COMP90049 Project 2: tweets r mad, or r they!?

Anonymous

1. Introduction

Sentiment analysis on tweets deals with determining the writers' attitude about the topic of the tweet. There is a lot of interest on this topic since it allows you to get an overall understanding of how the general population feels about a particular topic on social media (Twitter).

This paper focuses on finding out whether using tweet text as words is an effective method to identify people sentiment on Twitter. For this purpose, the project uses a list of 46 features to predict the sentiment using Naïve Bayes (NB), C4.5 (Decision Tree) and Support Vector Machine (SVM) as classifiers. The hypothesis that will be explored in this paper is that using tweet text (words in a tweet) is not an effective method to predict the author's sentiment regarding the topic.

2. Related Studies

Sidorov et al. (2012) describes methods of pre-processing and explores how different settings such as n-gram size, number of sentiment classes etc affect the effectiveness of machine learning (ML) algorithms.

Using basic ML techniques, Pang et al. (2002) researched sentiment analysis on movie reviews and discussed the factors that make the sentiment analysis problem more difficult.

Rosenthal et al. (2017) describes the fifth edition of the SemEval challenge and they also created the dataset that is used in this project.

3. Datasets

This project contains two datasets: 'train.arff' and 'eval.txt'.

Datasets	Description
train.arff	This dataset contains 22987 instances. This data is used as the training set to train the model that will be predicting classes.
eval.arff	This dataset contains 4927 instances. This data is used as the test set for evaluation when the is model developed. The predicted class for each instance in the test set is compared with the actual class and the effectiveness of the model is then evaluated.

Table 1: The description of the datasets.

3.1. Properties of Datasets

The instances in the datasets have 46 features that have been chosen from words in tweets. Each tweet represents either a negative, neutral or positive sentiment. Since the method used to obtain these features has not been specified it is difficult to predict whether these are the best available features for sentiment analysis. This limits the possibility for better accuracies for predictions

4. Baseline

Before starting on complex methodologies, it is important to set up a baseline for the project. The Zero-R classifier was used for this purpose. Since the most prevalent class is *"neutral"*, Zero-R chose *"neutral"*. For a balanced training set, the accuracy should be close to 33.3% (Rosenthal et al., 2017) however, Zero-R produced an accuracy of 48.7%, which is an impressive score for a baseline.

5. Methodologies

Initially the training set was used for a couple strategies of feature selection. After selecting the most useful features from the initial 46, a model was built for each: NB, Decision Tree (DT) and SVM classifiers.

5.1. Feature Selection

A couple methods of feature selection were tried out in order to find the most useful features for building the models. Initially the *"id"* feature was removed as it has no effect on the class. Firstly (**FS1**), the accuracy for each classifier when using all the features (46) was obtained.

The second method (**FS2**) of feature selection is called "CfsSubsetEval" on Weka. This method finds a subset of features that are highly correlated with the class and have a low correlation with other features. This produced 17 features, which are representative of negative, neutral and positive sentiments. Take for example (Figure 1 & 2), *"happy"* and *"stupid"*, these words clearly represent positive and negative sentiments respectively.

The third method (**FS3**) is classifier specific, where the subset of features which has highest 'merit' for each classifier is found. On Weka, this method is called "ClassifierSubsetEval". This is found by using a part of the training set as a test set for different subsets of features (holdout). This method selected 12 features for NB, 33 features for DT Pruned (38 for Unpruned) and 22 features for SVM.

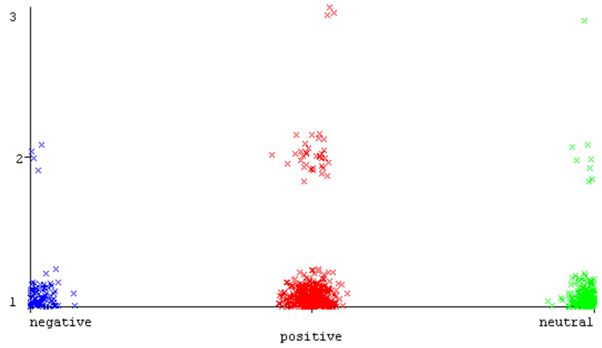


Figure 1: Distribution for "happy"

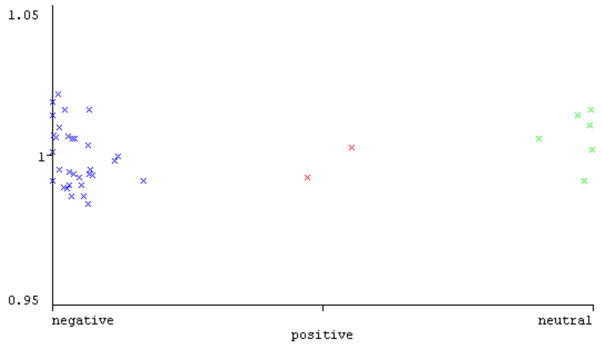


Figure 2: Distribution for "stupid"

5.2. Classifiers

The NB classifier is the most basic statistical ML technique which assumes that attributes are conditionally independent (Kotagiri, 2019).

The DT classifier is a rule-based approach to ML which creates a binary tree as the model. This paper discusses both pruned and unpruned DTs (Kotagiri, 2019).

The SVM classifier is a statistical ML technique which tries to find the hyper-plane with the maximum distance from the two classes (Kotagiri, 2019). SVM can only be used with two classes and since this project has 3 classes, multiple SVMs had to be modeled using the One VS One strategy.

6. Evaluation

6.1. Metrics

6.1.1. Accuracy

Accuracy is the fraction of predictions that are correct.

