Sentiment Classification in Tweets

Chengjun Shu

1 Introduction

Sentiment analysis is a popular topic in different fields, and the goal is straightforward. Given some text, classify them to be positive, negative, or neutral. Twitter is a popular micro blogging platform with 500 million tweets per day (Sayce, 2020). Furthermore, more and more experiments are being conducted using Twitter data because it can easily access large and valuable data.

This paper focuses on answering one question, "does unlabeled data improve Twitter sentiment classification?". We will examine this question by utilizing different machine models with varying amounts of supervision. Naive Bayes, Logistic regression and Self-Training are the three machine learning models used in this paper. The main task is to investigate whether the semi-supervised model is able to outperform the standalone supervised model by making use of the unlabeled data. Lastly, we will be using the results obtained to answer the proposed question.

2 Literature Review

Blodgett et al. (2016) conducts a case study on Twitter investigating specific dialectal language in the context of an online conversational text, and they provided the data set used in this paper.

Pang et al. (2002) used different ML (machine learning) techniques namely Naive Bayes (NB), ME (Maximum Entropy), and SVM (Support Vector Machine) to classify the sentiment of movie reviews into binary classes. They conducted that SVMs tend to perform the best, and Naive Bayes tend to perform the worst, although the gaps are not very large. Further to this, they experimented with the accuracies of NB and SVM using Feature presence, and investigate whether reliance on Feature frequency could improve the accuracies for NB and SVM. Besides, they explore the usage of Unigrams, Bigrams, Parts of speech (POS) and combined features as features, and they conducted that Unigram presence information outperforms others.

Da Silva et al. (2016) proposed a new semi-supervised framework that has proven to improve the accuracy of the sentiment analysis of tweets. Their proposed framework is integrated from Self-Training and combines an algorithm namely C^3E, which uses classification and clustering to achieve a better result. Further to this, they do a comparative analysis between two existing semi-supervised frameworks namely Self-Training and Co-Training. From their experiment results, all semi-supervised approaches offer better results than the standalone SVM approach, and their proposed approach outperforms others.

3 Data Set Description

Three data sets are being used. "dev.pkl" contains 4,000 instances and will be used as the test set. "train.pkl" contains 40,000 instances and will be used as the training set. "unlabeled.pkl" contains 100,000 unlabelled instances. Each instance is labelled with a sentiment label of either positive or negative, class distribution over different data set is shown below:

Name	Class Distribution
"dev.pkl"	50% positive, 50% negative
"train.pkl"	50% positive, 50% negative
"unlabeled.pkl"	NA.

Table 1: Class distribution on different data set

4 Baseline

The baseline chosen to use in this paper is Zero-R (Zero rule). It is an intuitive classifier as the prediction is made based on the majority class on the training set. Notice that in table 1, we

have uniform distributed data sets, either "positive" or "negative" will be chosen as a majority class for this classifier. From figure 1, the classifier tends to predict all instances to be "negative". If we force the classifier to predict all instances to be "positive", the result does not change. The accuracy for this baseline is 0.5, which is already a decent score.

```
Predicted labels :  ['negative' 'negative' 'negative'
Accurarcy for Zero-R:  0.5
Accurarcy for Zero-R (positive):  0.5
```

Figure 1: The accuracy of the Zero-R

5 Method

5.1 Feature set Selection

Teaching teams have provided two different feature presentations which we can use for the experiment stage. All of them are mapped from the raw data set by using a certain algorithm or model. We will be using the Embedding one which represents each tweet as a 384-dimensional feature vector. Using this feature set is because the feature spaces are much smaller compared with the TFIDF features which represent each tweet as a 1000 dimensional feature vector.

5.2 Classifier

5.2.1 Naive Bayes (NB)

Twitter Sentiment analysis is a binary classification task, and NB is capable of a binary classification problem. NB is a supervised learning algorithm that makes predictions by using prior probability. It makes a simple assumption by assuming that features are conditionally independent of the class labels. We will be using Gaussian NB (GNB) variants since it is based on a continuous distribution, and is suitable for the Embedding data set.

