Sentiment Analysis of Online Reviews using Naïve Bayes Classification and LSTM

Weiliang Hu, Nansu Wang, Robert Wan

Introduction

Sentiment Analysis is one of the most important applications of modern natural language processing. It identifies, extracts, quantifies, and studies subjective information. Sentiment analysis is widely applied to voice of the customer materials such as reviews and survey responses, online and social media. In this project, we conduct sentiment analysis using two datasets. The customer reviews of fine-foods from Amazon, and the customer reviews from Yelp. We apply Naïve Bayes model and LSTM to the data and explore interesting results.

1. Problem Definition

Our project aims to build models for sentiment analysis. Specifically, review sentences are used to predict if the reviews are “positive” or “negative”. It is a binary classification problem.

2. Generative Probabilistic Model

The generative probabilistic model that we use is the Naïve Bayes model. The idea behind the Naïve Bayes model is simple: We use Bayes theorem to forecast membership probabilities of each class, such as the likelihood that a given record of data point belongs to that class. The most likely class is defined as the one having the highest probability. For a hypothesis with two occurrences A and B (In our case, sentiments 1 (negative) and 2 (positive)),

MAP (A)

= max (P (A | B))

= max (P (B | A) * P (A))/P (B)

= max (P (B | A) * P (A)

The Naïve Bayes generation method that I build follows this idea: First we get an existing dataset (test.csv) with sentences and sentiment tags for each text sentence. Train the probability model on the dataset, then store the conditional probabilities of each word in dictionaries. We then draw a sentiment type based on the sentiment distribution of the existing data set and generate a sentence for the drawn sentiment type using the conditional probabilities. We repeat this process n times to generate the sentences that we call our synthetic data.

When actually applying the model, I find that the multinomial Naive Bayes model provided by Sk-learn package is the most suitable for classification with discrete features (e.g., word counts for text classification). As a result, we should use it for our sentiment analysis. It follows the same logic of my Naïve Bayes algorithm. Since it considers more cases, it out-performs the Naïve Bayes model that I built myself for generating the texts.

3. Neural Network Model

The neural network solution is based on LSTM that is suitable for sentiment analysis as we learned in the class. Three major steps are adopted in the project. First, reviews are pre-processed and encoded. Second, encoded reviews are padded and truncated into features. Third, an LSTM neural network is constructed and trained for solving the problem.

Specifically, the punctuations are removed from the reviews. Then, these reviews are converted to lower cases. The frequent histogram is made for each word using all reviews. The dictionary of the histogram is then sorted from the most frequently appeared word to the least one. Incrementing integers starting from 1 are used to encode the words. For example, the most frequently appeared word is encoded as 1, the second most frequently appeared word is encoded as 2, etc.

After step one, each review corresponds to a list of integers. To maintain critical information of the input data, and also to standardize the data, truncation and padding are adopted. All lists of integers are truncated into 256 dimensions. If the length of the list is less than 256, zeros are padded to the list.

After step two, a 256-dimension feature for each review is created and as the input of the neural network. The structure of the network is as shown in Fig. 1. Encoded features are input into an embedding layer of 512 dimensions. Then, two LSTM layers are connected followed by a dropout layer. At last, the sigmoid activation function is adopted.

```
LSTM(
(embedding):
Embedding(24896, 512)
(lstm): LSTM(512, 256, num_layers=2, batch_first=True, dropout=0.35) (dropout): Dropout(p=0.2, inplace=False)
(fc): Linear(in_features=256, out_features=1, bias=True)
(sigmoid): Sigmoid()
)
```

In the training phase, binary cross-entropy is used as a loss function since it is good for binary classification problems. The model is trained under the learning rate of 0.001 with the Adam optimizer. Five epochs of 600 steps are performed.

