LING 406 Term Project: Sentiment Classifier

Cheng-Yang (Peter) Liu

May 8, 2017

1 Introduction

To understand the opinions within the text becomes an interesting domain, which is often called opinion mining or sentiment analysis, for linguistics and computer scientists. Nowadays, with people actively "living" on Internet, a massive amount of textual data, such as reviews, has been available. Also, with the storage and computational power increase, we are more capable to deal with this amount of data today.

From the application aspect, to know what people think about is always a critical part when we make a decision. For example, we would ask people to recommend a restaurant for dinner, a dealer for buying a car, or even the future of stock market. By the Internet and Natural Language Processing (NLP) techniques, we now can gather more than thousand of millions opinions, and give a summary or conclusion of those words.

There are many applications raise in the past few years in different areas. In business, companies are able to know the reasons why specific products have bad sales by Amazon reviews, and improve it. Or, for the government election, candidates are able to track the opinions in certain cities or sates from Twitter, and hold more campaigns in that area.

Therefore, this paper, as a term project of Introduction to Computational Linguistics, University of Illinois at Champaign-Urbana, it presents an empirical study of the effect from feature selection and machine learning classifier on sentiment analysis task. We implement four features selections methods, stopwrods removal, negation consideration, adjective and adverb assembly, and stemming, to see the linguistics characteristics on this problem. On the other hand, we choose four machine learning classifiers, Naïve Bayes, Logistic Regression, Decision Tree, and Support Vector Machine for the classification.

For evaluation, we adopt two datasets. One is the movie review dataset, consisting of 1000 positive reviews and 1000 negative reviews, found at http://www.cs.cornell.edu/people/pabo/movie-review-data/, Pang et al (2002). The other is the reviews from Yelp of restaurants in the Champaign-Urbana area, originally created by John Hall, in LING 406 Spring 2016. This later dataset contains 10391 reviews, each with different rating from 1-star to 5-star.

2 Problem Definition

As we discuss previously, sentiment analysis or opinion mining try to reveal the attitude in the text, which can be words, sentences or documents. The attitude could be various, such as subjectivity (subjective or objective), polarity (positive or negative), or a scale from 1-star to 5-star.

This task can be approach from two different view points, computational linguistics (natural language processing) or machine learning (statistics). In terms of computational linguistics, we try to find the pattern from the language itself, and generate features to represent the words in a simplest way. For machine learning, we try to classify the given feature to different classes by their mathematical meaning.

The focus of our project is analyzing of the sentiments in movie reviews and restaurant reviews. They are polarity and ranking problem respectively.

1. polarity – The inputs of polarity problem are texts, which could be words, sentences, or documents. Here we have short documents as reviews. And the output are two classes whether they are positive reviews or negative reviews.

2. ranking - For the ranking problem, it has the same input as polarity. However, it tries to label every testing data with a range of numbers.

In this project, we try to discuss language features deeply in order to find a better representation for this task. The possible improvements will also be studied in the discussion and conclusions section. There are no best feature representation nor useless feature.

3 Previous Work

Sentiment analysis has been broadly studied for the past few years. Large amount of Internet context provides researchers ability to create their own dataset to do analysis and training. After the datasets are established, linguistic feature and machine learning technique then come into spotlight for the area.

Pang, Lee and Vaithyanathan [1] use the data of movie review. They apply three machine learning techniques, Naïve Bayes, Maximum Entropy, and Support Vector Machine, and eight different statistical feature selection methods. The result shows that, different from other Natural Language Processing task, the present or absent of a word has a better indication than the frequency of a word. Also, bigram model does not perform better than unigram, Bag-of-Word model.

For the feature selection, some linguistic characteristics are interesting to discuss. A well-known method to reduce the noise of textual data is the removal of stopwords. In [2], they study the effect of stopwords on data sparsity based on Twitter dataset. In the paper, they have six different stopwords removal methods. At the end, the results show that Naive Bayes classifiers are more sensitive to stopword removal than the Maximum Entropy ones.

