Ling 406 Term Project: Sentiment Analysis Classifier

Yun Mo Kang

ykang19

5/11/2020

Introduction:

Sentiment analysis is a very popular topic in natural language processing. It can be very useful since it can be used to gain an overview of public opinion on a certain topic. This ranges from a product or a movie review to a review of a clinical medicine. This information can help an organisation or a company to adjust their future products and make necessary adjustments to please the public. Companies can also use this data to gauge public opinion not only on their own products and services, but on their competitors' products and services to stay ahead. Government officials can also use sentiment analysis to adjust their future plans and policies.

Problem Definition:

In terms of computational linguistics, sentiment analysis is the systematic identification, extraction and quantifying subjective information. For this particular project, the aim is to extract the attitude of the author based on their movie review. In short, we are considering the polarity of the text, whether the text reveals positive attitude or negative attitude.

Previous Works:

Many of the previous works dealing with sentiment analysis take a similar approach. First they preprocess the data in many different ways, choose a method of tokenization, then finally train multiple different learning models to see which model yields the best metrics.

In the article by Guha et al (2015), data preprocessing such as stopword, punctuation removal, stemming were used. Their goal was to classify the sentiment in yelp reviews. After preprocessing, they extracted features as unigrams, and instead of using binary values to show the presence of the word, they used TF-IDF score to represent the words. This method is advantageous because it penalizes unhelpful words that are too common across all topics. Then they classified the reviews using many different learning models such as, Stochastic Gradient Descent, SVM, Adaboost multiple times to store confidence scores. Finally they use those confidence scores to build a SVM classifier to classify their data.

In the works of Kiritchenko et al.(2014), sentiment analysis on aspect terms was done. The major difference this work had from other sentiment analysis was their extraction of aspect terms, which are words that explicitly mention a feature or a component of the target product or service. Then they classify the polarity of these aspect terms by tokenizing them into unigrams and bigrams and training a SVM classifier with it.

Some works utilize more complex machine learning models instead of using bag-of-words approach. Santos and Gatti (2014) construct a deep convolutional neural network that utilizes the character-to-sentence level information to perform sentiment analysis on short sentences. This CNN uses two convolutional layers to extract important features from short texts which frequently contain very small numbers of relevant features.

Approach:

In order to find the best method, we have taken the incremental approach. For each iteration, the dataset was applied with stopword and punctuation removal, part of speech tagging to only include adverbs and adjectives, negation marking, and addition of emolex library one-by-one. These preprocessing techniques, excluding the emolex library addition, were done using the nltk library. After preprocessing, the words were tokenized and their frequency was counted. Then out of all the words or tokens that have been collected, only the top 20% most frequent features were extracted. This was done with the frequency distribution method from the nltk library. This dataset was finally put through a set of different learning models to measure the change in accuracy. The learning models considered are: multinomial and Bernoulli naive Bayes, decision tree, logistic regression, and support vector machines. After observing the accuracy output, features would be included or removed depending on if they improved the system or not.

Results:

Below are the table of accuracies calculated during the incremental feature addition. Features that negatively affected the systems were removed during the incremental procedure.

Baseline:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.838	0.826	0.58	0.84	0.844	0.854

Stopword, punctuation removal:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.834	0.784	**0.626**	**0.848**	**0.858**	**0.860**

Stopword, punctuation removal + negation marking:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.806	**0.79**	0.610	0.848	0.840	0.836

Negation marking has decreased the accuracies of almost all learning models so it will not be used in systems below.

Stopword, punctuation removal + Part of Speech tagging:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.812	**0.814**	0.622	0.844	0.850	0.846

Part of speech tagging has decreased the accuracies of almost all learning models so it will not be used in systems below

Stopword, punctuation removal + Emolex library:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.852	**0.816**	**0.628**	**0.868**	**0.864**	**0.874**

Discussion/Conclusion:

Over the course of the project, it was interesting to see that part of speech tagging and negation tagging decreased the accuracy of the system. It seemed like anything that decreased the words in the bag of words negatively influenced the system. Looking at how the addition of the emolex library to the training set increased the performance of the system, it could be generalized that the more training features the system gets, the better it performs.

Some of the flaws of this experiment could be that the training sets and testing sets were shuffled then divided. Perhaps using folding and obtaining the average accuracies of the learning models would have landed more reliable metrics, yet it would increase the computation time.

Project Questions

1. For the first stab approach, a unigram model using a bag of words approach would be appropriate to implement a basic sentiment analyzer. Since the sentiment analyzer has only two emotions: positive and negative, collecting all positive words and negative words and using the naive Bayes unigram algorithm would yield an acceptable result. Since each review can be represented as a bag of independent words Bayes algorithm can be applied ignoring words' position nor their tags, but only their frequency that they appear in the text.

2. To analyze the data, learning models such as Naive Bayes, decision tree, logistic regression, and support vector machines could be used. In this particular project, there are only two classes: positive and negative, and there are a large number of features.

