**CE306 - Information Retrieval**

Juntao Yu

February 2023

# Plagiarism

*You are reminded that this work is for credit towards the composite mark in CE306 and that the work you submit must therefore be your own. Any material you make use of, whether it be from textbooks, the Web or any other source must be acknowledged as a comment in the program, and the extent of the reference clearly indicated.*

# The context of your task

Suppose you are a consultant for a search solution company, and you are required to design processing pipelines for the Essex Move Search companies. As the company name suggests, they primarily focus on movies, hence a movie dataset is used for your task. The company want you to design the processing pipelines using Elasticsearch, with a small part to be done via a python library.

**Dataset**: The dataset[[1]](#footnote-1) contains descriptions of 34,886 movies from around the world. The plot summary descriptions are scraped from Wikipedia. This freely available dataset is provided to the global research community to apply recent advances in information retrieval and other AI techniques to generate models that can return a movie title based on an input plot description or return movie titles with plots similar to the user query. (WARNING: May contain spoilers!!!)

**Your task**

1. **Elasticsearch (45%)** The task is to design analyzers for all fields with a focus on plot field, in which the company specified that a minimum of 5 pipelines (Tokenization, Case folding, Stopword removal, Word N-gram extraction, Stemming) and TF-IDF scores need to be used in your design.
   * **Hint 1:** you are not expected to use those pipelines in a single analyzer, multiple analyzers can be used together for a single field via the multi-field support of Elasticsearch.
   * **Hint 2:**  you need to use the TF-IDF equation learned from the lecture, it is not provided as a standard similarity score in elastic search, you will need to use the [scripted similarity](https://www.elastic.co/guide/en/elasticsearch/reference/7.17/index-modules-similarity.html#scripted_similarity) to implement it.
   * **Hint 3:** the design is worth 15% of the marks here, so do take it seriously.
2. **Named entity recognition (20%)** named entities are important for searching, your task here is to use a python NLP library to extract the named entities from the plot field and then store them in a new column (plot\_ner). After that, you need to design an analyzer to read from the plot\_ner field to make the named entities searchable. The named entities are required to be searched by exact match. I.e. when the user searches on the plot\_ner field using the search query “New York” only documents containing the named entity “New York” should be returned.
   * **Hint 1:** The new plot\_ner column should only consist of named entities. For instance, if you find three named entities from a plot field (“New York”, “Tom Brown” and “Colchester”) your plot\_ner column should be “New York; Tom Brown; Colchester”.
   * **Hint 2:** Elasticsearch does have a tokenizer available that allows you to specify the split pattern e.g. the semicolon. The input text will be split into index terms according to the pattern you specified.
   * **Hint 3:** Even if you are not sure how to use python NLP libraries you can still finish the second half of the task by designing an analyzer to process the above example and show that you can output three terms (“New York”, “Tom Brown” and “Colchester”) as expected.
3. **Indexing (10%)** Once you’ve designed your analyzers add them to the mapping and start to process the data with your processing pipelines.You should collect the first 1000 documents of the dataset. And import them using the analysers you designed above.
4. **Searching (15%)** Once you have indexed the collection you want to be able to search it. The task is to create 3 textural queries that the user might come up with and write the corresponding Elasticsearch queries.
5. **Documentation (10%)** The last 10% is awarded to the presentation of your report, e.g. is the description of your report easy to follow, whether have you used the appropriate method (e.g. screenshots) to present your results etc.

# Submission

You should submit two separate files (Do ***NOT*** zip them):

* Report (**use the template below**)
* Codes **(copy both Kibana and python code in a single .txt format)**

*The guidelines about late assignments are explained in the students’ handbook.*

**CE306 - Information Retrieval 2023**

**Assignment 1**

Student ID

# Elasticsearch

*Include here the details of how you did this step including the intuition about your design. You can show how the analyzer works by giving sample text input (remember we did this in Lab 2). You need to include screenshots to demonstrate your approach works properly.*

# Named entity recognition

*Include here the details of how you did this step, e.g. which NLP library you used, how you processed the datasets, and how did you design your analyzer to fulfil the requirement. You need to include screenshots to demonstrate your approach works properly.*

# Indexing

*Include here the details of how you indexed your dataset, including any issue you had and how you faced it. You need to include screenshots to demonstrate your approach works properly. For instance, you could use match\_all to show there are exactly 1000 records indexed and use GET /your\_index\_name/\_mapping to show the mapping you designed above is correctly applied to your index.*

# Searching

*Include here the details of your textural and Elasticsearch queries as well as the system outputs. You need to include screenshots to clarify. The search needs to have a certain complexity in order to get a good score, the complexity of the search should be equivalent to the exercise part of Lab 1.*

1. <https://www.kaggle.com/jrobischon/wikipedia-movie-plots?select=wiki_movie_plots_deduped.csv> [↑](#footnote-ref-1)