Sentiment Analysis of Amazon Reviews Data for STA561

Yi Mi[*]
Department of Statistical Science
Duke University
Durham, NC 27705
yi.mi@duke.edu

May 3, 2020

Abstract

This project is focus on the sentiment analysis on Amazon Product Reviews of Unlocked Mobile Phones. The data set is collected from Kaggle, consisting of 400 thousand reviews of unlocked mobile phones sold on Amazon.com. The methodology mainly includes four parts: data cleaning, exploratory data analysis, and modelling. I construct LSTM algorithm with Word2Vec embedding for modelling.

Keywords: *Sentiment Analysis; Word2vec; LSTM.*

1 Introduction

Natural Language Processing (NLP) is one of the most popular research fields with strong application value and potential. NLP focuses on the interactions between computers and human languages. Sentiment Analysis, or Opinion Mining, is a branch of NLP concerned with the understanding sentiment of words or sentences. Recent years have seen an increasing amount of research in sentiment analysis since it has wide applications in many fields. For instance, merchants desire to know the public's attitude toward a product, and customers also want to know other customers' feedback on a product. For a government, understanding public opinion can better improve policies and help people. As a result, further improving the accuracy of sentiment analysis can optimize customers' purchase and government policy decisions and improve business opportunities. Sentiment analysis mainly makes judgments from two dimensions: sentiment tendency and sentiment intensity. Sentiment tendencies can be divided into positive, negative and neutrality. Sentiment intensity is mainly gauged by two aspects, strength of adjectives, such as "good" sentiment intensity is less than "excellent" and strength of adverbs, such as "a little bit" sentiment intensity is less than "very".

There are mainly three sentiment analysis techniques (Pollyanna & Matheus, 2013). The classic method is based on sentiment dictionary. Using sentiment dictionaries containing positive words and negative words, we can gauge sentiment tendencies of text resources by calculating the scores of sentiment words. However, this method ignores the semantic information and word order characteristics and thus loses sentiment information. The second is based on Machine Learning, such as Twitter sentiment analysis using SVM and naive Bayes algorithm. This method has the same drawback as the sentiment

[*] https://stat.duke.edu/people/yi-mi-0

dictionary method. Using these two methods is easy to lose sentiment information. Additionally, the Machine Learning method requires many feature extraction words. The third method is based on Deep Learning, which performs better than first two methods in empirical research.

With the development of internet, the mobile devices have had a huge impact on people 's lives and have revolutionized the way people eat, work, purchase and so on. Thus, it is meaningful to study uses' sentiment on mobile phones, which dominate the consumer market to some extent. In this project, I performed sentiment analysis on Amazon Product Reviews of Unlocked Mobile Phones based on the Deep Learning method with an open source Amazon reviews dataset collected from Kaggle, consisting of 400 thousand reviews of unlocked mobile phones sold on Amazon.com. The goal is to classify the reviews into positive and negative sentiment. The methodology mainly includes four parts: data cleaning, exploratory data analysis, and modelling. I construct LSTM algorithm with Word2Vec embedding for modelling. The audience could be people from computer science and statistical science field.

2 Methodology

I first implemented data cleaning on the reviews data. The dataset contains 413840 rows in total. The total number of brands and unique products are 385 and 4410 separately. The 5 features are "Product Name", "Brand Name, “Rating”, “Reviews” and “Review Votes”. The “Rating” column consists of the scores rated by users from 1 to 5. After that, I eliminated rows with blank score since they lack essential features and plotted the distribution to do an exploratory data analysis. Figure 1 shows the distribution of 5 classes of score. According to the figure, the score rated has a very imbalanced distribution with score 5 having the largest amount of data and roughly ten times that of score 2. This figure shows a counter-intuitive pattern, inconsistent with discerning user image in recent years. A reasonable conjecture may be that the dataset spans a period of several years and when online shopping first appeared, consumers were more likely to be satisfied. It is also worthwhile to note that the imbalanced dataset may influence the performance of modelling.

