SENTIMENT ANALYSIS FOR YELP REVIEW DATASET

Xiya Xia
Department of Computer Science and Engineering
University of South Carolina
Columbia, SC 29208
xia4@email.sc.edu

1 Background

Yelp is an American public company founded in 2004. The company developed Yelp.com website and Yelp mobile app which allow users to rate and review local businesses. Users can rate the business from 1 5 stars and write text reviews on this platform. Nowadays, an increasing number of customers use Yelp when choosing a restaurant for dining. It would be meaningful if we could learn to predict ratings based on review's text alone, because free-text reviews are difficult for computer systems to understand, analyze and aggregate [1].

Sentiment Analysis, also known as opinion mining, uses natural language processing and machine learning techniques to determine whether a text unit is positive or negative. It has a wide range of applications in different areas. For example, it is used to predict movie or book ratings based on news articles or blogs [2].

Deep learning models have a wide applications in areas including computer vision, speech recognition, pattern recognition, and natural language processing etc.. These models try to learn data representation using a layered and hierarchical architecture. Recurrent Neural Networks (RNN) is a special type of neural network designed for sequence problems. The range of contextual information which standard RNN can access is quite limited. The influence of a given input on the hidden layer, and therefore on the network output, either decays or blows up exponentially as it cycles around the network's recurrent connections. This problem is named as the vanishing gradient problem[3]. Long Short-Term Memory (LSTM) is a novel type of RNN architecture specifically designed to address the vanishing gradient issue. It is capable of learning order dependence in sequence prediction problems.

The purpose of this project is to predict customers' sentiment on the restaurants based on their Yelp reviews. In this project, I used Long Short-Term Memory (LSTM) networks to predict the sentiment of the Yelp reviews.

2 Exploratory Analysis

The dataset was downloaded from Kaggle website. It contains 1,125,458 rows and 10 columns. Only 100,000 rows were included in the analysis due to the computer memory issue. The ten variables of the dataset are: user_id, review_id, text, votes.cool, business_id, votes.funny, stars, date, type, vote.useful. Text is the customers' comments on the business. Stars are the customers' rating.

Fig. 1 presents the distribution of the variable stars.

2.1 Dependent Variable

The dependent variable is the polarity of the sentiment (positive/negative). If the rating of the review is equal or above four stars, it is considered as a positive review. The review is considered negative if the rating is equal or below two stars. Three stars reviews are excluded from the analysis. In this analysis, there are 64,471 positives and 21,247 negatives. Fig. 2 presents the distribution of the label.

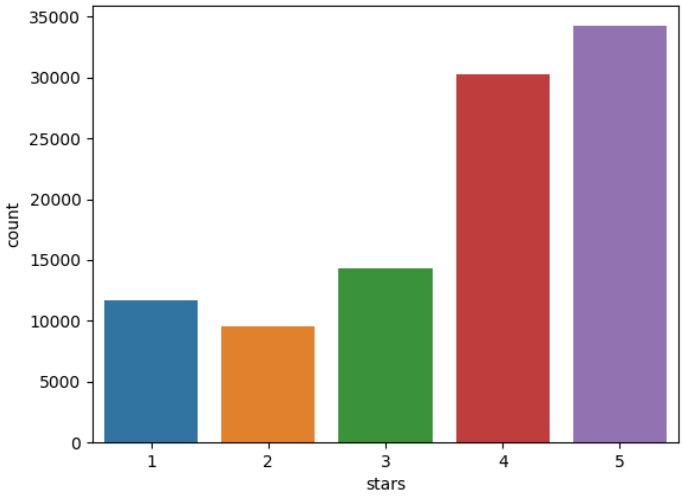

Figure 1: Distribution of Stars

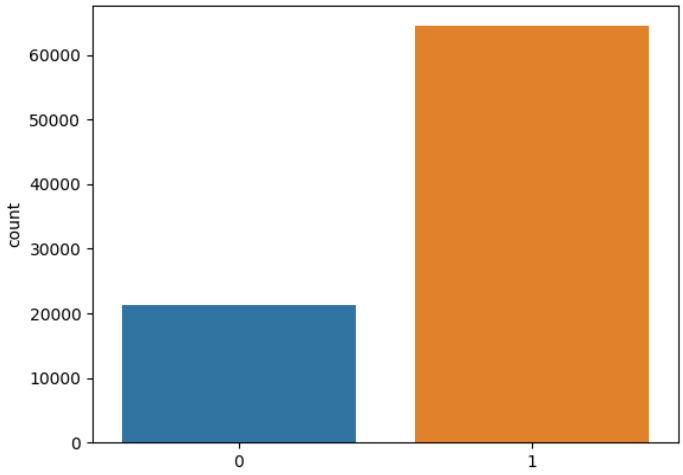

Figure 2: Distribution of Label

2.2 Independent Variable

The feature of this analysis is the text variable. Fig. 3 presents the distribution of the review length.The length of the most reviews is less than 200 words.

