Stage 1: Information Extraction from Natural Text documents

Name of Team Members:

1. Pradeep Kumar (pkumar46@wisc.edu)
2. Pulkit Kapoor (pkapoor4@wisc.edu)
3. Sonu Agarwal (sagarwal42@wisc.edu)

Links:

1. Link for all the 300 documents - https://github.com/pulkitkapoor98/CS839-DataScience/tree/master/Stage1/docs
2. Link for 200 docs used as development set - https://github.com/pulkitkapoor98/CS839-DataScience/tree/master/Stage1/Set-I-Docs
3. Link for 100 docs used as Test set - https://github.com/pulkitkapoor98/CS839-DataScience/tree/master/Stage1/Set-J-Docs
4. Link for the source code - https://github.com/pulkitkapoor98/CS839-DataScience/tree/master/Stage1/src

Entity Type of the Dataset:

For this project, we used names of any kind of educational institutions including universities, colleges, schools and schools of universities. We extracted 307 recent articles from The New York Times (https://www.nytimes.com/topic/subject/colleges-and-universities) which were related to universities and colleges. The occurrences of these entity were marked manually by enclosing them with a keyword-pair <uname></uname>. For eg: <uname>University Name</uname>.

Few examples (Excerpt from Articles):

1. James B. Milliken, <uname>CUNY</uname>'s chancellor, said in a statement.
2. Marcus the bomb-sniffing dog at the <uname>Butler University</uname> Police Department.
3. Earlier this week, <uname>Harvard University</uname> revealed that it had rescinded admissions offers to at least 10 students who shared offensive images within what they thought was a private Facebook group chat.

Details of documents we used for this project:

	Num. of documents	Number of Mentions
Development Set I	200	693
Test Set J	107	367
Total	307	1060

Pruning (Pre-processing):

All documents in the development set were pre-processed and possible candidates (of the entity) were generated using n-grams (n = 1:6) i.e., all phrases with consecutive words with 1 to 6 word length. The pruning rules were then run on initial candidates to eliminate obvious negative examples.

Few examples of pruning rules used –

1. Eliminate strings of length 1 character. (Example: A, I, etc.)
2. Eliminate strings with words in a negative list of prepositions, stop words, etc.
3. Eliminate strings with words starting with lowercase character except ['of', 'and']

Training:

We started designing our classifiers with 18 features based on the observations we made by looking at the development set documents and the properties of the entity to be extracted. All these features took Boolean values (0 or 1). Therefore, for each candidate (including positives and negatives), we obtained a feature vector of length 18. We fed this feature vector as the input to the below mentioned classifiers and used 5-fold cross validation to compute precision and recall on the validation set (Set I).

Initial values of Precision, Recall and F1 of all classifiers on Set I (Development Set):

Number of features - 18

Classifier	Precision	Recall	F1
Decision Tree	0.842583041958	0.412635866079	0.552352198074
Random Forest	0.826608708952	0.423339862867	0.558470453523
Support Vector Machine	0.851696832579	0.418097372606	0.559319484292
Linear Regression	0.81907978006	0.423094850786	0.556414373033
Logistic Regression	0.83984876643	0.41307553888	0.55181133901

We then analyzed the false positives and false negatives to improve precision and recall respectively. Based on our analysis and observation, we added some additional features to improve the accuracy and coverage. Based on the validation scores, 6 more features were added. We added some words in our negative list to prune out negative examples more effectively. Since SVM and Decision Tree performed better initially, we tweaked these two classifiers' hyperparameters to improve the scores further on validation set.

Final values of Precision, Recall and F1 of all classifiers on Set I (Development Set):

Number of features - 24

Classifier	Precision	Recall	F1
Decision Tree	0.935006037189	0.661293315502	0.774399585042
Random Forest	0.86929780204	0.67341278843	0.758600543713
Support Vector Machine	0.930958419726	0.666544506029	**0.776504826997**
Linear Regression	0.800357158036	0.691021003488	0.740973321939
Logistic Regression	0.847724749516	0.663638593586	0.744169800253

Finally, we got the best suited classifier SVM with precision 93%, recall 66% and maximum F1 score of 0.7765. We used this classifier to compute scores for the extraction of the entity from test set (Set J).

Final values of Precision, Recall and F1 of SVM classifier on Set J (Test Set):

Classifier	Precision	Recall	F1
Support Vector Machine	0.928853754941	0.640326975477	0.758064516129

Since we achieved the minimum required target of getting precision of 90% and recall of 60%, we did not need to perform any rule-based post-processing.