Sentiment Analysis on Product Reviews

Jun Wang

April 18, 2017

Abstract

In this final project, a baseline algorithm, Multinomial Naive Bayes Model, Bernouli Naive Bayes Model, and the MaxEnt Model are implemented with different feature selection strategies. Through 10-fold cross validation and t-test between different aspects of the comparison, it is concluded that MaxEnt Model has the best performance with bigrams and unigrams feature together, but the POS tagging and the feature selection with information gain are unnecessary for the implementation.

1 Introduction

Sentiment analysis is an important application of natural language processing. For example, people can apply the sentiment analysis to speeches of a politician to predict the political opinion. People can also utilize the sentiment analysis to the movie reviews to predict the performance of box-office of a specific movie.

There multiple ways to implement the sentiment analysis before involving the machine learning techniques. For example, people can mark the sentence with a positive mark or negative mark by reading through the whole sentence by hand, but it is very time consuming and cannot handle a dataset with thousands of or even more sentences. Another simple way is called lexcial ratio method. In this method, people built two dictionaries named positive-lexicons and negative-lexicons. Then, through comparing the number of positive words and negative words in a sentence according to these two dictionaries, it is possible to determine whether this sentence is positive, negative, or neutral. However, because of the ambiguity of the natural language and the fast generation of the new words in modern English, this simple algorithm is less and less useful.

Thus, it is necessary to implement some more powerful algorithms for sentiment analysis with the machine learning techniques. In NLP area, naive bayes model and the maxent model are very powerful for sequence labeling, so it is possible to utilize these two methods for sentiment analysis as well.

Naive Bayes model has been proved to be very useful and simple in sentiment analysis, but there are some ways to improve the performance of the algorithm. Researchers indicate that using Maxent model, POS-tagging, and bigram instead of unigram within the sentence can all improve the model[2][3][1], thus, these improvement methods will be utilized in this final project as well.

2 Dataset

Since nltk module has two datasets named "product_reviews_1" and "product_reviews_2", these two datasets were selected as the training set and the test set for implementing the models. Among these two datasets, there are many product reviews for 14 different products:

1. digital camera: Canon G3
2. digital camera: Nikon coolpix 4300
3. celluar phone: Nokia 6610
4. mp3 player: Creative Labs Nomad Jukebox Zen Xtra 40GB
5. dvd player: Apex AD2600 Progressive-scan DVD player
6. ditigal camera: Canon PowerShot SD500
7. digital camera: Canon S100
8. baby bath: Diaper Champ
9. routers: Hitachi router
10. mp3 player: ipod
11. routers: Linksys Router
12. mp3 player: MicroMP3
13. cellphone: Nokia 6600
14. software: norton

A example of the product reviews is below:

"camera[+3],size[+2]##I'm in high school, and this camera is perfect for what I use it for, carrying it around in my pocket so I can take pictures whenever I want to, of my friends and of funny things that happen."

where the string after ## is the a sentence of the product reviews, the number in [ ] represents whether the sentence is positive or negative, and the word before [ ] represents the specific feature of the corresponding number in the [ ].

Since each sentence will have only one tag in this final project, the sentiment tag will be determine by calculating the sum of all the numbers in [] in a single sentence. If the result is larger than 0, then it is positive, but if the result is smaller than 0, then it is negative. We don't consider neutral altitude here.

After preprocessing the dataset, we will have a dataset with sentences which has been marked as positive or negative sentiment tag. Thus, this is a supervised learning in fact. In order to avoid the bias when training the dataset, equal number of positive sentences and negative sentences were selected to form the whole dataset. Then, a 10-fold cross validation will be run on this dataset to calculate the accuracy of different models. According to the result of t-test between different models, the best model will be obtained.

3 Experimental method

The methods used in this final project include one baseline algorithm, naive bayes models and maxent model. Through different ways of feature selection, the performance of these models may vary and the accuracy of all these models with different features selection methods were considered.

3.1 Models

3.1.1 Baseline Algorithm

The baseline algorithm used here is called lexical ratio algorithm. Two dictionaries named "negative-words" and "positive-words" were downloaded from website(website name: Opinion Mining, Sentiment Analysis, and Opinion Spam Detection) and stored into two text files.

