Project 2: Twitter Trolls and the Tweeters who Love them

1. Introduction

The aim of this project is to find out if tweets can be used to identify trolls and gain knowledge about the problem. This report will first introduce some relevant literature, then analyse system behaviour by results and examples, finally, conclude some knowledge about the project.

2. Machine Learning

Machine Learning is a field which allows computers to learn a task by experience and examples without being detailed programming. It can be used to label data by classification (Erfani & Verspoor, 2018). In some recent researches, machine learning techniques are used to analyse sentiment behind tweets (Arif, Li, Iqbal, & Liu, 2018), which is similar to this project's topic. In the following section, we will try to use machine learning techniques to identify trolls by tweets.

3. Data Set

The dataset is a smaller version of 3 million Russian troll tweets which provide by Roeder (2018). It includes a large number of tweets which are classified into three classes: Right Troll (nativist and right-leaning populist), Left Troll (social liberalism) and Other (Linvill & Patrick, 2018).

The dataset has small, medium and large size which corresponds to different number of attributes. Each size has 223K tweets, roughly 60% of total for training, 20% for development and 20% for test. The mostX files include the top X frequent terms. The bestX files include the top X frequent terms with the greatest Mutual Information and Chi-Square values for each of the three classes respectively. Therefore, it has more than X attributes.

4. Baseline

To test the feasibility of our system, we use Zero-R as a baseline.

Data Set	Correctly Classified Instances	Data Set	Correctly Classified Instances
Most10	35.785%	Best10	35.785%
Most50	35.785%	Best50	35.785%
Most 200	35.785%	Best 200	35.785%

Table 1: Zero-R – Original Data

All datasets use the same tweets, so they have same accuracies by applying Zero-R.

5. Method

Naïve Bayes Classifier can classify an instance by combining every attribute's probability. It is simple and fast. Moreover, it can easily show each attribute's probability and value (Erfani & Verspoor, 2018), which makes it suitable for the experiments.

6. Original Data Set Results

Firstly, we apply Naïve Bayes Classifier to the original data set, the results are presented in the following table:

Data Set	Correctly Classified Instances	Data Set	Correctly Classified Instances
Most10	53.3669%	Best10	55.9384%
Most50	58.4155%	Best50	60.1167%
Most 200	60.8908%	Best 200	61.1702%

Table 2: Naïve Bayes Classifier – Original Data

6.1. The Feasibility of Identifying Trolls

There are three different classes, Left Troll, Right Troll and Other. If we classify a tweet randomly, the accuracy should be 1/3 which is 33.3333%. According to the results above, all data sets could be classified with much higher accuracy rates, which is also higher than the Zero-R baseline. Therefore, in general, tweets texts can be used to identify

trolls on Twitter, and Naïve Bayer Classifier is a feasible method to this case.

6.2. The Influences of Attributes

The general trend is a greater number of attributes could lead to a higher accuracy (It requires that the larger dataset includes all attributes which the small dataset has). For convenience, we use small and medium data set to illustrate here.

	Most10		**Most50**	
Class	Precision	Recall	Precision	Recall
Left Troll	48.2%	25.5%	51.3%	46.0%
Right Troll	49.3%	31.0%	56.6%	28.4%
Other	56.1%	98.2%	62.5%	97.0%

Table 3: NB – Most10 and Most50

The most10 dataset has 10 terms: a, and, for, in, is, of, on, the, to, you. All these words are very common English words without special political leanings. This is the reason why Class "Other" has the highest recall: most instances are classified to "Other".

In contrast, except the ten terms above, most50 has more meaningful words such as people, police, trump. For example, the tweet "*Wow. That's hard to watch. OMG. We have sick people in the Police force too. Just God awful.*" belongs to "Left Troll". It includes both "people" and "police". When we use most10 to build the model, the prediction is "Right Troll" with 76.7% probability. By using most50, it could be correctly predicted to "Left Troll" with 82.6% probability.

	Term
Most10	a, and, for, in, is, of, on, the, to, you
Best10	a, amp, and, bbsp, beeth, black, blacklives, breaking, dallas, Hillary, i, is, mage, news, obama, of, on, pjnet, retweet, rt, rtamerica, tcot, the, this, to

Table 4: The Terms in Most10 and Best10

Likewise, according to the results in Table 2, bestX data sets have better performance than mostX. The reason is bestX includes more meaningful attributes as well as more quantity of attributes, which can present one's political leanings better. Therefore, the following experiments will use bestX data sets.

