Sensitive Analysis based on Twitter

Handan YU

1 Introduction

Sentiment analysis is one of the most popular application areas of NLP (Natural Language Processing). Recently, an increasing number of sentiment analysis researches have been done on Twitter Data by numerous researchers. Twitter is a popular microblogging platform where users are willing to post their real time messages, including reactions, status, and opinions.

In this report, three types of sentiment classifiers were built to deal with a 3-way task of classifying tweet sentiment into positive, neutral, and negative. These classifiers were based on three different machine learning algorithms, namely Naive Bayes, Softmax Regression, and K Nearest Neighbor. Also, these classifiers were separately examined on both traditional TFIDF and the improved TFIDF combined with information entropy feature selector. Furthermore, the impacts of gender bias on the performances of classifiers were discussed.

2 Literature Review

Some relatively earlier researches on Twitter Data only focused on binary classification. For instance, [Go et al., 2009] used a distant supervision learning approach to automatically obtain the sentiment of Twitter messages. They also applied machine learning algorithms (Naive Bayes, Maximum Entropy, and Support Vector Machines (SVM)) to classify the Twitter Data as either positive or negative concerning the query terms, and they indicated the performance of SVM is exceeded than other classifiers'. Furthermore, in terms of feature extractors, they illustrated that the unigram feature extractor improved the accuracy of all models compared with bigrams, and the combination approaches. Also, they reported that POS features were not useful.

Moreover, other researchers work on multi-class sentiment classification. [Pak and Paroubek, 2010] created a specific corpus by collecting tweets using Twitter API and linguistic analysis of those tweets. They also used TreeTagger for POS-tagging to observe the difference in distributions among three sentiment data sets. The classifier they developed was based on the multinomial Naive Bayes algorithm that uses features, including N-gram and POS-tags. [Agarwal et al., 2011] investigated tree kernel and feature based models. In terms of feature selections, they concluded that both the tree kernel and the hybrid model outperformed the unigram model in both binary and multi-class classification tasks.

3 Data Description

The original data in this report is derived from the resources published in [Vadicamo et al., 2017] and [Go et al., 2009]. Additionally, the training data and hold-out testing data used in my experiments are provided by the course teaching team which was obtained from the original raw tweets using TFIDF algorithm. Specifically, these data are processed from the raw data through filtering the very frequent and very infrequent words unsupervised concerning TFIDF values of words. Also, the words that remained in the final data were the top 5000. Both data sets include Tweet IDs, their corresponding keywords, mapping TFIDF values, and sentiment labels.

4 Baseline

In this report, two kinds of baselines were illustrated.

The first baseline would be established based on Zero-R. According to the characteristics of the training data set which are shown in Table.1, The most common class in the training data is negative, and therefore this classifier returns negative polarity. (i.e., the accuracy of this baseline is 59.33%)

The other is based on the opinion lexicon, which is to use a dictionary of opinion words to identify sentiment orientation. The dictionary is publicly available[1] and can be downloaded through Python directly. The dictionary consists of 2006 positive words and 4783 negative words. For each training tweet, the number of negative keywords and positive keywords that appear were counted. This classifier then returns the class with the higher count. If there is a tie, then class 'neutral' will be returned. This approach is simple but efficient, thus applicable to this task as well. (i.e., the accuracy of this baseline is 67.08%)

Sentiment	Number of Samples	Percent
Positive	62447	39.2%
Neutral	31934	20.1%
Negative	64872	40.7%

Table 1: The categories of the training dataset

Baseline	Accuracy
Zero-R	59.33%
Opinion lexicon	67.08%

Table 2: The accuracy of baselines

5 Method

In this section, the method of this research is illustrated.

5.1 Feature Selection

The size of the training data set processed by TFIDF feature selection is still large, which seriously impacted the implementation of model training. Therefore, taking advantage of text keywords properties, removing some feature words that are independent with sentiment class is of great importance to reduce feature space.

5.1.1 TFIDF with information entropy

Firstly, the traditional TFIDF was obtained using unsupervised, which ignored the distribution of feature words among classes. Therefore, to a certain degree, using traditional TFIDF would affect the accuracy of sentiment analysis greatly. The motivation was from work by [Yantao Zhou and Wang, 2007]. They analyzed a new TFIDF feature selection approach with the concept of information entropy, which was valid in improving the accuracy of text categorization using the similarity method.