When predicating the sentiment label of a sentence, the number of the positive words and negative words were counted at first if the word was also in these two dictionaries. Suppose the count of these two types of words are: $Count(pos)$ and $Count(neg)$.

If $Count(pos) \geq Count(neg)$, the sentiment tag for this sentence is positive(the neural sentiment tag is not considered here). Otherwise, the sentiment tag for this sentence is negative.

3.1.2 Multinomial Naive Bayes Model

In Multinomial Naive Bayes model, all the probability is calculated based on the count of the words in the dataset. The prior probability is below:

$$P(c) = \frac{Count(wordsinclassc)}{Count(wordsindataset)}$$

and the conditional probability is below:

$$P(t_k|c) = \frac{Count(wordt_koccurredinclassc) + 1}{Count(wordsinclassc) + |V|}$$

where V is the vocabulary of the training dataset. Thus, the total probability of the sentence with tag c is:

$$P(c|t_1...t_n) \propto P(c) \prod_{k=1}^{n} P(t_k|c)$$

where $t_1...t_n$ is a sentence with n words.

Through this model, the sentiment tag of a sentence can be predicted.

3.1.3 Bernoulli Naive Bayes Model

In Bernoulli Naive Bayes Model, most of the probabilities were calculated based on the number of sentences in the dataset instead of the words. The prior probability is below:

$$P(c) = \frac{Count(sentencesinclassc)}{Count(sentencesindataset)}$$

and the conditional probability is below:

$$P(t_k|c) = \frac{Count(sentenceswithwordt_kinclassc) + 1}{Count(wordsinclassc) + 2}$$

Through this model, the sentiment tag of a sentence can be predicted.

3.1.4 Maxent Model

MaxEnt Model can be thought as a kind of modification from naive bayes model. There are two significant differences between MaxEnt Model and the Naive Bayes Model:

1. MaxEnt Model considers features more widely, in other words, it can not only works with sequence based feature. If necessary, it can add more features based on the tags of the words like noun or verb, etc.

2. MaxEnt Model applys weight before all the features to indicate whether this feature is significant or not. Then, through some algorithm like gradient decent algorithm to optimize the value of the weight vector. Thus, the result of MaxEnt Model is usually more accurate comparing to the Naive Bayes Model.

Here, in order to use the MaxEnt Model, the MaxEnt classifier in nltk module was utilized with iteration of 50 times each time.

3.2 Features Selection

Features can also affect the performance of the model significantly. According to the reference, most frequent feature is the unigram of the dataset. However, sometimes phrases are also important. Thus, bigram is important as well. Thus, we will implement the model with unigram alone and unigram plus bigram as the features.

After that, in order to add more features, POS tagging were applied with bigram hidden markov model to all the sentences in the dataset. After tagging all the words in the dataset, all the two-words phrase composed by an adjective word and a noun were extracted and selected as features added into the unigram models.

However, too many features will affect the efficiency of the algorithm especially for the MaxEnt Model. Thus, the information gain of all the features were calculated before feeding into the models. Since information gain can represent how useful this feature is for classification, all the features were ranked according to the value of the information gain. 2000 features with highest information gain were selected to feed into the models for classification.

4 Evaluation

4.1 Result and Discussion

For all the models, the evaluation is based on the accuracy of 10-fold cross validation. Thus, after running through all the models, a dataset ("data.xlsx") was built. In this dataset, each column represents the accuracy of a 10-fold cross