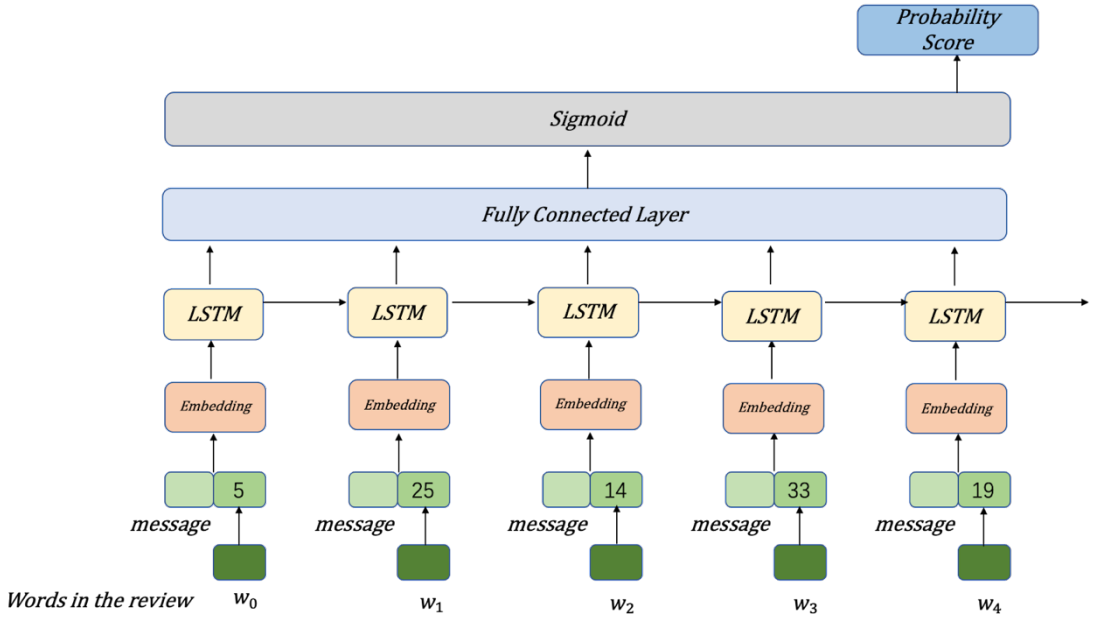

Fig. 1 LSTM Structure for Sentiment Analysis

4. Results and Findings

(1) Synthetic Dataset

The synthetic data is generated using the Naive Bayes model. The approaches are mentioned above.

When applying the Naïve Bayes model to the synthetic data, the results are shown below: since we basically generated the data using the same model, there is a problem of overfitting here. As a result, the accuracy is very high: 95%. The result of the log-loss in our output csv does not show significant deviance from the predicted result and real values. Splitting the synthetic data to training and testing sets does not make a difference due to overfitting.

	precision	recall	f1-score	support
1	0.94	0.96	0.95	5000
2	0.96	0.94	0.95	5000
accuracy			0.95	10000
macro avg	0.95	0.95	0.95	10000
weighted avg	0.95	0.95	0.95	10000

By using 64% training data, the LSTM model achieves 86.9% on the synthetic data. The overall accuracy is good. It gives the right results in most cases. The model did not give the right result when the sentence is short and vague. For example, in the synthetic dataset, the review “always full order” is positive. It is even hard for a human to predict it correctly without context.

(2) Real Dataset

The real data is part of the Amazon Fine Food Review dataset on Kaggle (https://www.kaggle.com/snap/amazon-fine-food-reviews). We use 10,000 lines of the dataset.

When applying the Naive Bayes model to all the real data (Reviews.csv), we have an accuracy of around 70%. The result of the log-loss in our output csv shows stronger deviance from the predicted result and real values. However, for comparison with the neural network approach, we also need to apply our model using the same training and testing dataset (64% training and 20% testing), we chunk Reviews.csv using the first 10000 lines of data, using 6400 observations as training set and 2000 as testing set, our result is shown below, we achieve an overall accuracy of 79%:

	precision	recall	f1-score	support
1	0.74	0.67	0.70	732
2	0.82	0.86	0.84	1268
accuracy			0.79	2000
macro avg	0.78	0.77	0.77	2000
weighted avg	0.79	0.79	0.79	2000

The results are generally as expected. For the naïve bayes approach, we have a good accuracy because most of the time features in the sentences are independent of the labels. However, in some cases where the features in the reviews (such as personal feelings and biased opinion on the companies) are not independent of positive or negative sentiments, the naïve bayes would make wrong decisions.

By using 64% training data, the LSTM model achieves 83.5% on the real data. The overall accuracy is good. It gives the right results in most cases. The model has a hard time predicting the true sentiment in some cases. For instance, “I was surprised … ”. It is hard for the model to understand the true meaning when the customer is surprised and then tells some facts about the merchandise.

5. Conclusion

Comparison:

	Naïve Bayes	LSTM
Accuracy	95%, 79%	89%, 84%
Time	2min on CPU	10min+ on CPU, 2min on GPU
Computational Requirements	GPU is NOT needed	GPU needed

Interpretability	High, conditional probabilities are obtained	Low, parameters are in blackbox

Pros and Cons:

To sum up, the LSTM model performs better than the Naïve Bayes in the real data. Though Naïve Bayes performs better in the synthetic data, it is because it is the data generator and also the predictor. In some sense, the Naïve Bayes model suffers from overfitting as for the synthetic data. However, the Naïve Bayes model requires less computational resources (i.e. does not require GPU) and is trained faster than the LSTM. The Naïve Bayes model is more interpretable than the LSTM.

Github Repo: https://github.com/wh153/Sentiment-Analysis-of-Human-Reviews-