Part-of-Speech tags are also a popular feature that researches take into consideration. Altough most of the work focuses on nouns, verbs and adjectives, [3] brings up the idea of combination of adjectives and adverbs on sentiment analysis task. Adverb shows strong effect on the strength of a given sentiment. Their results conclude that adjectives indeed are more important than adverbs. However, integrating with adverbs, they can get a higher accuracy than using adjectives alone.

4 Approach

4.1 Feature Representation

4.1.1 Baseline: Bag-of-Words (BoW) Model

Bag-of-Words model takes words individually in a dictionary as the feature representation. Each feature in a feature set considers whether the corresponding word exists or not. This can also be equivalent to unigram model. The challenges are the feature selection problem.

In the following section we do not use the bigram or other N-gram models. Because the result in [1] does not show the advantage of adopting them in sentiment analysis task. Also, in the classification, we apply 5 fold cross validation for avoiding bias.

1. Feature Size:
   First question is, how many words/features should we include into our feature set? In other words, here we are deciding the feature size. Figure 1 shows that the accuracy increase while features size grows for the baseline model. However, the accuracy boost slowly when the feature size reach a certain point. On the contrary, the computational time increases rapidly, while the feature set gets larger. In the experiments, we choose 100, 300, 500, 700, 900, 1000, 3000, 5000, 7000, 9000, 10000, 20000, and 30000 features to plot the comparison between sizes.

2. Feature Collection:
   Next question, is every word in the feature set important to our purpose? The feature set contains a lot of noisy information. In this section we will experiment several feature selection strategies, and as well as the last section, we use Naïve Bayes classifier to test different feature collection effect.

   - Stopwords Removal: Stopwords is a list of word that are not discriminative, such as "of" and "the". Theoretically, removing stopwords reduces the noise of textual data and the feature space, and produce more accurate results. This approach is widely implemented in NLP tasks, such as document classification. In this experiment, I use pre-compiled stopword lists in NLTK library.

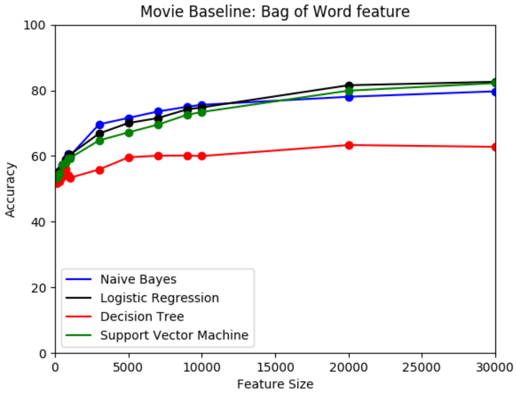

Figure 1: Accuracy of different Bag-of-Words feature size with baseline model.

- Negation Consideration: In sentiment analysis, negation is critical to determine polarity. For example, "This is not a good movie" has a total opposite meaning from "This is a good movie". However, for our BoW model they have similar feature representation with respect to a huge feature set. Therefore, this part of experiment take the negation into concern. A popular implementation is to create new features by adding "not" to original features, if the words are follow by a negation word.
- Adjective and Adverb Assembly: Adjectives and adverbs are good indicators of evaluating subjective sentences. For instance, in the sentence "a very cool ideal", the adjective, "cool", express the polarity of this sentence; and the adverb, "very", illustrate the degree of this tendency. Hence, We evaluated the performance with only adjectives and adverbs are considered as features.
- Stemming: Stemming is a prevailing preprocession technique for feature extraction task. It converts all the inflected words into a simplfied form, stem. By this method, we can reduce the complexity of feature space and the sparsity of the data. We adopt Porter Stemmer [4] provided by NLTK library.

4.2 Machine Learning Classifier

4.2.1 Naïve Bayes

Naive Bayes classifier applied Bayes' rule on the input text. The assumption is that each word is independent to others in the feature set. It then calculate the prior probability for each feature in the training data. For testing, it estimate the likelihood probability from the given testing sentence, and assigned the polarity with highest likelihood estimate.

4.2.2 Logistic Regression

On the contrary, logistic regression classifier compute parameters that directly maximize the likelihood of the training corpus. In other words, Naïve Bayes classifier is a specialy case of logistic regression classifier. The intuition is that the words with higher weights indicating better features.