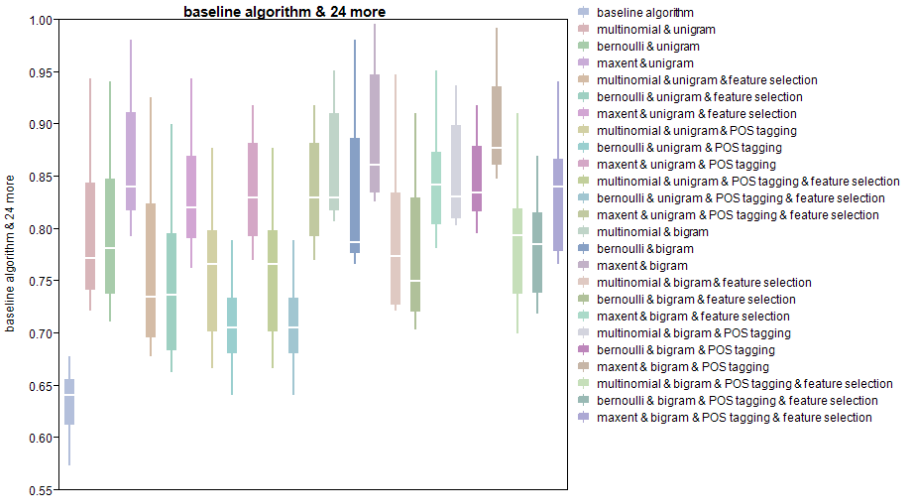

Figure 1: The average accuracy of different models.

Model1	Model2	p-Value
maxent	bernoulli	< 0.0001
maxent	multinomial	< 0.0001
multinomial	bernoulli	0.0478

Table 1: t-test between the models.

validation of a model. Considering different models, feature selection methods and the baseline algorithm, there 25 different columns in total. The average accuracy and the standard deviation is described in Figure 1.

In Figure 1, the leftmost one represent the accuracy of the baseline algorithm(lexical ratio algorithm described in the method section), then the next one represent the Multinomial Naive Bayes Model with unigrams as the features, and the rightmost one represent the accuracy of the MaxEnt model with bigrams and unigrams considering POS tagging and feature selection.

In order to give a more reasonable comparison between the models, we compare these models with t-test:

Naive Bayes Models and MaxEnt model

The result of paired t-test is below(See Table 1): where

- average accuracy of maxent model is: 0.854
- average accuracy of multinomial model is: 0.795
- average accuracy of bernoulli model is: 0.773

According to the average accuracy and the p-value of the t-test, it is obvious that the performance of the maxent model is significantly better than the naive bayes models, but the difference between the performance of the multinomial model and the bernoulli model is insignificant(See Figure 2).

feature types

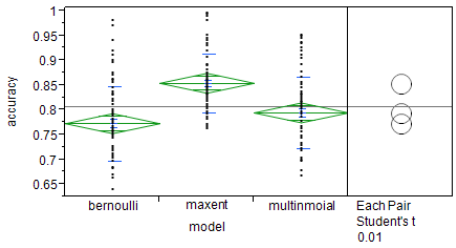

Figure 2: The average accuracy of different models.

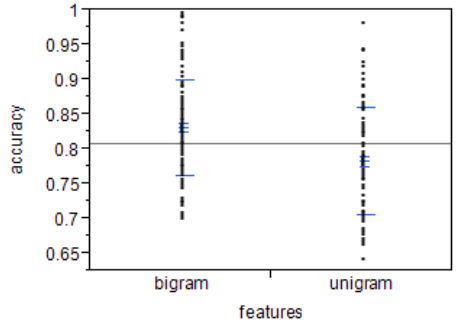

Figure 3: The average accuracy of different feature types.

Since we can use unigrams as features or add bigrams as additional features, it is necessary to compare their performance.

After running the t-test, the p-value is smaller than 0.0001, thus there is significant difference between the performance of these two ways(See Figure 3). Thus, since the accuracy of bigram is higher, it is better to consider bigram when selecting features.

feature selection

In some models, with the increase of the features, the models will become very time consuming, so it is necessary to see whether we can run a feature selection before running the model.

Like the previous analysis, we run t-test on different models to see the performance before and after feature selection. The p-value is smaller than 0.0001, thus, the difference of the performance is significant. Considering the accuracy before feature selection is higher than that after feature selection, so it is not advisable to run feature selection for the sentiment analysis if time allows(See Figure 4).

POS tagging

POS tagging can give all the words a specific tag in the sentence. Then, according to the type of the words, it is possible to extract some specific feature for models. For example, in this final project, if the feature type is unigram, all the two-word phrase formed by an adjective word plus a noun were taken as additional features. If the feature type is unigram and bigram, all the three-word phrases formed by a verb, an adjective, and a noun were taken as additional features.

