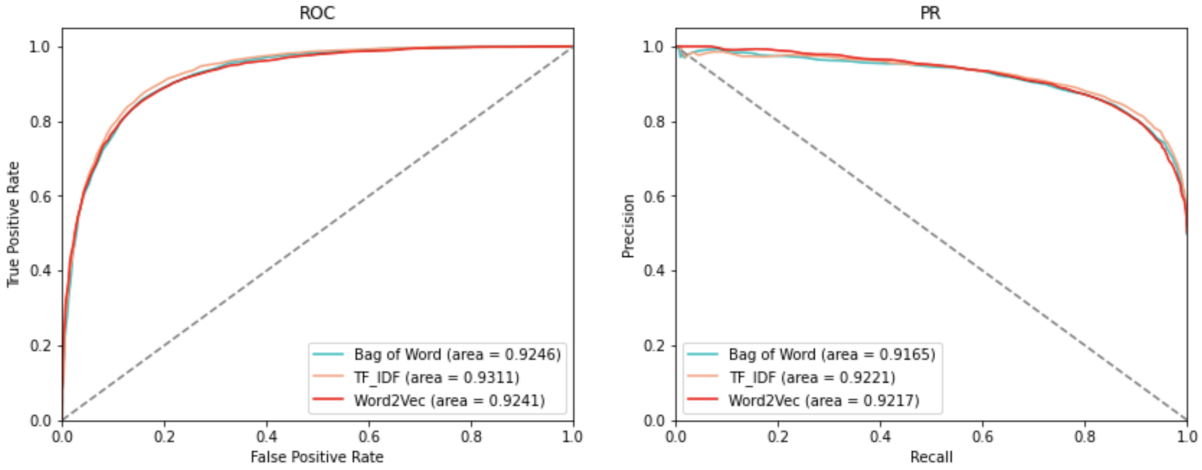



Figure 5: ROC and PR curve of Random Forest

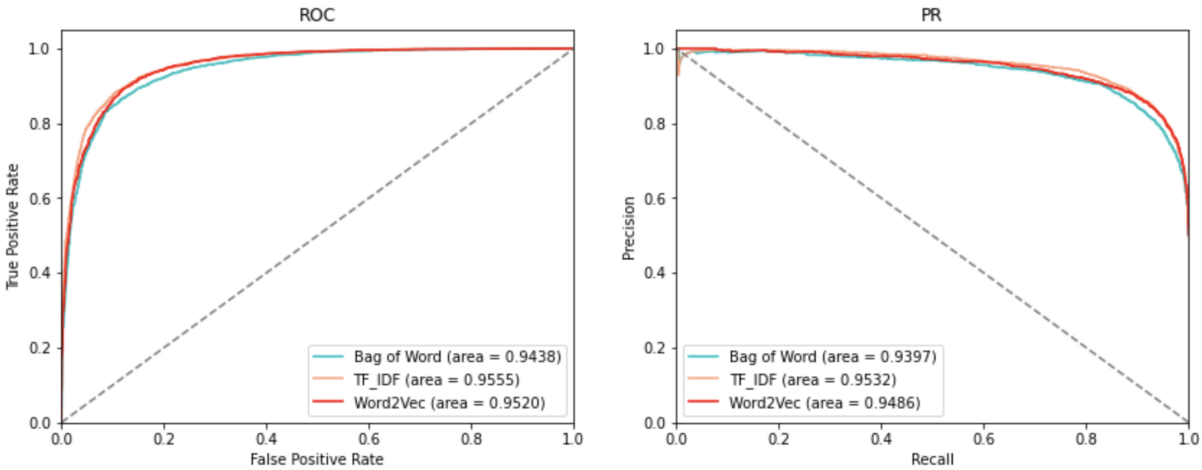



Figure 6: ROC and PR curve of Support Vector Machine

# 4 Conclusion

According to the comparative analysis of the test results, we can conclude that the three feature extraction techniques and the two classifiers have very good performance on this data set and the gap is not obvious, but Word2Vec is significantly more time expensive than the other two extractors. In terms of classifier, SVM is extremely inefficient compared to random forest, and each test may even take several hours. In summary, considering the time cost, both BOW and TF_IDF with the Random Forest model are the best sentiment analysis method in this experiment. In further research, we can try to use deep learning or heuristic algorithms for text analysis.

# References


[1] M.D. Devika, C. Sunitha, and Amal Ganesh. Sentiment analysis: A comparative study on different approaches. *Procedia Computer Science*, 87:44–49, 2016. Fourth International Conference on Recent Trends in Computer Science Engineering (ICRTCSE 2016).

[2] Walaa Medhat, Ahmed Hassan, and Hoda Korashy. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4):1093–1113, 2014.

[3] Li Ran, Lin Zheng, Lin Hailun, Wang Weiping, and Meng Dan. Text emotion analysis: A survey. *Journal of Computer Research and Development*, 55(1):30, 2018.

[4] Erik Cambria, Björn Schuller, Yunqing Xia, and Catherine Havasi. New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, 28(2):15–21, 2013.

[5] Lipika Dey and S K Mirajul Haque. Opinion mining from noisy text data. In *Proceedings of the Second Workshop on Analytics for Noisy Unstructured Text Data*, AND '08, page 83–90, New York, NY, USA, 2008. Association for Computing Machinery.

[6] Hong Liang, Xiao Sun, Yunlei Sun, and Yuan Gao. Text feature extraction based on deep learning: a review. *EURASIP journal on wireless communications and networking*, 2017(1):1–12, 2017.

[7] Harpreet Kaur, Veenu Mangat, and Nidhi. A survey of sentiment analysis techniques. In *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pages 921–925, 2017.

[8] Rudy Prabowo and Mike Thelwall. Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157, 2009.

[9] V. K. Singh, R. Piryani, A. Uddin, and P. Waila. Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification. In *2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s)*, pages 712–717, 2013.

[10] Ravinder Ahuja, Aakarsha Chug, Shruti Kohli, Shaurya Gupta, and Pratyush Ahuja. The impact of features extraction on the sentiment analysis. *Procedia Computer Science*, 152:341–348, 2019. International Conference on Pervasive Computing Advances and Applications- PerCAA 2019.