4.2.3 Decision Tree

The decision tree tries to sort the input data by building a classification trees. At each level, it selects features as split nodes, and sorts the sentences by the present of words. The decision is made when the sentence reach the leaf nodes of the tree.

4.2.4 Support Vector Machine (SVM)

SVM is a method to classify both linear and nonlinear data. It increases the dimension of the training data, and try to find a linear optimal separating hyperplane in the higher dimension. Additionally, the SVM finds hyperplane by using support vectors, which are the significant features in the feature set.

Feature Size	NB	LR	DT	SVM
Pure BoW	*69.65%*	66.85%	55.90%	64.75%
Stopwords	*70.35%*	67.80%	60.40%	65.15%
Negation	*66.75%*	63.95%	56.65%	62.95%
Adj. + Adv.	*70.80%*	70.75%	**60.50%**	66.95%
Stemming	***73.75%***	**72.50%**	57.85%	**70.00%**

Table 1: Small datasets: accuracy of four feature methods comparing with four machine learning techniques. NB: Naïve Bayes, LR: Logistic Regression, DT: Decision Tree, SVM: Support Vector Machine. Boldface numbers are the highest accuracy amount all feature selection methods for one machine learning classifier. Italic numbers are the highest accuracy amount all machine learning classifier for a fix feature selection method.

Feature Size	NB	LR	DT	SVM
Pure BoW	79.70%	*82.60%*	62.80%	82.20%
Stopwords	78.25%	*84.20%*	62.05%	82.45%
Negation	76.75%	*81.95%*	61.25%	80.90%
Adj. + Adv.	**80.00%**	*81.30%*	**63.05%**	79.45%
Stemming	79.45%	***84.50%***	61.85%	**83.45%**

Table 2: Large datasets: accuracy of four feature methods comparing with four machine learning techniques. NB: Naïve Bayes, LR: Logistic Regression, DT: Decision Tree, SVM: Support Vector Machine. Boldface numbers are the highest accuracy amount all feature selection methods for one machine learning classifier. Italic numbers are the highest accuracy amount all machine learning classifier for a fix feature selection method.

5 Result

The evaluation will be discussed in two parts. First, we have four BoW feature collection methods. Although the best performance appears in the pure BoW, we did a deep research of finding the reason why the performance does not improve for each added feature. Second, we have four machine learning classifiers in total. As well, we study their natural characteristics, and conclude the best learner in our experiments of the task.

Since we know the performance increase along with the size of features, we use two amounts of feature, 3000 and 30000, for convenience. The two sizes represent small and large feature sets respectfully. As a result, we can see that some of the modification have better accuracy in larger feature set. Table 1 and Table 2.

1. Bag-of-Words Feature Collection:
   In general, adding those features improve the accuracy. However, it is worth to study deeply of these result, in order to improve the system in the future.

   Considering adjectives and adverbs, and stemming improve the most amount four feature selections both in small dataset and large dataset. Stopwords does not have relatively better performance in large dataset. Adding negation does not improve the accuracy.

   Please refer Table 1 and Table 2 for the discussion of this section.

   - Stopwords Removal: The difference of removing stopwords is not significant. Furthermore, the accuracy drops. For sentiment analysis, whether removing stopwords is a effective operation has been debated for few years. Although in other application, such as document classification, removing stopwords discards non-discriminative words, reduces the feature space, and produces more accurate results, from our observations, it might carry sentiment information and removing them could have negative effect on classification performance. In [5], also shows that classifiers learned with stopwords outperform those learned without them.
   - Negation Consideration: In order to deal with the effect from negation, we create new features which related to negative words. However, the shortcomings is that it makes one words into two entities. This approach not only increase the total feature space, but also increase the portion of non-polar expression words. (since we add the negation

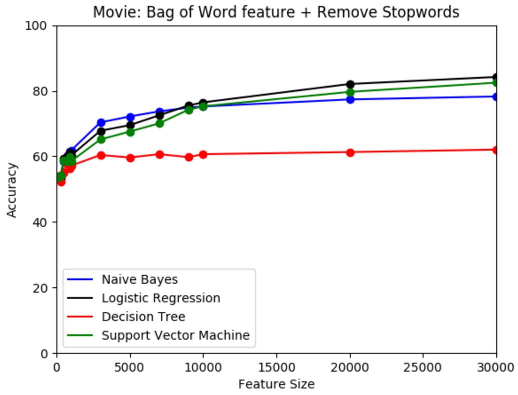

Figure 2: Accuracy of different Bag-of-Words feature size with removing stopwords.

