Categorical Text Classification based on keywords

Tingcong Liu, Angela Lam

October 2023

1 Group Member

- Angela Lam, puiyuyl2@illinois.edu
- Tingcong Liu, tl17@illinois.edu, captain

2 What is your free topic

Detecting events automatically from news data set is an essential endeavor in the pursuit of extracting rapidly evolving structured information. Current text classification methods mainly focused on distinguishing between different major events, but there are few studies about how to classify documents under the same overarching event.

Our task is a categorical text classification problem that giving a collection of news topic centered around one major event e.g. "Hong Kong protest", we try to classify these documents into more specific sub-categories such as the "Hong Kong airport protest" and the "Hong Kong University protest".

Our methods can be divided into three steps: 1. we employ keyword extraction techniques (e.g. TF-IDF) to extract keywords for every documents. 2. Based on keywords and date of the news, we weight these information and cluster them into different categories. 3. Based on clustered keywords and date, we classify these news into different sub events.

The outcome of our methods is different lists of documents, each distinctly grouped based on the sub event with same key event. To ensure the accuracy and relevance of our classifications, we will evaluate our work manually.

3 Language

The language we try to use is Python. We plan to explore Python's rich ecosystem of libraries for text processing and machine learning, such as scikit-learn and numpy. This makes it particularly suitable for our task of keyword extraction and text classification.

4 Workload

Since we are a group of 2, the total workload we think is 20*2 = 40 hours. The main tasks are listed below:

- Data collection and prepossessing (5 hours or more): Find news for our tasks data set. News should focus on a major event but can be classified into several sub events.
- Model Implementation (25 hours or more): Design our source code by using key word extraction, embedding and clustering.
- Evaluation (5 hours or more): Evaluate our results with the ground truth.
- Report (3 hours or more): Write documentations and report.
- Demo (2 hours or more): Make a demo.

I believe that total amount of work is definitely above 40 hours.