Sentiment Classification of Yelp Business Reviews by Supervised Machine Learning and Deep Learning Approaches

Fengdi Li Jie He Tianyi Yang

Abstract

This project aims to compare and contrast models for sentiment classification. Specifically, it seeks to predict Yelp star rating from the text comments. Both supervised and deep learning methods are applied. Supervised models include Decision Tree, Random Forest, Logistic Regression and Naive Bayes Classification combining with four features (BoW, Word Tf-idf, N-gram Tf-idf, N-char Tf-idf).

Besides supervised ones, deep learning models are also worth considering, since it saves the time cost on feature extraction without significant reduction in accuracy. This project tests the performance of neural networks based models such as Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN) in sentiment classification. We found that in general, deep learning models perform slightly better than supervised models. Among all supervised models, logistic regression with Word Tf-idf is the best performer.

1 Objective

With the development of technologies, gathering a large amount of comments have become easier but interpreting them harder. Thus, the need to interpret these comments in an efficient way emerged, and sentiment analysis has proved to be a viable method. This project studies how well reviews in forms of text documents predict the overall attitude. Yelp review is an ideal dataset for this task as ratings from one to five stars are provided alongside the text comments, serving as labels in supervised machine learning models. Besides classical ones, deep learning models are also worth considering, since it saves the time cost on feature extraction without significant reduction in accuracy. Thus, this project will also test the performance of neural networks based models such as Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN) in sentiment classification.

2 Data

The original data source we used, published by Yelp (Yelp Inc., 2019), contains the information about approximate 0.2M businesses across 11 metropolitan areas in four countries, along with over 5M user reviews. It was originally put together for the Yelp Dataset Challenge which is a chance for students to conduct research or analysis on Yelp's data and share their discoveries. Considering the fact that our personal machines are not capable of handling such a giant data modeling without help of adequate server or cloud services, we managed to downsize our finalized dataset after binarization of individual ratings (rating larger than or equal to 4 assigned to class 1, less than 4 assigned to class 0) by selecting 40 randomly picked reviews under each restaurant business, calculating the average star ratings, and then picking 40% of businesses in each class. This down-sampling method selected a total of 73,825 restaurants with corresponding reviews created in 2017, which also alleviated the class imbalance issue.

3 Background

Mejova (2009) gave us a fundamental overview of the basic concepts in sentiment analysis. In general, sentiment analysis classifies texts in dichotomy: positive and negative, or sometimes as a range. When conducting a sentiment classification, two important considerations are feature selection and model selection. Carrillo-de-Albornoz et al. (2018) evaluated many different features when performing sentiment analysis on patient opinions from online health forums. They used traditional bag-of-words as the benchmark and compared it with word embeddings. Its result highlights the importance of feature selection in sentiment analysis and motivates us to try different features in our project.

On model selection, Ye et al. (2009) applied Naïve Bayes and SVM algorithms for sentiment analysis of the online reviews on travel destinations and indicated that the SVM approach generally outperformed the Naïve Bayes approach. In our project, we will also test different models and compare them. Deep learning methods like convolutional neural networks (CNN) and recurrent neural networks (RNNs) have also been employed. Wang et al. (Wang, 2016) proposed taking local features extracted by CNN, with windows of word-embedding length and weight matrices, as input to RNN to capture long-term dependencies. This model achieved higher accuracy than existing models. Thus, we will also try Deep Learning in our project.

Researches conducted based on the Yelp reviews data, however, are limited. Yu et al. (2017) tested a Supported Vector Machine Model to classify these reviews as positive and negative. This paper further expands on machine learning model selection.

4 Methodology

4.1 Feature Extraction and Review Representation

To extract features in a format supported by previously mentioned classical machine learning algorithms from a dataset formed by text, we built feature vector representations using methods well-known in Natural Language Processing including Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) at word，N-gram and character levels. A review S will be made up of a set of pairs of <t_i, w_i>, where t_i represents the ith term (word, n-gram or character) in the vocabulary and w_i represents corresponding weight calculated by raw counting or inverse-document frequency.

4.2 Classical Supervised Models

Among all available classification methods, models with ease of interpretation are priorly chosen. As sentiment analysis involves much more human judgement than other classification tasks, we try to keep away from models that are completely “black boxes”. Generally, there are two types of models: statistical models and machine learning models. In this project, two models from each type are implemented.

Statistical Models: Statistical models are chosen for their interpretability. While many machine learning models produce “black boxes”, it is generally possible for humans to understand statistical models. Two simplest methods are chosen for this specific task: Naive Bayes and Logistic Regression.

Machine Learning Models: Decision tree is one of the most interpretable machine learning models that might be suitable for our task, as the tree-diagram it produces mimic the human decision process. However, single trees typically perform better on smaller datasets. Thus, under the assumption that ensemble learning methods might improve performance, random forest will be included as well. In short, two methods are implemented: Decision Tree and Random Forest.

4.3 Deep Learning Models

Besides classical supervised learning methods, deep learning is part of a broader family of machine learning algorithms that attempts to imitate how the human brain works by employing artificial neural networks to process data. It aims at learning feature hierarchies with features from higher levels of the hierarchy formed by the composition of lower level features, which allows the system to learn complex functions mapping the input to the output directly from data, without depending completely on human-crafted features, such as BoW and TF-IDF. Compared to classical machine learning methods, the performance of larger neural networks continues to increase when feeding on more and more data.

LSTM: LSTM is a special RNN architecture designed to model temporal sequences and their long-range dependencies. It has a memory that "remembers" previous data from the input and makes decisions based on that knowledge. Thus, LSTMs are well-suited for text data inputs, since each word in a sentence has meaning based on the previous and upcoming words.

