COMP90049 Project 2 Report: Which emoji is missing?

Anonymous

1. Introduction

Machine learning is the ability of computers to learn and predict outcomes without explicitly programming. There are two types of Machine learning methods – Supervised and Unsupervised.

The aim of this report is to critically analyze the effectiveness of some Supervised Machine learning methods on the problem of determining which emoji was used in a tweet and express the knowledge gained.

In this report, we have based our analysis on the Naïve Bayes method and Decision tree method.

2. Dataset

The data used in this project has been collected from the Twitter API. The dataset used in this project are of three types: training, development and testing. For each type of dataset, we have been given a .txt file, .csv file and an .arff file.

The .txt files contain the raw text of the tweets in the format as follows: id TAB class TAB tweet, where the 'id' corresponds to the line number, 'class' corresponds to the emoji class and the 'tweet' is the text of the tweet.

The .csv files are comma-separated-value files, where we have recorded the frequencies of tokens. This file has the following format: ID, List-of-token-frequencies, Class.

The .arff files contain the vector representations of all the tweets and are suitable for use with WEKA.

The training dataset will be used to train and pre-process the model. The development model will be used to identify if the model was able to predict the emoji correctly and calculate accuracy. The test dataset is used to output the predictions made by the method as the class values were not available.

3. Weka

In our project, for implementing the supervised machine learning algorithms on our dataset, we have made use of an open source software, 'Weka'. Weka is a collection of machine learning algorithms for the purpose of data mining which can be directly applied on a dataset. It includes various tools for data pre-processing, clustering, classification and visualization [1].

4. Evaluation Metrics

In this report, we have made use of some evaluation metrics to analyze the effectiveness of the supervised machine learning methods, which are as follows:

a) Accuracy: It is provided by WEKA in the form of percentage of correctly classified instances. It also provides the percentage of incorrectly classified instances.

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

b) Precision: WEKA provides the precision for each emoji class. It refers to the measure of the proportion of positive predictions that are correct. This proportion is based on the total number of positively predicted values. It is calculated as:

$$\text{Precision} = \frac{TP}{TP+FP}$$

c) Recall: WEKA provides the recall for each emoji class. It is the proportion of the positive predictions to the sum of the instances that have a positive actual result irrespective of the predicted value. It is calculated as:

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

5. Naïve Bayes

Naïve Bayes classifier is a supervised machine learning method based on the Bayesian theorem and mainly works on the concept of 'probability' in order to classify or predict new entities and assumes that there are no dependencies amongst attributes [2].

6. Decision tree

Decision trees have a flow-chart-like structure, where attributes are tested on each internal nodes and outcomes of the tests are represented on the branches and the classes are represented on the leaf nodes [3].

7. Results

Using the Naïve Bayes and Decision tree classifier on the development data in WEKA

resulted in the following data, which has been represented in the form of tables.

Algorithm	Classes	Precision	Recall
Naïve Bayes	Clap	0.245	0.534
	Cry	0.361	0.294
	Disappoint	0.191	0.141
	Explode	0.459	0.261
	FacePalm	0.232	0.148
	Hands	0.629	0.378
	Neutral	0.25	0.235
	Shrug	0.208	0.184
	Think	0.298	0.385
	Upside	0.279	0.279
Weighted Average		**0.33**	**0.304**
Decision tree	Clap	0.54	0.53
	Cry	0.497	0.459
	Disappoint	0.291	0.221
	Explode	0.439	0.549
	FacePalm	0.395	0.286
	Hands	0.803	0.673
	Neutral	0.316	0.429
	Shrug	0.389	0.333
	Think	0.485	0.451
	Upside	0.359	0.356
Weighted Average		**0.464**	**0.453**

Table 1: Precision and Recall

The accuracy as given by Weka for Naïve Bayes is as follows:

Correctly classified instances		Incorrectly classified instances	
Count	**%**	**Count**	**%**
3695	30.3765	8469	69.6235

Table 2: Naïve Bayes - Accuracy

The accuracy as given by Weka for Decision tree is as follows:

Correctly classified instances		Incorrectly classified instances	
Count	**%**	**Count**	**%**
5516	45.3469	6648	54.6531

Table 3: Decision tree – Accuracy

8. Analysis

It can be observed from Table 2 and Table 3, that Naïve Bayes has an accuracy of 30.37% Whereas, Decision Trees has an accuracy of 45.34%. This is due to, the manner in which both the algorithms predicted emojis. Decision tree is a rule-based classifier, wherein the classifier makes prediction on the basis of a set of rules that are defined according to the training data. Whereas, in Naïve Bayes, the method predicts results on the basis of Prior Probability and offers a low accuracy when independence assumption doesn't hold i.e. the attributes in the training data are independent of each other [2].

After implementing both the methods on the given dataset we get the following Precision and Recall for each emoji class, which have been represented in Chart 1 and Chart 2 respectively.