7. Adjust Attributes

According to analysing above, there are some attributes cannot express users' political leanings, which means they are meaningless to this project. To test which attributes may have negative influences on the system accuracy, we will try to delete each most10's attribute in best50 separately, then compare the results.

original	**tweet-id**	**user-id**
60.1167%	60.1274%	61.5475%
a	**and**	**for**
60.4584%	60.2805%	~~60.0616%~~
is	**of**	**on**
60.7681%	60.4139%	~~59.7982%~~
the	**to**	**you/in**
60.8232%	60.5527%	**-**

Table 5: Accuracy after removing each term

Overall, after removing these common words, the accuracy will slightly improve. In particular, tweet-id is generated by the system, which is not related to tweets' texts. In contrast, user-id should be an important attribute because each user id is corresponding to one label. In other words, all tweets which have the same user-id should have the same label. However, the result shows the different way. After analyzing datasets, we find out that the reason is the users in the training dataset won't show again in the development data set. Therefore, the user-id is meaningless in the model.

8. Using Pre-process Data Set

According to the analysis above, all tweets which are posted by the same user belong to the same class. When we classify a tweet, we are actually classifying the tweet's author.

Therefore, instead of only considering the tweet itself, we can pre-process the data set by combining all tweets posted by the same person. For example, if user A posts tweet 1, tweet 2 and tweet 3:

	apple	orange	banana
Tweet 1	1	1	0
Tweet 2	1	0	1
Tweet 3	1	0	0

Table 6: Pre-process example

After pre-processing, we see all tweets posted by user A as:

	apple	orange	banana
Tweet 1	3	1	1
Tweet 2	3	1	1
Tweet 3	3	1	1

Table 7: After Pre-process

The pre-process is implemented by C++. The C++ code will create new csv files for both train and dev data. Then we use Weka to process the new files.

For each tweet, there is more quantity of attributes. For the system, we can consider each user more comprehensively.

	Original	After Pre-process
Best10	55.9384%	80.2826%
Best50	60.1167%	80.9375%
Best200	61.1702%	80.7204%

Table 8: The Results of After Pre-process

After pre-processing, the performance of the system has a significant improvement. It also shows that after considering all tweets from the same user, the number of attributes has less influence on the results.

9. Verification by Other Methods

To check the feasibility of the system, we apply other Machine Learning methods to the same processed data set Best200. The results are showing below:

	Original	After Pre-process
J48	24.9261%	80.1029%
SMO	64.6243%	81.3503%

Table 9: The Results of Other Methods (Both J48 and SMO use the default parameters in Weka)

The results show that the knowledge about the problem in the report is reasonable and can be applied to other algorithms.

10. Conclusion

During the experiments, we gain some knowledge about the problem. Firstly, tweet text can be used to identify trolls on Twitter. Secondly, more quantity of attributes leads to higher accuracy. Moreover, the quality of the attributes also influences the results. Using terms with the greatest Mutual Information and Chi-Square values has better performance than using the most frequent terms. In addition, user-id could be used as an attribute in the model if the test data set has repeated user id with the training data. Otherwise, user id is useless as same as tweet id and other meaningless terms which are common in English. Finally, combining the same user's twitters and classifying them together is a powerful improvement of this project. These improvements are suitable for different Machine Learning algorithms.

References

Arif, M. H., Li, J., Iqbal, M., & Liu, K. (2018). Sentiment analysis and spam detection in short informal text using learning classifier systems. Soft Computing - A Fusion of Foundations, Methodologies and Applications, (21), 7281. https://doi-org.ezp.lib.unimelb.edu.au/10.1007/s00500-017-2729-x

Erfani, S., & Verspoor, K. (2018). Introduction to Data Mining and Machine Learning [Powerpoint slides]. Retrieved from https://app.lms.unimelb.edu.au/bbcswebdav/pid-6595057-dt-content-rid-31507430_2/courses/COMP90049_2018_SM2/lectures/11-ml_dm_intro.pdf

Erfani, S., & Verspoor, K. (2018). Lecture 14: Classification [Powerpoint slides]. Retrieved from https://app.lms.unimelb.edu.au/bbcswebdav/pid-6595057-dt-content-rid-31507436_2/courses/COMP90049_2018_SM2/lectures/14-classification.pdf

Roeder, O. (2018) Why we're sharing 3 million Russian troll tweets. In

FiveThirtyEight. 31 Jul 2018,https://fivethirtyeight.com/features/why-were-sharing-3-million-russian-troll-tweets/

Linvill, D. & Patrick, W. (2018) Troll factories: The Internet Research Agency and state-sponsored agenda building (working paper). Clemson University.