DATA SCIENCE REPORT

STAGE 1

DETECTING LOCATIONS FROM BBC ARTICLES

Team Members

1. **Saranya Baskaran**
2. **Shivanee Nagarajan**
3. **Varun Ramesh**

TOPIC AND ENTITY TYPE

In this stage, we are extracting **locations as an entity from BBC articles.** The locations were marked up using '#L' before and after the locations to train the model. -----> **(#LLOCATION_ENTITY#L)** <----------

Example entities that we located are -

- US
- New York
- China
- India
- California
- Chicago
- United States
- America

DATA AND MENTION SIZES

- **Total -**
  - Number of Documents --- **316**
  - Number of Mentions --- **2497**
- **SET I -**
  - Number of Documents --- **216**
  - Number of Mentions --- **1693**
- **SET J -**
  - Number of Documents --- **100**
  - Number of Mentions --- **804**

FEATURES ENGINEERING

We initially ran a cleaner script where we removed all punctuations excluding our markups and a delimiter to split lines. 10 fold cross-validation was performed on set I for different classes of machine learning algorithm that include - decision tree, random forest, support vector machine, linear regression and logistic regression.

In our first iteration, we were only able to hit precision of about 0.55 in both the Decision Tree and Random Forest classifier. The other classifiers performed below par when compared to these two algorithms.

The initial features that we used were -

1. Are all locations in the n-gram starting with uppercase?
2. Is it preceded by a preposition?
3. Stop words were not added to bigrams to reduce neg samples.
4. Followed by a possible location following word.

To pump the precision up to about 0.8 - 0.9 we used the context of a sentence to increase extraction of location. Following features based on noun, adjectives, adverbs and prepositions were added.

1. Each possible combination was encoded with a combination of three digits varying from 0 - 6.
2. We also extracted if a possible location noun was followed by a delimiter (.). This in turn would indicate if this particular word occured at the start or end of the string.
3. We also had a list of words in static lists which included possible prepositions, stop words, followed by words, substr, negative-noun list.

METRICS

Based on the above features, the Random Forest algorithm outperformed the others with the following metrics as it had recorded the highest F-1 score with 0.839. It was followed closely by the Decision Tree algorithm with a F-1 score of 0.837.

SELECTED FEATURE AFTER CROSS VALIDATION

M = RANDOM FOREST

METRICS (10 FOLD CV)	VALUES
Precision	0.821
Recall	0.642
F1 Macro Score	0.839

The following values were observed on the test set J of 100 documents.

FINAL CHOSEN CLASSIFIER

X - RANDOM FOREST

METRICS (TEST SET)	VALUES
Precision	0.914
Recall	0.681

RULE BASED POST-PROCESSING

No post processing rules were added as we were able to push precision above 0.9 and recall above 0.6 using the above features.