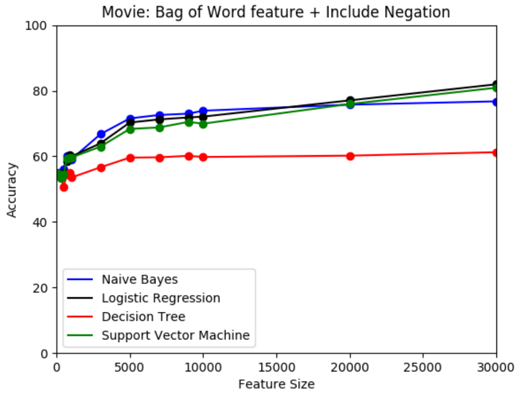

Figure 3: Accuracy of different Bag-of-Words feature size with considering negation.

sign to every words after a negative words). [6] Therefore, the result does not increase, but drop a small amount of accuracy.

- Adjective and Adverb Assmebly: Since adjectives and adverbs have been emphasized in previous work in sentiment analysis [3], we experiment the performance with only adjectives and adverbs. Intuitively, adjectives carry sentiment information, and adverbs carry the intensity of the expression. The result does improve well. The method reduces the feature space. So if we include the same amount of feature, this method would consider more information.
- Stemming: The result of using stemming increase in the small dataset, while decrease in a larger feature set comparing to baseline. The reason is that the stemmer is a set of rules, which could destroy words' semantic meaning. Therefore, in a larger dataset some of the unique feature might be shorten to some other words with different meaning. For example, it could map two words with opposite polarity into the same stemmed form. Such as, "captivation" and "captive" are both stemmed into "captiv" by Poter Stemmer. [7]

2. Machine Learning Classifier:
The Table 1 and 2 also shows the results from different machine learning classifiers with same feature settings.

For the small dataset, Naïve Bayes has the best performance amount all classifiers. However, in a larger dataset, logistic regression and SVM have better accuracy. In general, decision tree do a worst job on this task.

- Naïve Bayes: In the small feature set, Naïve Bayes perform the best. However, in a larger feature set, logistic regression runs in front of it. Worth to mention Support

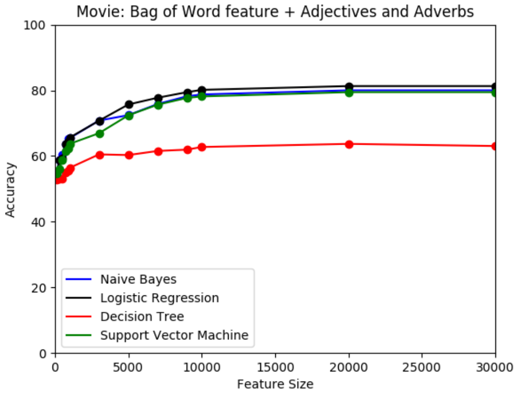

Figure 4: Accuracy of different Bag-of-Words feature size with only adjectives and adverbs.

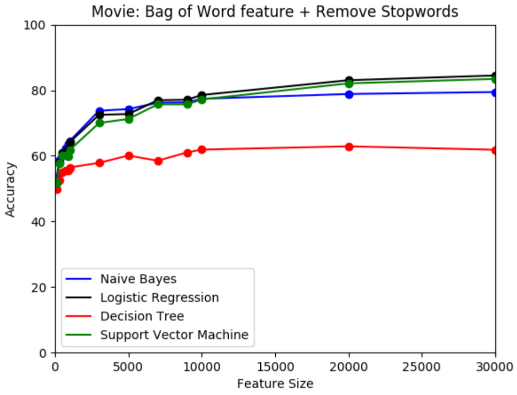

Figure 5: Accuracy of different Bag-of-Words feature size with stemming.