5.2.2 Logistic regression (LR)

Logistics regression is a supervised learning algorithm and it is also a probabilistic model similar to NB. However, it does not make the conditional feature independence assumption, and it uses the Sigmoid function to map the prediction result to one of the classes. We use LR to analyse and compare the impacts of using a semi-supervised approach on different probabilistic models.

5.2.3 Self-Training

Self-Training is a semi-supervised learning algorithm and involved different approaches. We will be using the Pseudo-label in this paper. This was proven to be an efficient semi-supervised method for image classification over the deep neural network (Lee et al., 2013). It has to pre-train the unlabelled data by using a supervised model first, predictions satisfying the confidence threshold will be given pseudo-labels. Finally, retrained the model with the labelled data and pseudo-labels data.

5.3 Evaluation Metrics

We will use the Accuracy and AUC (Area Under The Curve) score to evaluate the performance of classifiers. Accuracy can easily interpret the performance of our classifier and is appropriate in our case since we have a balanced data set.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

AUC measures the classification separability for a given classifier. It ranges from 0 to 1. A higher AUC score for a model illustrates that it can more accurately predict class 0 as 0 and class 1 as 1 (Narkhede, 2018).

5.4 Parameters Settings

We will first standardise our Embedding data set to have a mean value of 0 and a standard deviation of 1. This will minimise the impact of data on our ML models.

For GNB we will use the default parameter. For the LR, we will set the max iteration to 1000 to ensure the classifier will converge, and set the weight for class to be None to ensure the classifier will not bias towards a certain class. Moreover, set the random state to be 42, so we can produce the same result.

Besides, for the Self-Training, to avoid overfitting and long-time computation, we only set the max iteration to None, which informs the classifier to terminate when there are no more labels to add or all unlabelled data has been labelled.

5.5 Experimental Settings

In the experiment, we want to see whether using the unlabelled data set could boost the performance of our supervised model by using the Self-Training algorithm. Firstly, we will train our supervised models which use to compare the performance changes after using a semi-supervised approach. Then, apply the semi-

supervised approach to all our supervised models, and use both the training data set and the unlabelled data set to train the new models. All the unlabelled data will be given pseudo-labels. Lastly, analyse the result obtained.

Moreover, we also like to investigate how will the semi-supervised performance vary if we only use unlabelled data to train. Hence, we will conduct one additional experiment and compare it with the one using both labelled and unlabelled data. To ensure this experiment follows the same data distribution as the original, we will label 30% of unlabelled data by using our supervised models, remains data will be given pseudo-labels.

Experiment	Data Size
Original	40,000 (labelled) + 100,000 (unlabelled)
New	30,000 (labelled) + 70,000 (unlabelled)

Table 2: Illustration of the experiment

6 Result

Three sentiment classifiers, NB, LR and Self-Training examined the Embedding feature representation, and the results are shown below:

Classifier	Accuracy
NB	0.61875
Semi-NB	0.58775
LR	0.69825
Semi-LR	**0.70050**

Table 3: The Accuracy score of classifiers. Best score is shown in **bold face**.

Classifier	AUC
NB	0.68329
Semi-NB	0.64026
LR	0.76241
Semi-LR	**0.76266**

Table 4: The AUC score of classifiers. Best score is shown in **bold face**.

From table 2, we can see all of the classifiers outperform the baseline. Applying the Self-Training algorithm to LR, it comes with the highest accuracy score (0.7005) among others, followed by LR (0.69825), and NB (0.61875). Besides, the Self-Training algorithm does not seems to boost the performance too much. We only increase 0.2% (from 0.69825 to 0.70050) compared with the standalone LR. Moreover, it even decreases the classifier performance, where there is about a 3.1% significantly decrease (from 0.61875 to 0.58775) in NB after using this.

Classifier	Accuracy	AUC
Semi-NB	0.58350	0.62735
Semi-LR	**0.69525**	**0.76154**

Table 5: The Accuracy and AUC score of using the unlabelled data only. Best score is shown in **bold face**.

In terms of the AUC score, from table 3, all of the classifier scores are over 0.5 which denotes that they are not making a random guess. LR with Self-Training receive the highest score (0.76265) again which dominated the others.

