**CE306 - Information Retrieval**

Juntao Yu

February 2022

# 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

The idea of this assignment is that you apply the information retrieval knowledge you acquired during this term and put it into practice. You are already familiar with Elasticsearch. You also know the processing steps that turn documents into a structured index, commonly applied retrieval models and you know the key evaluation approaches that are being employed in IR. Now is a good time to put it all together.

**Scenario**: The dataset[[1]](#footnote-2) 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**

This task comes in stages. Marks are given for each stage. The stages are as follows:

* **Indexing (20%)** The first step for you will be to obtain the dataset. Once you have done so choose a **sample of 1000** articles as your corpus (the simplest thing is to use the first 1000 documents). This will need to be imported to Elasticsearch later (after you defined your processing pipeline).
* ﻿**Tokenization and Case folding (10%)** The next step should be to transform the input text into a normal form. For this task you are required to use Elasticsearch’s build-in analyzers or other libraries (as learned in Lab 2) to tokenize the document and perform case folding to the tokens.
* ﻿**Selecting Keywords (20%)** One aim of your system is to identify the words *and* phrases in the text that are most useful for indexing purposes. For this task you are required to do include stopword removal and (n-gram extraction or named entity recognition). As well as apply *tf.idf* as part of your selection and weighting step. (**Hint**: the stopword removal, n-gram extraction can be done with Elasticsearch’s build-in tokenizer and tf.idf scores can also be configured using Elasticsearch similarity module.)
* **Stemming or Morphological Analysis (10%)** Writing word stems to the database rather than words allows to treat various inflected forms of a word in the same way, e.g. *bus* and *busses* refer to exactly the same thing even though they are different words.
* **Searching (10%)** Once you have indexed the collection you want to be able to search it. You can do that on the command line (like in Lab 1), but it would be easier to do it Kibana’s dev tool. The task is to create 3 textural queries that the user might come up and write the corresponding Elasticsearch queries.
* **Working with Elasticsearch API (10%)** Finally the 10% will be given if you could make everything work with the Elasticsearch API.

You will have noticed that the percentages above only add up to **80%**. This is because one of the important aspects of the project is that your work should be **well documented** and your code well commented. **20% of your mark will come from this.**  The report should contain:

* Instructions for running your system
* Screenshots illustrating the functionality you have implemented
* A description of the document collection you have chosen
* Discussion of your solution focusing on functionality implemented and possible improvements and extensions.

The report does not need to be long as long as it addresses all the above points.

# Software

The backend search engine to be used is *Elasticsearch*. Apart from that you are free to write additional code in any language of your choice, and employ any open source tool that you find suitable.

# Submission

You should submit:

* Report (**use the template below**)
* Code

The submission of all two completed tasks should be submitted as a single *zip file* via the electronic submission system. Please check the details of the submission deadline with the CSEE School Office.

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

**CE306 - Information Retrieval 2022**

**Assignment 1**

Student ID

# Instructions for running your system

*Include here instructions to run your system, this could be as simple as start Elasticsearch and Kibana if you are not using Elasticsearch API. You may include screenshots to clarify.*

# Indexing

*Include here the details of how you download your datasert and index it including any issue that you had and how did you face it. Explain which documents you have selected for your experiments. You may include screenshots to clarify.*

﻿Tokenization and Normalization

*Include here the details of how you did this step including any issue that you had and how did you face it. Present examples to show how your system works, e.g., if you use elastic analyzers, you can show how the analyzer works by given sample text input (remember we did this in Lab 2). You may include screenshots to clarify.*

﻿Selecting Keywords

*Include here the details of how you did this step including any issue that you had and how did you face it. Present examples to show how your system works, e.g., if you use elastic analyzers, you can show how the analyzer works by given sample text input (remember we did this in Lab 2). You may include screenshots to clarify.*

# Stemming or Morphological Analysis

*Include here the details of how you did this step including any issue that you had and how did you face it. Present examples to show how your system works, e.g., if you use elastic analyzers, you can show how the analyzer works by given sample text input (remember we did this in Lab 2). You may include screenshots to clarify.*

# Searching

*Include here the details of your textural and Elasticsearch queries as well as the system outputs. You may include screenshots to clarify.*

1. <https://www.kaggle.com/jrobischon/wikipedia-movie-plots?select=wiki_movie_plots_deduped.csv> [↑](#footnote-ref-2)