Vector Machine also perform better than Naïve Bayes learner in the large feature set. The conditional independence assumptions of Naive Bayes ignore the relation between two features which overestimating the evidence. The problem become significant when the feature set includes more words.

- Logistic Regression: Intuitively, as a discriminative model, logistic regression trys to directly maximize the likelihood of the training corpus. Also, logistic regression model correlation between features by the weight between them. The more two features are correlated, the closer their weights are assigned. Therefore, if we have enough features, the improvement from the correlation between words appears compellingly. [8]
- Decision Tree: Decision tree has the poorest performance amount all classifiers, even with large feature set. The main reason is that decision tree try to find the optimal result by creating over-complex tree. With the default unlimited layer tree, the decision tree classifier tend to over-fit the training data.
- Support Vector Machine: As well as logistic regression, SVM is a linear classification or regression algorithm. The advantage of SVM is that it can deal with high dimensional input, while not assuming its conditional independence. However, it does not model the correlation between words closely as logistic regression.

6 Discussion and Conclusions

In the discussion and conclusion section, I will focus on how to improve the feature collection methods, instead of machine learning classifier.

Previous experiment implement four feature preprocessing techniques. However, they are the simplest approaches by default setting from NLTK. There are possibilities to improve them according to their linguistic characteristics.

- Stopwords Removal: Pre-compiled lists of stopwords are used in the experiment. It does not care about whether the stopwords contain sentiment meaning or not. A possible fix could be applied dynamic generation of stopwords lists. [2] By removing infrequent stopwords, we could maintain the performance while reducing the sparsity and feature space.

- Negation Consideration: The purpose of adding this feature is to detect the reverse polarity situation. However, simply creating negation feature for every words after a negative cue does not work in the case. It is a complex topic in linguistic. Tottie [8] presents negation categories, such as denials, rejections, and questions. Therefore, a future work could be constructing a negation system that can identify different forms of negation, and discard the misclassified words. [What's Great and What's Not: Learning to Classify the Scope of Negation for Improved Sentiment Analysis]

- Adjectives and Adverbs Assmebly: A possible path to improve the performance from collecting adjectives and adverbs is to consider their semantic meaning. Hatzivassiloglou & Wiebe [9] studied the effects of dynamic adjectives, semantically oriented adjectives, and gradable adjectives on a subjectivity classifier. We can also implement the restriction both on adjectives and adverbs as well.

- Stemming: Stemming is added in order to decrease the feature space, not only to speed up classification, but also increase the accuracy. However, stemmer sometimes breaks the semantic meaning of words. A better approach is to consider lemmatization instead of stemming. For instance, "caring" and "cars" have the same result for stemming, which is "car". On the other hand, lemmatization results in "care" and "car", which is a better representation. [10]

From this project, I realized two main issues for dealing with computational linguistics or natural language processing problem. First, feature representation is important, although the amount of data is also required. In other words, finding the natural characteristics is the first important step to start. Second, there is no perfect universal machine learning method. For example, for a lot of application, decision tree is a better method, since it is flexible to the feature. However, for sentiment analysis, decision tree is a poor classifier due to its flexibility tend to over-fit the training data.

Feature Size	NB	LR	DT	SVM
Pure BoW	32.63%	37.11%	36.74%	*37.61%*
Stopwords	33.97%	36.79%	36.7%	*37.03%*
Negation	32.11%	36.66%	36.28%	*36.75%*
Adj. + Adv.	**35.59%**	**39.14%**	**36.95%**	***39.30%***
Stemming	32.57%	*38.00%*	36.60%	37.67%

Table 3: Small dataset: accuracy of four feature methods comparing with four machine learning techniques for Yelp's reviews. NB: Naïve Bayes, LR: Logistic Regression, DT: Decision Tree, SVM: Support Vector Machine. Boldface numbers are the highest accuracy amount all feature selection methods for one machine learning classifier. Italic numbers are the highest accuracy amount all machine learning classifier for a fix feature selection method.