CNN: CNN is a network initially created for image-related tasks and also been applied on text mining, as they can learn to capture specific features regardless of locality. In the case of text data, it could capture a phrase regardless of where it appears within the sentence.

Three deep learning architectures were implemented in our study, including a LSTM baseline model, a CNN-LSTM combined model, and a BiLSTM-CNN combined model. As our deep learning baseline, the LSTM layer receives word embeddings as input, outputs the sequence representations to fully connected dense layers, and ultimately predicted as either a positive or negative class. CNN-LSTM, explained by Sosa (2017), will receive word embeddings as input, where its outputs will be pooled to a smaller sequence of higher-level phrase representations which then fed into an LSTM layer to obtain the sentence representation. The intuition behind this model is that the convolution layer will extract local features and the LSTM layer will then be able to use the ordering of said features to learn about the input's text ordering. CNN-LSTM is able to capture both local features of phrases as well as global and temporal sentence semantics. On the contrary, BiLSTM-CNN model consists of an initial Bidirectional LSTM layer which will receive word embeddings for each token in the review as inputs. The intuition is that its output tokens will store information not only of the initial token but also any previous tokens; In other words, the LSTM layer is generating a new encoding for the original input. The output of the LSTM layer is then fed into a convolution layer which we expect will extract local features.

4.4 Word Embedding and Review Representation

Word embedding is the collective name for a set of language modeling and feature learning techniques in NLP where words or phrases from the vocabulary are mapped to vectors of real numbers. In our research, we built the word embedding layer based on the well-recognized word2vec tool, published by Google Code, where word vectors were trained on Google News dataset. Let v be the size of vocabulary and d be the length of a word embedding, then the word embeddings of all the words in the vocabulary are encoded by column vectors in an embedding matrix $Q \in R^{d \times v}$. For a review length of l, it can be represented by $S = [w_1, w_2, ..., w_l\,]$.

5 Result

We trained previously described models on 70% of randomly selected samples from our processed dataset and evaluated them on the rest 30%. In particular, both classical machine learning models and deep learning models were trained and tested on the same dataset split in order to maintain the consistency by controlling the random seed. Accuracy and F1 scores were chosen as the performance indicators in our study.

		Models			
		Decision Tree		**Random Forests**	
		Acc	F1	Acc	F1
Features	**BoW**	0.637	0.621	0.688	0.604
	Word TF-IDF	0.640	0.625	0.691	0.611
	N-Word TF-IDF	0.611	0.600	0.669	0.580
	N-Char TF-IDF	0.615	0.590	0.667	0.579
		Naïve Bayes		**Logistic Regression**	
		Acc	F1	Acc	F1
Features	**BoW**	**0.756**	**0.746**	**0.759**	**0.733**
	Word TF-IDF	**0.754**	**0.726**	**0.773**	**0.751**
	N-Word TF-IDF	0.736	0.710	**0.750**	**0.727**
	N-Char TF-IDF	0.736	0.688	**0.766**	**0.742**

Table 1: Results of the four classical supervised learning models combined with four different features.

According to the table above, we can see that in general, the model chosen, rather than the feature, is the primary driver of the accuracy. In terms of features, Word TF-IDF generally performs the best. In terms of model choice, logistic regression is the best. The best performing model is generated by the combination of the two.

Models					
LSTM		**CNN-LSTM**		**BiLSTM-CNN**	
Acc	F1	Acc	F1	Acc	F1
0.752	**0.726**	**0.756**	**0.728**	**0.756**	**0.740**

Table 2: Results of the three deep learning models.

We observed that all these three models have a fair accuracy over 75% on the test set without evident difference on the scores. Comparing to the performance of previous classical supervised models, we found that the logistic regression model slightly outperforms our deep learning models.

6 Discussion

First, due to limited resources, we had low computing power and were not able to train on a dataset with a larger size. We expect a larger dataset would render better performance. In the future, this issue can be addressed via parallel computing on cloud infrastructures.

Second, yelp star ratings are treated as "golden labels", that is, these reviews represent the sentiment accurately. However, yelp star ratings are still somewhat objective. Sometimes, we see very positive reviews, but a low rating (3 stars). In the future, we could probably try taking more extreme reviews (i.e. 5 stars & 1 star only) and build classification models to avoid the gray area in the middle.

Third, we could do multiclass classification instead of binary classification, which might give us a better insight.

Finally, we noticed that sometimes the language used to write the review is not always English. This can also cause problems in building our model. In the future, if have time, we need to further clean the data to address this language issue.

References

Yelp, Inc. Yelp Dataset. https://www.kaggle.com/yelp-dataset/yelp-dataset (accessed April 5, 2019).

Mejova, Y. (2009). Sentiment analysis: An overview. University of Iowa, Computer Science Department.

Carrillo-de-Albornoz J, Rodríguez Vidal J, Plaza L (2018) Feature engineering for sentiment analysis in e-health forums. PLoS ONE 13(11): e0207996. https://doi.org/10.1371/journal.pone.0207996.

Yu, Boya & Zhou, Jiaxu & Zhang, Yi & Cao, Yunong. (2017). Identifying Restaurant Features via Sentiment Analysis on Yelp Reviews.

Wang, X., Jiang, W., & Luo, Z. (2016). Combination of convolutional and recurrent neural network for sentiment analysis of short texts. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers (pp. 2428-2437).

Ye, Q., Zhang, Z., & Law, R. (2009). Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert systems with applications, 36(3), 6527-6535.

Sosa, P. M., & Sosa, P. M. (2017). Twitter Sentiment Analysis using combined LSTM-CNN Models. https://www.academia.edu/35947062/Twitter_Sentiment_Analysis_using_combined_LSTM-CNN_Models