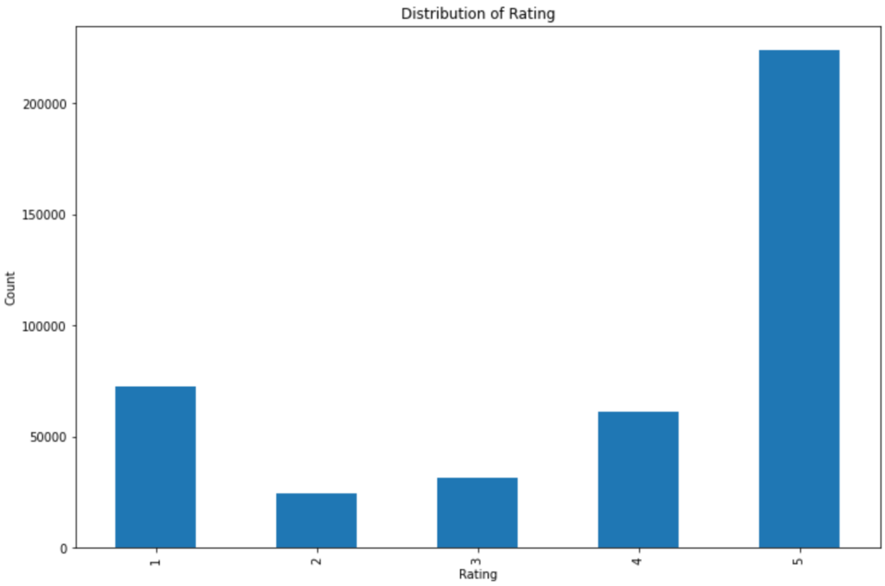

Figure 1: Rating Distribution.

And I also visualized the number of reviews of the top 20 brands as shown in figure 2. The figure shows that the most reviewed three brands are Sumsung, BLU and Apple. The least three are Alcatel, Asus and verlkool. The number of the least three is not comparable to the most three, indicating there is an oligopoly in the mobile phones market.

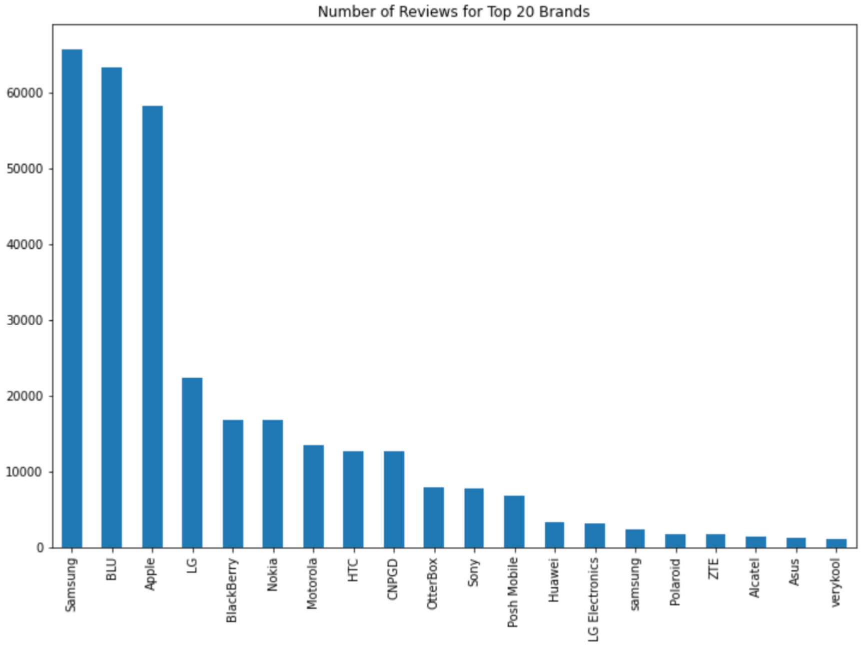

Figure 2: Number of Reviews for Top 20 Brands.

I randomly sampled 10% of the data from the total data set. To make the problem more illustrative, I eliminate the reviews with neutral sentiment, which is rating 3. Thus the ratings remained are only include 1, 2, 4, 5. After doing that, I split the data into train and test sets and the size of test data account for 10%. I could then do data clean by removing non-character such as digits and symbols, converting words to lower case, removing stop words such as ”the” and converting to root words by stemming.

To transfer the text into the language that can be read by computer, word embedding is essential, by converting a text into a numerical representation. Word2vec is a two-layer neural net used to produce word vectorization, or word embeddings, which was developed by Tomas Mikolov in 2013. It can map words from high-dimensional space to low-dimensional space in a distributed manner and retain the position between word vectors, solving the two problems of vector sparseness and semantic connection. It contains two methods, CBOW (continuous bag-of-words) and skip-gram. The training input of the CBOW model is the word vector corresponding to the context word of a certain word, and the output is the word vector of this word. Skip-gram model is opposite to CBOW in that the input is a word vector of a certain word, and the output is its context word vector.