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

6.1.2. Recall

Recall is the fraction of total relevant classes that are correctly classified.

$$Recall = \frac{TP}{TP + FN}$$

Average Recall is the recall averaged across all classes which is necessary to determine effectiveness when using an imbalanced test set (Rosenthal et al., 2017).

$$Average\ Recall = \frac{1}{3}(R^P + R^N + R^U)$$

6.2. Results

	Count	**Percentage (%)**
Neutral	11454	49.8
Positive	6471	28.2
Negative	5062	22.0

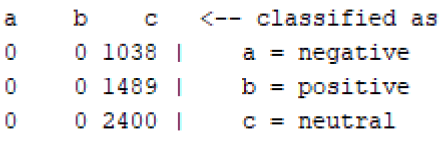
Table 2: Class distribution for training set.

	NB	**DT (Pruned)**	**DT (Unpruned)**	**SVM**
FS1	46.1%	48.4%	47.7%	49.0%
FS2	47.1%	48.6%	48.4%	48.9%
FS3	48.0%	48.8%	47.8%	48.9%

Table 3: Accuracy for Classifiers for different Feature subsets/strategies.

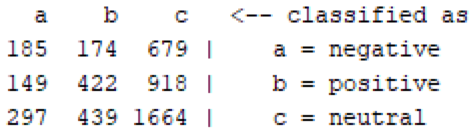
	+ive	**Neutral**	**-ive**	**AvgRec**
Zero-R	0	1	0	0.333
NB	0.189	0.851	0.077	0.372

Table 4: Recall for each class and Avg. Recall, for Zero-R and NB.



```
   a    b    c   <-- classified as
   0    0 1038 |    a = negative
   0    0 1489 |    b = positive
   0    0 2400 |    c = neutral
```

Figure 3: Confusion Matrix for Zero-R



```
   a    b    c   <-- classified as
 185  174  679 |    a = negative
 149  422  918 |    b = positive
 297  439 1664 |    c = neutral
```

Figure 4: Confusion Matrix for NB

6.3. Analysis

6.3.1. Baseline and Bias

As shown on Table 2, the number of instances that belong to the "neutral" class is almost 50% of the entire training set. This introduces a massive class imbalance, and this has a huge effect on Accuracy since accuracy is highly sensitive to class imbalance (Rosenthal et al., 2017).

This effect can be analysed using the average recall since it is not affected by class imbalance (Rosenthal et al., 2017). Even though the accuracy for NB is much lower than that of Zero-R, it has a significantly higher average recall than Zero-R (Table 4) which means NB is closer than Zero-R to be the perfect classifier.

The confusion matrix in Figure 4 clearly shows how low the precision of the predictions is. The confusion matrices for other classifiers also follow a similar pattern. Figure 4 shows how biased the test set is, which is not ideal for evaluation purposes.

6.3.2. Classifier Accuracy (Table 3)

The results produced from the ML algorithms were very similar, with SVM producing the highest accuracy closely followed by DT, and NB producing the lowest accuracies.

A key interesting finding was that using pruned decision trees helps to prevent the model from overfitting the training set. This makes sense since pruning removes outliers reducing chances of overfitting.

6.3.3. Feature Selection (Table 3)

When the original feature set was used, the accuracies were low since there were a lot of features which weren't necessary helpful for classifying the tweets to the writer's sentiment.

However, the second feature selection strategy significantly improved the results.

To go a step further, the original feature set was used individually for each classifier to select the features most helpful for the process of each classifier. This produced even better results for NB and DT. However, there is a slight decrease for SVM which stems from the fact that a smaller feature set reduces the effectiveness of SVM.

6.4. Discussion

Despite trying several classifiers in combination with varied feature selection strategies, it was not possible to obtain a significant improvement in accuracy, with only SVM (all feature subsets) and DT (for one feature subset) slightly surpassing the effectiveness achieved by the baseline classifier.

This is mainly due to the fact that a list of words, when considered out of context, could include words indicating the opposite sentiment as described in Pang et al. (2002). In order to conduct sentiment analysis on tweets, it is important to find the context of the words used. Take for example the phrase "not bad", this has a sentiment of positive/neutral, however typically a system similar to that described above would classify this as negative. Turney (2002) sums this up aptly, "the whole is not necessarily the sum of the parts".

7. Conclusions

The results could be much better if different/more features are selected from the original tweets and if the "neutral" class is removed reducing the possible number of classes to positive and negative (Sidorov et al., 2012). Since it is given that deep learning is guaranteed to produce much better results (Rosenthal et al., 2017), it could be considered as a more prospective future improvement.

With all the discoveries in this paper, it can be concluded that the initial hypothesis was correct, that tweet text is not helpful in identifying people sentiment on Twitter. In order to apply sentiment analysis on tweets a more comprehensive feature set has to be created.

8. References

Rosenthal, Sara, Noura Farra, and Preslav Nakov (2017). SemEval-2017 Task 4: Sentiment Analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval '17)*. Vancouver, Canada.

Grigori Sidorov, Sabino Miranda-Jiménez, Francisco Viveros-Jiménez, Alexander Gelbukh, Noé Castro-Sánchez, Francisco Velásquez, Ismael Díaz-Rangel, Sergio Suárez-Guerra, Alejandro Treviño, and Juan Gordon. Empirical Study of Machine Learning Based Approach for Opinion Mining in Tweets. *LNAI 7629*, 2012, pp. 1–14.

Kotagiri, R. (2019). *Part B: Machine Learning*. Lecture, The University of Melbourne.

Pang, B., Lee, L., and Vaithyanathan S.: Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL*, pp. 79–86. Association for Computational Linguistics (2002)

Peter Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proc. of the ACL*.