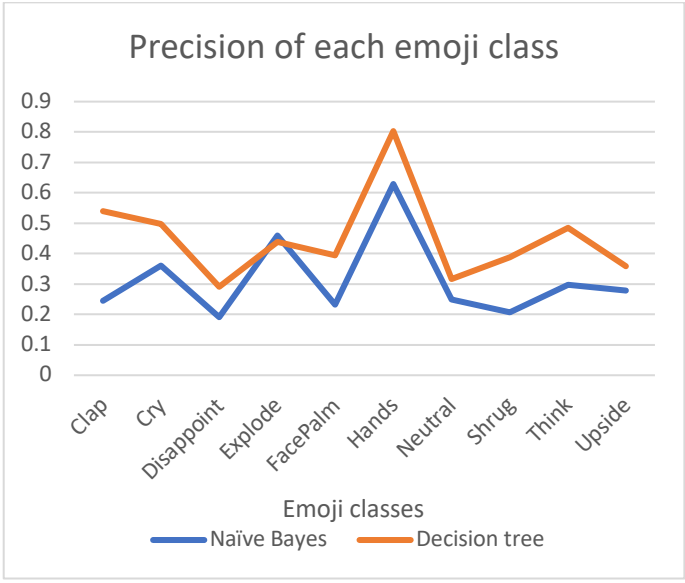

Chart 1: Precision of Naïve Bayes v/s Decision tree

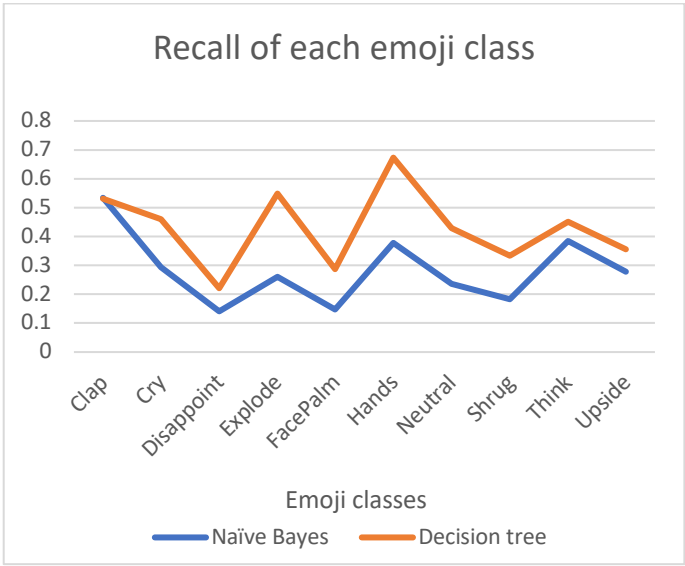

Chart 2: Recall of Naïve Bayes v/s Decision tree

From Chart 1, it can be seen that for 9 out of 10 emoji classes Decision tree gets greater Precision than Naïve Bayes, which in turn leads to better accuracy for Decision tree method.

Also, it can be seen from Table 1, that Decision tree had a Precision (Weighted average) of 0.464, which means that when Decision tree method predicted an emoji for a tweet, it was correct 46% of the time. Whereas, for Naïve Bayes, it was correct only 33% of the time.

This can be understood better by taking the case of one of the emoji classes, such as 'Hands', where Decision tree had a Precision of 0.803 and Naïve Bayes had a Precision of 0.629. Decision tree had a higher precision in this case as they were able to find 890 True Positives and 218 False Positives. Whereas, Naïve Bayes was able to find only 500 True Positives and 295 False Positives, which resulted in a lower Precision.

Chart 2 reflects that Decision tree also gets greater Recall in most emoji classes, which also leads to the method getting better accuracy.

Also, it can be seen from Table 1, that Decision tree had Recall (Weighted average) of 0.453, which means that it correctly identified emojis in 45% of total tweets. Whereas, for Naïve Bayes, it correctly identified emojis in 30% of total tweets.

This can be better understood by taking the case of one of the emoji classes, such as 'Disappoint', where Decision tree had a Recall of 0.221 and Naïve Bayes had a Recall of 0.141. Naïve Bayes method had a lesser recall because it was able to find 396 False Negatives and only 65 True Positives. Whereas, Decision tree was able to find 102 True Positives and 359 False Negatives, which resulted in a greater Recall for the class.

9. Conclusion

After careful analysis of both the methods, Naïve Bayes and Decision Trees with respect to Accuracy, Precision and Recall. Decision Tree was able to correctly predict emojis in more number of tweets and had 14.97% more Accuracy, 13.4% more Precision and 14.9% more Recall when compared to Naïve Bayes method. Thus, making Decision Tree a better method out of the two for the given dataset.

However, this does not mean that Decision Tree is a better method for data mining than Naïve Bayes in general. The results of Naïve Bayes may vary greatly with changes in the type of dataset that is used for training the model, such as changes in the number of attributes or the level of dependency between the attributes. These changes can result in an increase in the performance of Naïve Bayes method compared to any other method.

References

[1] https://www.cs.waikato.ac.nz/ml/weka/ viewed on 18-May-2018.

[2] Ashari, A., Paryudi, I., & Tjoa, A. M. (2013). Performance comparison between Naïve Bayes, decision tree and k-nearest neighbor in searching alternative design in an energy simulation tool. *International Journal of Advanced Computer Science and Applications (IJACSA)*, *4*(11).

[3] Gupta, B., Rawat, A., Jain, A., Arora, A., & Dhami, N. (2017). Analysis of Various Decision Tree Algorithms for Classification in Data Mining. *International Journal of Computer Applications*, *163*(8).