# Comp Sci 839 Project Stage 1

Chen Qian, Fangfei Lin, Jiahan Li

**Summary:**

We decided to extract all the people names from the text. There are 1665 names in all the documents. Our train set (set I) contains 200 files, including 1006 names, while our test set (set J) contains 100 files, including 659 names.

After performing cross validation on set I, we chose the Passive Regressive model: the precision is 93.88% ,the recall is 88.22% , and the F1 is 90.72%. Before the rule-based postprocessing step, we settled down with the Passive Regressive model: the precision is 92.32%, the recall is 62.33% and the F1 is 74.73%. Since we have reached 90-60 scale, we didn't apply any postprocessing rule-based steps.

**Process:**

For all the words in the both set I and set J, every word is a feature, whether such word is included or not. We converted each words into features, then converted into a sparse matrix. Also, we included the prefix and suffix of each word as a feature, and examine whether those prefix and suffix are likely to be follow or follow by a name. In addition, we included several features to train our models, including the number of words, whether the first letter is capitalized, whether it was followed by verb such as 'said', whether it follow some typical titles, whether we can match the names in dictionary.

We applied linear models such as linear regression, logistic regression and SGD, ensemble methods such as ExtraTrees , Adaboost, Gradientboost, Random Forest, as well as Naïve Bayes and Passive Regressive classifier. By applying cross validation and selecting F1 score as our criterion , the Passive Regressive classifier performs the best, thus we decide to use Passive Regressive algorithm to train our model.

In order to improve the performance of our selected model, we first used the grid search, however it didn't help. Moreover, we adjusted the class weights of the Passive Regressive model. We tried 1:1, 2:1 and 3:1. Since Passive Regressive Algorithm is a stochastics process, so for each class weights proportion, we ran 30 times on our datasets. When the class weight is 3:1, the model performs the best. Among 30 times, 4 times reached the precision rate 90%, the recall rate 60%. The best precision rate is 92% and the best recall rate is 62%, in average. The average of precision rate is 88%, and the recall rate is 64%. The best result has been attached. Fortunately, the precision and recall rate have been improved after adjusting the parameters.

Though we have applied multiple common classification algorithms on our datasets, the performances of our chosen models are not quite ideal. Referring to other papers on NLP, compared to the classification methodologies, the Conditional Random Fields may be a more appropriate approach to process the natural languages.

**Passive Regressive Model Result:**

| Precision Rate | Recall Rate |
|---|---|
| 0.8841 | 0.6627 |
| 0.8794 | 0.6598 |
| 0.8695 | 0.6715 |
| 0.8943 | 0.6423 |
| 0.8974 | 0.6386 |
| 0.9232 | 0.6233 |
| 0.8823 | 0.6620 |
| 0.8987 | 0.6153 |
| 0.8693 | 0.6650 |
| 0.8985 | 0.6336 |
| 0.8981 | 0.6496 |
| 0.8961 | 0.6357 |
| 0.9183 | 0.5905 |
| 0.8713 | 0.6671 |
| 0.6672 | 0.8755 |
| 0.8754 | 0.6723 |
| 0.9183 | 0.5905 |
| 0.8713 | 0.6672 |
| 0.8755 | 0.6723 |
| 0.9137 | 0.6029 |
| 0.8636 | 0.6978 |
| 0.8796 | 0.6613 |
| 0.8760 | 0.6861 |
| 0.9144 | 0.6160 |
| 0.8657 | 0.6730 |
| 0.8158 | 0.7438 |
| 0.8840 | 0.6620 |
| 0.9259 | 0.5562 |
| 0.8792 | 0.6481 |
| 0.8694 | 0.6803 |