However, the result of t-test indicate that the difference is not significant,

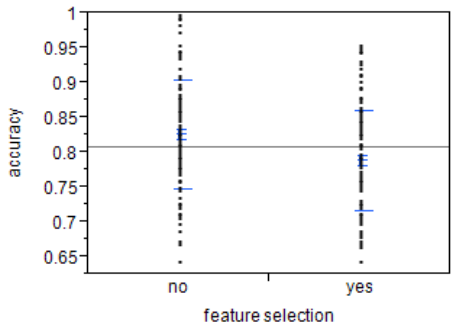

Figure 4: The average accuracy before and after feature selection through information gain.

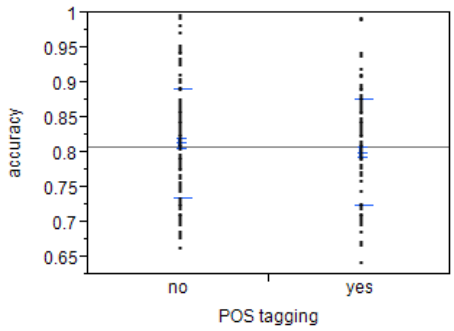

Figure 5: The average accuracy with or without POS tagging.

because the p-value is 0.0921 which is even larger than 0.05(See Figure 5).

Thus, unless some more useful phrases were found as the additional features, it is not necessary to run POS-tagging on the dataset for improving the accuracy.

4.2 sample test

From previous result and discussion, it is obvious that the bigram model without feature selection should have the best performance in predicting the sentiment tag of the sentence.

Thus, baseline algorithm, Multinomial Naive Bayes Model, Bernoulli Naive Bayes Model and the MaxEnt model were implemented with bigram and unigram features together without feature selection. The first fold of the dataset was taken as the the test data and the remained data was taken as the training data.

The accuracy of the corresponding model is below(See Table 2): For the sentence is "I've taken about hundred shot with the Canon, in varying lighting situations, all in auto mode, and not one blurry picture!!! ". After preprocessing the sentence, the predicting result for baseline algorithm was -1, which means it was a negative sentence. However, it was obviously a positive sentence. When we utilize the naive bayes model or the maxent model, the predict results were

Model	accuracy
baseline	0.678
multinomial	0.933
bernoulli	0.970
maxent	0.996

Table 2: accuracy of different models for sample testing.

all correct. This also indicated that our model was more reliable comparing to the baseline algorithm.

4.3 Future Improvement

There are several ways which might be necessary to implement in order to improve the performance of our model.

1. When analyzing the data of unigrams with POS tagging, it was very strange that the accuracy for all models before and after feature selection are exactly the same. However, when analyzing the data of bigrams with POS tagging, the accuracy for all models before and after feature selection are different. It might be caused by some bugs when preprocessing the dataset, or it might be caused by some bugs when calculating the accuracy(such as memory cleaning issue).

2. Though in the discussion, we pointed out that it seems that the POS tagging is not very useful to improve the performance of the sentiment analysis. However, considering the performance is also seriously restricted by the selection of the feature, it is possible to find some better feature combined with POS tagging.

3. Some complicated phrases are not considered like double negation, etc. However, these complicated phrases may affect the performance of the models, especially for the models of unigram features.

5 Conclusion

In this final project, through comparison between different models and the features selection strategy, a reasonable NLP model with proper feature selection method was proposed for sentiment analysis. It is concluded that MaxEnt Model shows the best performance with bigram features. In fact, comparing to the baseline algorithm whose accuracy is 63.5%, the average accuracy for the MaxEnt Model with bigram features is 88.6%. Thus, the future direction for implementing the sentiment analysis should focus on the MaxEnt Model. Since the features combined with POS tagging is too simple in this final project, it is also necessary to consider some features with much more complicated pharses according to the POS tagging result.

References

[1] Oaindrila Das and Rakesh Chandra Balabantaray. Sentiment Analysis of Movie Reviews using POS tags and Term Frequencies. *International Journal of . . .*, 96(25):36–41, 2014.

[2] Xing Fang and Justin Zhan. Sentiment analysis using product review data. *Journal of Big Data*, 2(1):5, 2015.

[3] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02*, 10(July):79–86, 2002.