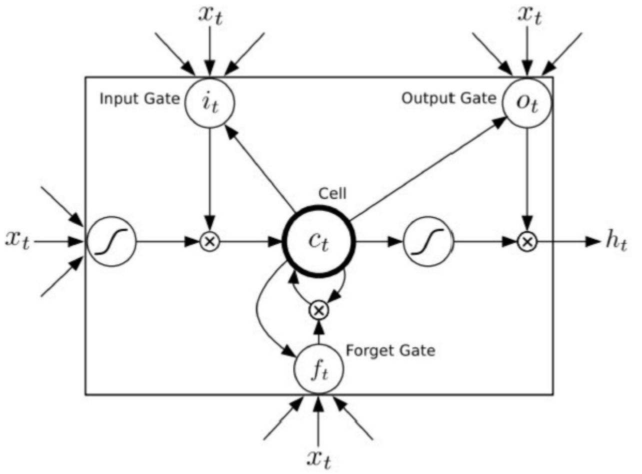

Figure 3: Internal structure of LSTM (Junfei, Zhisheng, Xiaoling, 2018).

The Long Short-Term Memory (LSTM) algorithm is an artificial Recurrent Neural Networks (RNN) published by Sepp and Jürgen in 1997. RNN uses back propagation and memory mechanisms to process sequences of any length and can maintain information. It can transfer information according to time, but the drawbacks are gradient explosion or gradient vanishing and long-term dependence. The gradient problem affects the backward transmission of errors in the network structure, and long-term dependence refers to semantically related information that is far away from each other. Therefore, LSTM was proposed. It uses a memory unit to replace the hidden nodes in the original RNN, which can learn long-term dependence information. This memory unit is composed of memory cells, input gates, forget gates, and output gates. Figure 3 shows the internal structure of LSTM algorithm. The memory cells are used to store and update historical information. The three gate structures determine the degree of information retention through the Sigmoid function. In 1999, Felix Gers introduced forget gates into LSTM algorithm, and define the goal of forget gate as “controlling the extent to which a value remains in the cell”. The input gates selectively save the new information into the cell state. The responsibility of the output gates is to determine which parts of the cell state are useful and which parts are useless. The LSTM model can avoid the gradient problem of RNN with a stronger "memory ability". It can also make good use of context feature information and retain the order information of the text, automatically selecting features for classification.

3 Results

For the modelling part, I trained a LSTM with the trained Word2Vec embedding to classify the reviews into positive and negative sentiment using Keras libarary. The main steps include first trained word embedding model, then constructing embedding layer using embedding matrix as weights, and training a LSTM model and fitting the model using log loss function and ADAM optimizer. I also transferred the data by vectorizing train and test predictors to 2D tensor and encoding labels by one-hot encoding. Then, I constructed LSTM with Word2Vec embedding layer with hyperparameters set, such as 128 hidden units, 32 batch size and 3 epochs. As for the result, the test loss is 0.1597 and the test accuracy is 0.9440 after 3 epochs.

4 Discussion and Conclusions

In conclusion, LSTM has good performance on sentiment analysis. However, there is still room for improvement since the LSTM used in this work is unidirectional LSTM. It processes the sequence in historical order, ignoring the information after the current information (Li, Zhong, Wu, 2006). Future work could focus on fixing the problem of unidirectional LSTM or building bidirectional LSTM to get better performance.

References

[1] Pollyanna Gonçalves, Matheus Araújo, Fabrício Benevenuto, and Meeyoung Cha. 2013. Comparing and combining sentiment analysis methods. In Proceedings of the first ACM conference on Online social networks (COSN '13). Association for Computing Machinery, New York, NY, USA, 27–38. DOI:https://doi.org/10.1145/2512938.2512951

[2] Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780. doi:10.1162/neco.1997.9.8.1735. PMID 9377276.

[3] Li Ji, Zhong Jiang, Wu Zhongfu. An Artificial Im-mune Algorithm for Job Scheduling in Grid Environment with Fuzzy Processing Time[J]. Computer Science. 2006, 33(2): 35-37, 64.

[4] Rumelhart, David E.; Hinton, Geoffrey E.; Williams, Ronald J. (1986-10-09). "Learning representations by back-propagating errors". Nature. 323 (6088): 533–536. doi:10.1038/323533a0. ISSN 1476-4687.

[5] Gers, F.A. (1999). "Learning to forget: Continual prediction with LSTM". 9th International Conference on Artificial Neural Networks: ICANN '99. 1999. pp. 850–855. doi:10.1049/cp:19991218. ISBN 0-85296-721-7.

[6] Junfei Zhang, Zhisheng Bi, Xiaoling Wu. Bidirectional LSTM sentiment analysis based on word vector Doc2vec[J]. Computer & Digital Engineering. 2018, 46(12): 2385-2398.