In this paper, this updated TFIDF feature selection method would be applied to guarantee each feature word's TFIDF is reasonably closed to the real data, and also remove some of the words which evenly distributed in each class. The new TFIDF can be computed concerning the following mathematical formula.

$$TFIDF' = TFIDF \times \log(\frac{1}{I_k(p) + \phi_k(1)} + 1),$$

where $I_k(p)$ is the information entropy of word k in the training data, and $\phi_k(1)$ is the second smallest information entropy of word k based on the number of word k count in the whole training data (i.e., $\phi_k(1) = -\frac{n_k-1}{n_k}\log\frac{n_k-1}{n_k} - \frac{1}{n_k}\log\frac{1}{n_k}$, n_k refers to the number of instances containing word k).

5.2 Evaluation

In this report, accuracy and F1-Score will be chosen as the evaluation metrics to evaluate the classifiers' performances. In this task, a good classifier should have an excellent performance on three types of sentiment rather than a specific type of sentiment. Also, in theorem, the higher precision leads to the lower recall. However, a perfect classifier is always required to have a high precision and a high recall as well. F1-Score actually deals with this confusion.

$$F1 = \frac{2 \times precision \times recall}{recall + recall}$$

Additionally, the approach of calculating precision is micro-averaging, because the purpose of this task does not focus on a certain type of sentiment, hence all kinds of sentiments are equal. Meanwhile, the micro-averaging performances of precision is equal to the accuracy.

$$F1' = \frac{2 \times accuracy \times recall}{recall + recall}$$

The performances of different classifiers built in this paper would be evaluated on the labeled development sets, which were split by a holdout strategy.

5.3 Training classifiers

5.3.1 Selection of classifiers

One of the simplest and intuitive machine learning algorithm is the K Nearest Neighbor. This

[1]The dictionary is linked off of `https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html`

was selected because it is an instance-based learning algorithm. Also, it supports multi-class classification. Another common classification algorithm is the logistic regression. Since it is easy to explain complex problems due to its simple solutions. Moreover, it has no restrictive assumptions on features and is particularly suited to frequency-based features. However, the task is three classification problem, the generalization of logistic regression algorithm, that is Softmax Regression would be chosen to build a sentiment classifier. On the other hand, considering that Naive Bayes, a probabilistic generative mode, takes care of the distribution and frequency of the words presented in the tweets and ignores their positioning corresponding to each other, it is reasonable to train a sentiment classifier.

Consequently, the sentiment classifiers were built using K Nearest Neighbor (KNN), Softmax Regression, and Naive Bayes (NB) machine learning algorithms. Additionally, each of the classifiers is trained using two different feature sets.

5.3.2 Data processing

Considering that the feature matrix is a sparse matrix where contains a number of zeros and its variance is extremely small, the Min-Max normalization rather than standardization would be taken to process the training data before training classifier. The reason is that using Min-Max normalization can not only remain the 0s in the sparse matrix, but also help to increase the stability of the feature matrix.

Additionally, the processed training data would be used in both Softmax Regression and KNN. Because the key point of Softmax Regression is to find out the optimal parameters using gradient descent, if without normalization of data, it will take the amount of time to converge. Furthermore, as for KNN, it is a model based on calculating the distance between every two instances. Therefore, theoretically, normalization will improve the performances of classifiers based on LogisticRegression and KNN, respectively. However, in terms of Naive Bayes, there is no need to use processed data to train this type of classifier. Since it is a probability model which focuses on the distribution of data rather than real values of data.

5.3.3 Setting parameters

In order to prevent overfitting or underfitting, some of the certain parameters in different models have been set reasonably. As for Softmax Regression, selecting a best regularization coefficient is necessary. Also, for Naive Bayes, it is easy to occur Zero probability problem due to the sparse feature matrix. Moreover, in terms of KNN, finding a suitable neighbor size is essential to build a perfect model. Therefore, in this paper, these parameters will be searching by GridSearchCV. Finally, in this experiment, the regularization coefficient of Softmax Regression (C) assigns to 10, and then smoothing coefficient of NB (α) assigns to 10^{-5}. Also, the neighbor size of KNN (k) assigns to 5.