Feature Size	NB	LR	DT	SVM
Pure BoW	33.60%	*40.75%*	37.11%	39.80%
Stopwords	**35.70%**	*41.06%*	**37.51%**	39.90%
Negation	34.08%	*39.79%*	37.16%	39.39%
Adj. + Adv.	35.68%	*39.97%*	37.41%	39.89%
Stemming	33.52%	***41.84%***	**37.51%**	**41.07%**

Table 4: Large dataset: accuracy of four feature methods comparing with four machine learning techniques for Yelp's reviews. NB: Naïve Bayes, LR: Logistic Regression, DT: Decision Tree, SVM: Support Vector Machine. Boldface numbers are the highest accuracy amount all feature selection methods for one machine learning classifier. Italic numbers are the highest accuracy amount all machine learning classifier for a fix feature selection method.

7 Extra-credit: Yelp's reviews

In this section we will run our baseline and the improved systems on a different dataset: Champaign-Urbana Yelp restaurant reviews (a collection of 10391 reviews of restaurants in the Champaign-Urbana area scraped from Yelp by John Hall). We also generate same tables and figures, and compare them as previous sections. Since, this dataset turns our problem from binary classification to multi-class classification, the accuracy drops overall. Due to the computational power, the maximum feature size is 10000 features. The small dataset is still applying 3000 features.

7.1 Baseline:

For the baseline, different from the polarity problem, Naïve Bayes perform the worst for both small and large datasets. Also, decision tree perform not bad in this case. It is possibly due to the multi-classification problem increase the complexity of the problem, which make decision tree less possible to over-fit.

7.2 Improved System:

Considering selected feature scenarios, adjectives and adverbs collection perform best in the small datasets, but stopwords and stemming get higher accuracies. The reason might be people are using similar adjectives and adverbs in Yelp, so the large dataset does not give more information (food diversity might be lesser than movie's).

Comparing to previous dataset, stemming perform generally well, even though it has it's own problem. However, stopwords has a good influence on the Yelp's dataset. It could be the reason of the complexity of the language meaning in restaurant reviews is less complicated than in movie reviews. Also it could just because we have not enough feature to describe the multi-class problem.

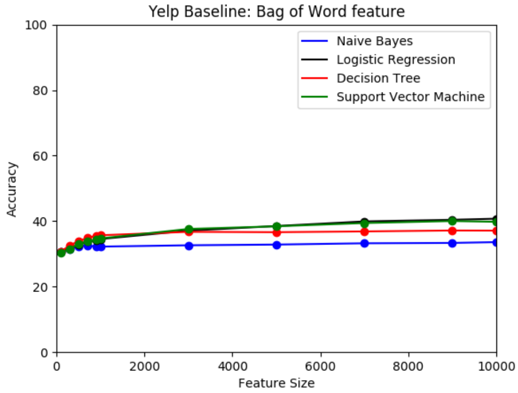

Figure 6: Accuracy of different Bag-of-Words feature size with baseline model, Yelp's reviews.

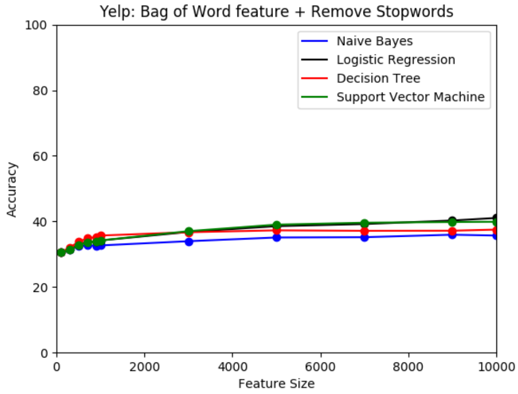

Figure 7: Accuracy of different Bag-of-Words feature size with removing stopwords, Yelp's reviews.