   Naive Bayes, as already mentioned previously, is a simple yet good model to classify the given data. Because of its simplicity, it also has a very fast runtime. If the given dataset were bigger, Naive Bayes would have been a very competitive model.

   Decision trees utilize the tree structure to decide a class for a given feature. Each internal node represents a test on a feature, each branch represents the outcome, then each leaf node represents a class label. This method would not be a strong learning model for this sentiment analysis task, because this classifier is a non-linear one. Thus the task of two-class classification would not be appropriate with this learning model.

   Logistic regression is also very strong in handling two-class identification. It is a generalized linear regression model that determines the probability of something being positive or negative (values between 0 to 1). It calculates that probability by applying a logistic function to a linear combination of data. This method is very suitable when the data is very spread out, and has a lot of noise.

Support vector machines are similar to logistic regression but it is different in that it tries to find the best margin in the data based on the geometric aspects of the data. The algorithm creates hyperplanes that separates given data into different classes. Support vector tries to draw this hyperplane equidistant from the boundaries of the different classes. In that sense, support vector machines may have a slight upper hand in analyzing these reviews since the size of the training set is large and could have many of the same words showing up on both classes. Also, since our dataset is relatively small, runtime for support vector machines will be relatively reasonable.

3. In order to improve the baseline model, many improvements can be made. First, all punctuations and stop words such as articles and conjunctions that are irrelevant to sentiment classification can be removed. With part of speech tagging, only adverbs and adjectives could be considered, since other parts of speech words do not reflect sentiment as much as those two parts of speech. In addition, nouns and verbs can become ambiguous since they can appear equally on each negative and positive document. Negation marking could also be added to enhance the classification, as this could clarify some nuances and ambiguity in text. Lastly, the given emolex dataset can be added to the list of features to reinforce the learning models by adding robustness to the training data.

4. Referring to the report, the best combination of features was punctuation, stopword removal and the addition of emolex library to the training features. These features work the best with the NuSVC learning model as expected. This is because the given dataset only had two classes: positive and negative. The dataset also had features closely associated with each class, since each review tended to have certain words appear more frequently than the other.

Extra Credit: Yelp Reviews

The sentiment analysis of the Yelp reviews is essentially the same as the sentiment analysis of the movie reviews. If the Yelp reviews had to be classified by one of the five stars, this task would have been very complex. However, the task was simplified by collapsing all reviews above three stars to positive, and below three and lower stars into negative. The main difference is that these reviews are generally shorter than the movie reviews and maybe harder to extract important features since they may only have one or two words that have much influence on sentiment analysis. Otherwise the problem is the same as the previous one: polarity sentiment analysis. Therefore the same preprocessing: stopword, punctuation removal, negation marking, part of speech tagging to only consider adjectives and adverbs, and the addition of emolex library will be done. The same leaning models will be used to train: naive Bayes, logistic regression, decision trees, and support vector machines. The only concern that arises with this task is that the support vector machines may take very long time to process the data. Since the dataset for Yelp reviews is much larger than the movie reviews, the runtime for support vector machines will increase tremendously. Support vector machines' kernel matrix requires memory that scales quadratically with the number of data points.

Baseline:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.838	0.722	0.778	0.862	0.866	0.865

Stopword, punctuation removal:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.835	**0.751**	**0.781**	0.854	0.861	0.863

No significant improvement was found, so this feature was not used.

negation marking:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.830	0.715	0.766	0.847	0.863	0.863

No significant improvement was found, so this feature was not used.

Part of Speech tagging:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.837	**0.747**	**0.790**	0.837	**0.871**	**0.870**

No significant improvement was found, so this feature was not used.

Emolex library:

Multinomial NB	BernoulliNB	Decision Tree	Logistic Regression	SVC	NuSVC
0.840	**0.726**	**0.780**	**0.864**	**0.864**	**0.874**

The addition of the emolex library to the training set improved the overall accuracy.

Result:

Compared to the movie reviews, removing stopwords and punctuations was not helpful for the Yelp reviews. Only the addition of the emolex library increased the accuracy of all learning models. But the results were oddly too high. The expectation was that since these Yelp reviews are shorter than the movie reviews, the learning models would have more difficult time extracting features. Perhaps the use of top 20% most frequent vocabulary has helped the extraction of most important features. In addition, it is not very practical to use support vector machines for this dataset, since the large size of the dataset does increase the computation time tremendously. On average, SVC and NuSVC took about 10 minutes to run while logistic regression took only 2-3 minutes. Meanwhile, their accuracies don’t differ by much, so it seems more sensible to use logistic regression.

References:

Cicero Nogueira dos Santos and Maira Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In Proc. COLING.

Satarupa Guha, Aditya Joshi, Vasudeva Varma. SIEL: Aspect Based Sentiment Analysis in Reviews. Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 759–766,Denver, Colorado, June 4-5, 2015.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif M. Mohammad. NRC-Canada-2014: Detecting Aspects and Sentimentin Customer Reviews. Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 437–442,Dublin, Ireland, August 23-24, 2014.