6 Results

Three types of sentiment classifiers which are separated based on Softmax Regression, Naive Bayes (NB), and K Nearest Neighbor (KNN) will be examined on traditional TFIDF and new TFIDF feature data respectively. The results would be shown as follows:

Features	NB	Softmax	KNN
Original TFIDF	0.72405 *	0.73972 *	**0.59887**
Updated TFIDF	**0.72546***	**0.74218***	0.59670

Table 3: The accuracy of classifiers. The best results are in **bold**. Results are marked with * if they are significantly better than baselines in this paper.

Features	NB	Softmax	KNN
Original TFIDF	**0.74147**	0.75596	**0.60528**
Updated TFIDF	0.74100	**0.75834**	0.60239

Table 4: The F1-Score of classifiers. The best results are in **bold**.

Traditional TFIDF. The TFIDF feature extracting approach is the common way to indicate features from tweet. Also, in this experiment, it is obvious to find that Naive Bayes and Softmax Regression algorithm outperform than all of the baselines in this paper. Specifically, the performance of Softmax Regression algorithm (73.972%) was best among the three classifiers, following by the Naive Bayes classifier (72.405%). The little loss of Naive Bayes classifier might be due to its naive assumption

(i.e., all of the features are conditionally independent). Meanwhile, the accuracy of KNN classifier was no match for that of Opinion lexicon baseline. The main reason might be the size of neighbors.

Updated TFIDF. The TFIDF based on information entropy can better reflect features of each tweet. As the gentle increase of accuracy and F1-Score (about 0.1%∼0.2%) can be seen using updated TFIDF. Specifically, the improvement of Softmax Regression was the most remarkable (approximately 0.24%).

During this experiment, normalization of data seems not helpful to increase the accuracy. The accuracy of KNN classifiers decreased while that of Softmax Regression increased negligibly when considering the time of implementation.

7 Discuss

[Lu et al., 2020] demonstrated that gender-related bias may mingle freely during feature engineering, specifically word embedding, and training machine learning models. Also, [Thelwall, 2018] presented that the sentiment classifier can perform better when it would be training on separate male and female data.

In this section, how gender bias affects classification tasks and the performances of diverse classifiers were being investigated.

The train and test data in this experiment were respectively divided into three types (i.e., male, female, and without gender information) concerning the number of keywords that are conveying gender information. Additionally, as this paper proved above, the feature selection method chosen in this section is updated TFIDF.

The detailed rules to identify the gender with respect to each tweet can be explained in the following phases.

1. Let parameter C_m as the number of keywords which occur in 'male word bag' (e.g., boy, king, etc.) and let parameter C_f as the number of keywords that appear in 'female word bag' (e.g., girl, queen, etc.)
2. If $C_m > C_f$, regard this tweet as a male tweet. In contract, it would be regarded as female tweet. Also, if there was a tie, this tweet would be regarded as female tweets due to majority vote.

The gender word bag used in this experiment was referenced from Gender Pairs in [Lu et al., 2020]. According to the rule presented above, the train and test dataset would be split into the following three types. From Table.5 and Table.6, it is obvious to see that there were the majority of tweets without gender information, also the number of male tweets and female tweets was relatively balanced. Therefore, the report only focused on male and female tweets to discuss the influences of gender bias on different classifiers.

Gender	Number of tweets
Male	817
Female	680
Without gender information	18409

Table 5: Gender Description in Test Dataset

Gender	Number of tweets
Male	6490
Female	5637
Without gender information	147126

Table 6: Gender Description in Train Dataset

To evaluate the impacts of gender bias on diverse classifiers, there was a specific mathematical formula defined in this report to compute the sentiment tendency under a certain gender. The formula was represented by the following:

$$T_i^g = \sum_{j \in S, j \neq i} \frac{C^g(i|j)}{C^g(j)}.$$

In this formula, g refers to the type of gender (i.e., g∈{female, male}) and S represents the set of sentiment (i.e., S= {negative, neutral, positive}), meanwhile i and j are all one of the types of sentiment. Parameters $C^g(i|j)$ and $C^g(j)$ are obtained through the confusion matrix of the classifier under a certain gender type. $C^g(j)$ is the count of tweets labeled j among the group of tweets whose gender type is g. In terms of $C^g(i|j)$, it is the number of tweets labeled j but would be predicted to label i among the group of tweets whose gender type is g.

