CS 839 Project Stage 1: Information Extraction from Natural Text: Extracting Cryptocurrencies from News Articles

Aishwarya Ganesan
ag@cs.wisc.edu

David Liang
david.liang@wisc.edu

Viswesh Periyasamy
vperiyasamy@wisc.edu

DataSet

1. **Entity type:** The entity type that we decided to extract is *Cryptocurrency*. A few examples of mentions of this entity type include *BitCoin*, *Ethereum*, *Ripple*, *BitCoin Cash*, etc.

2. **Number of Mentions:** The total number of mentions that we marked up is **9082**. The total number of negative instances in the data set after pruning is **243256**.
   Set I: The number of documents in set I is **200**, and the number of mentions in set I is **6080**.
   Set J: The number of documents in set J is **100**, and the number of mentions in set J is **3002**.

Classifier M

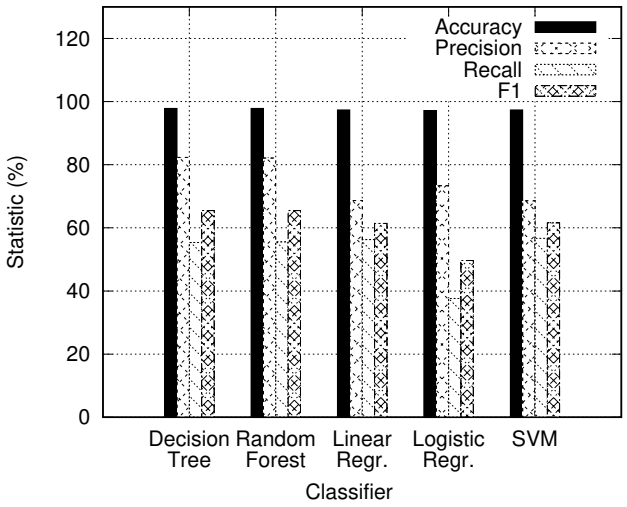

Figure 1. Initial Cross Validation Performance of different classifiers on set I

Figure 1 shows the results of performing cross validation with five folds on set I the first time using different classifiers. We tried the following features when we initially ran our classifiers:

Word length
Number of capital letters in the string
Does the string has all capital letters?
Does the string contains cash as substring?
Does the string contains coin as substring?
Is first letter of the word capitalized?
Is word surrounded by parentheses?

Based on the results (shown in Figure 1) we picked *Decision Trees* as the classifier M. The precision, recall, F1 of performing cross validation with M on set I with five folds are **82.33%**, **55.35**%, and **65.45**% respectively.

Classifier X

We added more features to improve the performance of the extractor. Some features that we added include:

The next string in the sentence that immediately follows the current word.
Counts of different characters in the word
Check for substrings *ium* and *eum*
Check for special characters like forward slash, dash, apostrophe, dot, etc.

Figure 2 shows the results of performing cross validation with five folds on set I using different classifiers after debugging (using set I) and the above features to improve the precision and recall. The type of the classifier that we finally settled on before the rule-based post-processing step is *Random Forest*.

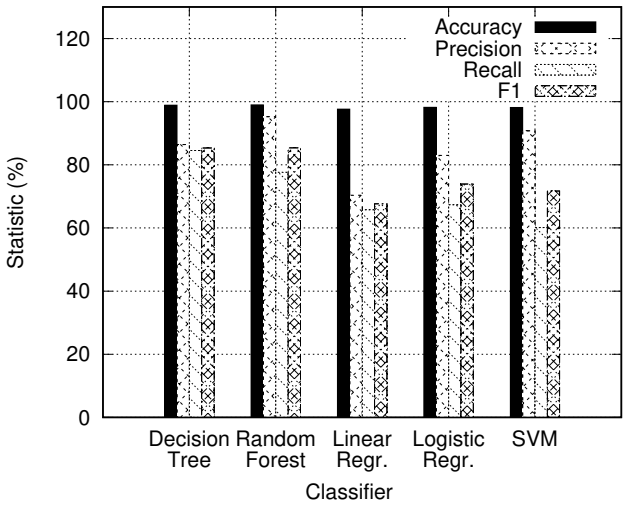

Figure 2. Cross Validation Performance after debugging different classifiers on set I

The results of running 5-fold cross validation using Random Forest on the test set J are as follows:
Accuracy: 99.31%
Precision: 96.86%
Recall: 82.21%
F1: 88.93%

Final Classifier Y

Rule-Based Post-Processing: We added a set of 25 common crypto currencies like *BitCoin*, *Ethereum*, etc., and 25 abbreviations of crypto currencies into a whitelist. If a word we are trying to classify is part of the whitelist, we mark it a cryptocurrency. The code snippet that uses the whitelist can be found in classify.py.

The final classifier Y is *Random Forest* along with the above post processing. The final results of running Y on the test set J are as follows:
Random Forest accuracy: 99.33%
Precision: 96.86%
Recall: 82.21%
F1: 88.93%