Furthermore, comparing the result from table 3,4 and 5. The performance of using a semi-supervised model trained with only unlabelled does not outperform the combined data set.

7 Discussion

From table 3 and 4, we notice there is a performance gap between LR and NB in training with the same labelled data. The reason for that is LR tends to work better on a large data set and feature spaces than NB (Ng and Jordan, 2001). Since we have 40,000 training instances and each has 384-dimensional feature vectors. Besides, the conditional independence assumption in NB is been violated. Feature vectors of the data set represent the "meaning" of a given instance, and there is some correlation between them.

Moreover, from table 3 and 4, we also notice that LR slightly increase its performance while NB decreases after using the semi-supervised algorithm. The structure of the Self-Training model and the properties of the supervised model itself could be the main reason. Self-Training model retrains itself with its own generated labels, the labels it generated are what it thinks is the correct one. This means the model itself can continuously generate wrong labels over and over until the termination conditions have been satisfied. Besides, it needs pre-trained with a supervised model first. This illustrates that the performance of a semi-supervised

model will vary on the accuracy of the labels generated by the supervised model.

Recall that AUC measures the separability of classification, and some researchers elaborate more on the AUC score. They state that AUC range from 0.75 to 0.85 means moderate classification accuracy and those less than 0.75 means low accuracy (Bowers and Zhou, 2019).Based on the result from table 4, we will find the LR (0.76241) is in the moderate range, while the NB (0.61875) is in the low. This illustrates that using NB in this semi-supervised model is more likely to generate wrong labels. Besides, NB is a generative model, fitting with wrong labels will lead the joint probability model to be incorrect, thus effect using the prior probability to make a prediction. This will cause a decrease in performance. By contrast, LR is more likely to generate correct labels. LR is a discriminative model and does not model the joint probability but instead directly uses the training data to predict. Therefore, even training with a data set with wrong labels, LR can still learn more patterns, hence gaining better performance on prediction.

In addition, this is also further demonstrated in table 5 where we only use unlabeled data to train the semi-supervised model. The labelled data are generated by using supervised models, and we have no clue what the ground truth labels are. However, It turns out that LR still reaches a decent accuracy score (0.69525), while NB receives a low score (0.58350) again.

8 Conclusions

This paper creates three different machine learning models namely Naive Bayes (NB), Logistic Regression (LR) and Self-Training. We conduct comparative experiments on them by trying to examine whether applying the semi-supervised method could increase the performance of our supervised model. We found that LR with Self-Training achieves the highest accuracy score (0.70050) and our NB with Self-Training results in a worse accuracy score (0. 58775) than before.

The answer to the proposed question is yes, using the unlabelled data can improve the Twitter sentiment classification. However, this comes with a condition that we have to use a "good" supervised model to pre-trained the unlabelled data first. This is proven by our results, the choice of a supervised model in the semi-supervised approach will make a lot of difference.

References

Blodgett, S. L., Green, L., and O'Connor, B. (2016). Demographic dialectal variation in social media: A case study of African-American English. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1119–1130, Austin, Texas. Association for Computational Linguistics.

Bowers, A. J. and Zhou, X. (2019). Receiver operating characteristic (roc) area under the curve (auc): A diagnostic measure for evaluating the accuracy of predictors of education outcomes. *Journal of Education for Students Placed at Risk (JESPAR)*, 24(1):20–46.

Da Silva, N. F. F., Coletta, L. F., Hruschka, E. R., and Hruschka Jr, E. R. (2016). Using unsupervised information to improve semi-supervised tweet sentiment classification. *Information Sciences*, 355:348–365.

Lee, D.-H. et al. (2013). Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896.

Narkhede, S. (2018, June 27). Understanding AUC-ROC curve. *Medium*. `https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5`.

Ng, A. and Jordan, M. (2001). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In Dietterich, T., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems*, volume 14. MIT Press.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics.

Sayce, D. (2020, December 16). The number of tweets per day in 2020. *David Sayce*. `https://www.dsayce.com/social-media/tweets-day/`.