Theoretically, a higher T_i^g means tweets that convey more information related to g-type gender were more likely predicted as i-type sentiment.

Generally, from Fig.1 the performances of Naive Bayes and Softmax Regression presented

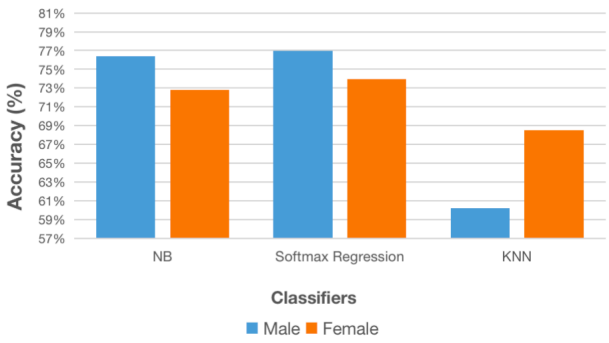

Figure 1: Comparison on basis of Gender among different classifiers.

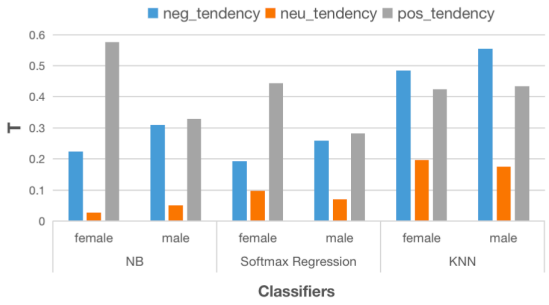

Figure 2: Sentiment Tendency due to gender under different classifiers.

diversely on male tweets and female tweets, whilst there was a slight differences using KNN classifiers. After calculating the sentiment tendency of each classifier under different gender groups, more detailed results can be found in Fig.2. While using Naive Bayes and Softmax Regression as a sentiment classifier, tweets which were related to female highly tended to be predicted as positive-type tweets.

These results imply that Naive Bayes and Softmax Regression Classifiers are more likely to amplify the bias of gender from the data it was trained on. According to the contingency table of sentiment by gender in train dataset Table.7, as for female kind of tweets, they were more likely to be positive due to a larger count in train data, which means there was a high correlation between female and positive. Similarity, there was a weak correlation between male and negative.

8 Conclusions

This paper investigated three kinds of machine learning models to build sentiment classifiers: Naive Bayes, Softmax Regression, and K Nearest Neighbor. Also, this report illustrated that although the performance of KNN is relatively poor and only better than Zero-R baseline, both other classifiers outperformed the baselines. Additionally, compared with the traditional TFIDF feature selection, the results in this paper shows that a new TFIDF feature selection method proposed by [Yantao Zhou and Wang, 2007] can improve the accuracy of all of these classifiers. This paper also discussed the influences of gender bias on various classifiers and found that Naive Bayes was the most sensitive with the gender bias on training dataset.

Sentiment	Male	Female
Negative	**2728**	2057
Neutral	1151	1073
Positive	2611	**2507**

Table 7: The Contingency table of sentiment by gender in Train Dataset. The numbers of the majority class are in **bold**.

References

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., and Passonneau, R. (2011). Sentiment analysis of Twitter data. In *Proceedings of the Workshop on Language in Social Media (LSM 2011)*, pages 30–38, Portland, Oregon. Association for Computational Linguistics.

Davidov, D., Tsur, O., and Rappoport, A. (2010). Enhanced sentiment learning using Twitter hashtags and smileys. In *Coling 2010: Posters*, pages 241–249, Beijing, China. Coling 2010 Organizing Committee.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N project report*, 1(12).

Lu, K., Mardziel, P., Wu, F., Amancharla, P., and Datta, A. (2020). Gender bias in neural natural language processing. In *Logic, Language, and Security*.

Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Thelwall, M. (2018). Gender bias in machine learning for sentiment analysis. pages 343–354.

Vadicamo, L., Carrara, F., Cimino, A., Cresci, S., Dell'Orletta, F., Falchi, F., and Tesconi, M. (2017). Crossmedia learning for image

sentiment analysis in the wild. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 308–317.

Yantao Zhou, J. T. and Wang, J. (2007). Improved tfidf feature selection algorithm based on information entropy. *Computer Engineering and Application*, 43(35):156–158.