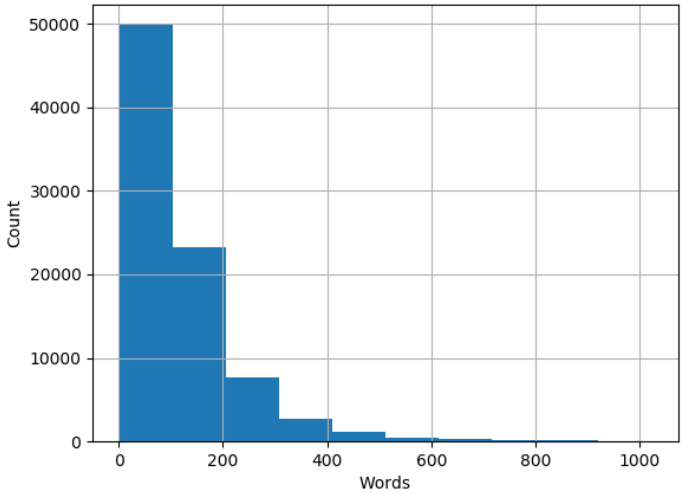

Figure 3: Review Length

3 Data Pre-processing

3.1 Feature Extraction

- Remove punctuation: All punctuation symbols are predefined in Python. They can be removed easily using built-in function.
- Create list of reviews: Text is split into individual words.
- Create mapping dictionary: This step is to create an index mapping dictionary. The words with higher frequency are assigned with lower indexes.
- Encode the words: After creating an index mapping dictionary, the words in the reviews can be replaced with integers.

3.2 Padding

I pad all the reviews to a specific length. As a result, all the reviews were adjusted into a pre-determined sequence length. For reviews shorter than the sequence length, I will pad the blanks with 0s. For the reviews longer than the sequence length, I will truncate the reviews into words within the sequence length. After padding, the features and the label were used to fit a Long Short Term Memory networks (LSTM) model.

3.3 Using TF-IDF package

I also tried use the TF-IDF package to extract features from text and vectorize the words. TF-IDF reflects how important a word is to a document. TF-IDF considers both counts and frequencies of words in a text. It counts the number of times each word appears in a text. It also calculates the frequency that each word appears in a text out of all the words in the text.

4 Model Development

The data was randomly split into training (80%) and test (20%) dataset. I used logistic regression, Naive Bayes Classifier and LSTM model for the analysis. The logistic regression and Multinomial Naive Baye classifier are used as benchmark models in the analysis.

4.1 Logistic Regression

Logistic regression uses a logit function to model a binary target. Mathematically, logistic regression is defined as:

$$Pr(Y_i = 1|X_i) = \frac{exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + ... + \beta_n X_n)}{1 + exp(\beta_0 + \beta_1 X_i + \beta_2 X_2 + \beta_3 X_3 + ... + \beta_n X_n)} \tag{1}$$

Logistic regression models the probability of the default class (i.e.,Positive sentiment). The logistic regression serves as a benchmark in the analysis.

4.2 Multinomial Naive Bayes

The basic idea of Naive Bayes technique is to find the probabilities of classes assigned to texts by using the joint probabilities of words and classes. The Naive Bayes classifier serves as a benchmark in the analysis.

4.3 Long Short Term Memory networks

An LSTM layer consists of a set of recurrently connected blocks, known as memory blocks. Each block contains one or more recurrently connected memory cells and three multiplicative units: the input, output and forget gates. These provide continuous analogues of write, read and reset operations for the cells [3].

Here are the architecture of the LSTM model.

- Embedding layer: Embedding layer creates a look up table where each row represents an embedding of a word. The embedding layer converts the integer sequence into a dense vector representation.
- LSTM layers: This layer is defined as the number of hidden nodes and the number of layers to be stacked.
- Fully connected layer: A fully connected output layer that maps the LSTM layer outputs to a desired output size.
- Sigmoid activation layer: This layer adjust all outputs into a value from 0 to 1.

Hyperparameters in the LSTM model is shown as below:

- Batch size: Dataset is divided into batches to pass through the neural network. The batch size is 50 in the analysis.
- Hidden layers: This is the number of units we use in our LSTM cell. The number of hidden layers is 256.
- Dropout:Dropout is introduced that specifies the probability at which outputs of the layer are dropped out. The dropout is 0.3 in the analysis.
- Learning rate —Learning rate is 0.001 in the analysis.
- Epochs — This is the number of iterations (forward and back propagation) the model needs to make. The epoch number is 4 in the anlaysis.

5 Evaluation

I evaluate the model performance using accuracy. Accuracy is calculated by the equation below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

where TP, TN, FP, FN are the number of True Positives, True Negatives, Fasle Positives, False Negatives. The results are shown in the table below:

Metric	LSTM	TF-IDF +Logistic Regression	TF-IDF +Naive Bayes Classifier
Accuracy	0.93	0.90	0.75

For the above-mentioned table,LSTM model performs best in terms of accuracy. The accuracy of logistic regression is comparable to LSTM. The performance of Naive Bayes classifier is the worst among the three approaches.

6 Discussion

From this analysis, I have several findings. Firstly, the sentiment of the Yelp reviews can be effectively classified. LSTM model shows best performance in predicting the sentiment. Secondly, The data size affect the classification performance. More training data can lead to better performance. I select 100,000 rows from the original data which reaches the limit of the memory in my computer. Thirdly, some reviews are of 0 length. This will affect the model performance on the test data because these reviews' sentiment can't be predicted.

In future work, I will focus on exploring hybrid approaches. Multiple models and techniques can be combined in order to improve the accuracy of the sentiment classification model. In addition, BERT (Bidirectional Encoder Representations from Transformers) [4] model may also be used in the sentiment analysis. The BERT transformer model is designed to pre-trained deep bidirectional representations from unlabeled text by using information from neighboring words. The advantages of BERT includes quicker development and more efficiency. I am learning with BERT and will implement it in this dataset later on.

References

[1] Gayatree Ganu, Noemie Elhadad, and Amélie Marian. Beyond the stars: improving rating predictions using review text content. In *WebDB*, volume 9, pages 1–6. Citeseer, 2009.

[2] Namrata Godbole, Manja Srinivasaiah, and Steven Skiena. Large-scale sentiment analysis for news and blogs. *Icwsm*, 7(21):219–222, 2007.

[3] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610, 2005.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.