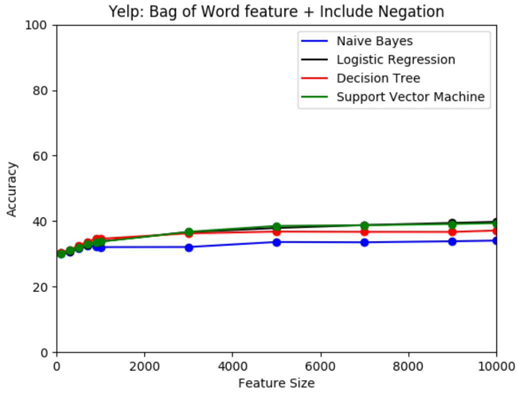

Figure 8: Accuracy of different Bag-of-Words feature size with considering negation, Yelp's reviews.

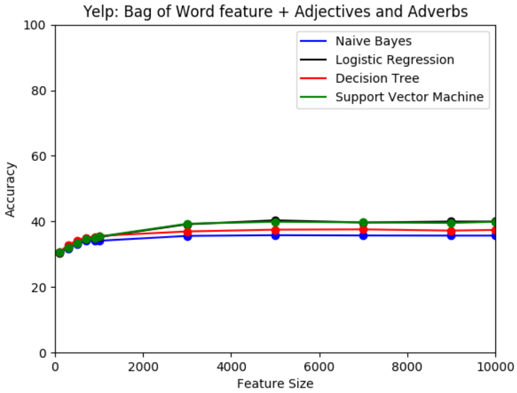

Figure 9: Accuracy of different Bag-of-Words feature size with only adjectives and adverbs, Yelp's reviews.

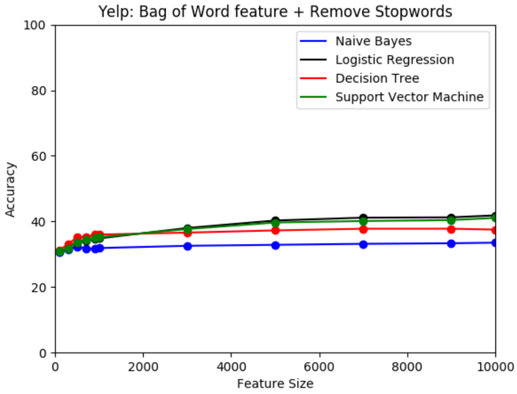

Figure 10: Accuracy of different Bag-of-Words feature size with stemming, Yelp's reviews.

References

[1] Pang, Bo, 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-Volume 10. Association for Computational Linguistics, 2002.

[2] Saif, Hassan; Fern´andez, Miriam; He, Yulan and Alani, Harith. *On stopwords, filtering and data sparsity for sentiment analysis of Twitter.* In: LREC 2014, Ninth International Conference on Language Resources and Evaluation. Proceedings., pp. 810–817.

[3] Benamara, Farah, Carmine Cesarano, Antonio Picariello, Diego Reforgiato Recupero, and Venkatramana S. Subrahmanian. *Sentiment analysis: Adjectives and adverbs are better than adjectives alone.* In ICWSM. 2007.

[4] Porter, Martin. *The Porter Stemming Algorithm.* http://www.tartarus.org/˜martin/PorterStemmer, 2002.

[5] Saif, Hassan, Yulan He, and Harith Alani. *Semantic sentiment analysis of twitter.* The Semantic Web–ISWC 2012 (2012): 508-524.

[6] Wiegand, Michael, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. *A survey on the role of negation in sentiment analysis.* Proceedings of the workshop on negation and speculation in natural language processing. Association for Computational Linguistics, 2010.

[7] Potts, C. *Sentiment Symposium Tutorial.* Retrieved May 1, 2017, from http://sentiment.christopherpotts.net/

[8] Jurafsky, Dan, and James H. Martin. *Speech and language processing.* Vol. 3. Pearson, 2014.

[9] Hatzivassiloglou, Vasileios, and Janyce M. Wiebe. *Effects of adjective orientation and gradability on sentence subjectivity.* Proceedings of the 18th conference on Computational linguistics-Volume 1. Association for Computational Linguistics, 2000.

[10] Asghar, Muhammad Zubair, Aurangzeb Khan, Shakeel Ahmad, and Fazal Masud Kundi. *A review of feature extraction in sentiment analysis.* Journal of Basic and Applied Scientific Research 4.3 (2014): 181-186.