ASSIGNMENT

Review Data Analysis and Processing

AI6122 Text Data Management and Processing

2021/2022 Semester 1

NANYANG TECHNOLOGICAL UNIVERSITY

1 Objective

The objective of this assignment is to let you getting familiar with the main components in end-to-end text management and processing applications, the challenges faced by each component and the solutions. Through this assignment, you shall also get hands on experiences on various packages available for information retrieval and natural language processing tasks.

2 Assignment Format

1. This is a group assignment. Each group has 4 to 5 students.
2. One report is to be submitted by *each group* and all members in the same group receive the same grade. However, **contributions of individual members** to the assignment shall be *cleared indicated* in the report. Group size is not a factor in grading.
3. You may use ANY programming language of your choice, *e.g.,* Java, Python, C#.
4. You may use any NLP, IR, and Machine Learning library/software as long as its license allows free use for education and/or research purpose. Some example packages are listed below. However, relational database like MySQL is not allowed.
    - All-in-one library: NLTK (Python), spaCy (Python), LingPipe (Java), Stanford NLP(Java), OpenNLP (Java)
    - Indexing and Search: Lucene (Java)

3 Assignment (100 marks)

The assignment consists of the following components: Dataset Analysis (50 marks), Development of a Simple Search Engine (20 marks), Extraction of Indicative Adjective Phrases (20 marks), and Application (10 marks).

3.1 Dataset

We will use the Yelp Open Dataset (`https://www.yelp.com/dataset`). We will only process the **review.json** file under the **JSON** dataset.[1] Each review is one line in the JSON file, and each review has the following components: review_id, user_id, business_id, stars, date, text, useful, funny, cool. Detailed explanation of these components for **review.json** is available at `https://www.yelp.com/dataset/documentation/main`.

3.2 Dataset Analysis (50 marks)

Tokenization and Stemming. Select a business b_1 randomly from the dataset, then extract all reviews for b_1 to form a small dataset B_1. Show word frequency distributions in B_1 before and after stemming, respectively.

[1] Due to its big size, one download per group is strongly encouraged.

You may choose the stemming algorithm implemented in any toolkit. You may consider to plot the word frequency distributions in log-scale. Repeat the same process for another randomly selected business b_2. Discuss your findings based on your plots. List the top-10 most frequent words (exclude stopwords) before and after performing stemming, for each of the two selected businesses. Discuss your findings.

POS Tagging. Randomly select 5 sentences from the dataset, and apply POS tagging. Show and discuss the tagging results. You need to show two sets of tagging results by using two different tagging methods (*e.g.,* results from two tagging methods implemented in one toolkit, or results from two toolkits).

Writing Style. Randomly sample the following: (i) two posts from StackOverflow, (ii) two posts from Hardwarezone, (iii) two news articles from ChannelNewsAsia. Discuss the differences on their writing styles (*e.g.,* is the first word in a sentence capitalized; do sentences follow good grammars; are the proper nouns capitalized; etc). You need to provide the URLs of the sampled posts/articles in your report. Can the tools used for tokenization and POS tagging be directly applied to posts in StackOverflow?

Most frequent ⟨ Noun - Adjective ⟩ pairs for each rating. Each review has a "star" rating in the range of 1 to 5. Randomly select 50 reviews (one from each business) of rating 1, extract the top-10 most frequent noun-adjective pairs from the sentences in these selected reviews. Example noun-adjective pairs are service-great, food-delicious, that appear in the same sentence. Do the same for 20 reviews of ratings 2, 3, 4, and 5, respectively. Discuss your results and limitations of your method.

3.3 Development of a Simple Search Engine (20 marks)

Write a search engine to index and search reviews, by using Lucene or other libraries specific to IR.[2] In this part of the assignment, you may use (i) One main IR specific library for most of the operations; (ii) Any other third-party libraries if and only if the main library does not provide the required functionality; and (iii) Any stopword list of your choice.

In this search engine, each review is a "document" and you may discuss what filed(s) (*e.g.,* review_id, user_id, business_id, stars, date, text, useful, funny, cool) shall be indexed and searchable. Detail your choice of parsing/linguistic processing on the words/terms in the chosen fields, *e.g.,* whether to perform stemming, case folding, stopword removal. Based on the number of "documents" to be indexed in the dataset, collect the time needed to index every 10% of the documents. Discuss your findings on the indexing time.

Your search engine should at least support free text keyword queries (including single keyword query and phrase query) on the "text" field in a review. Top N (the number of N is configurable) results should be returned via the console[3] along with rank, scores, docID, and snippets whenever possible. Randomly choose a few queries (including both single keyword query and phrase queries), discuss whether the results returned by the search engine are as expected. You may also record the time taken to process a query.

3.4 Extraction of Indicative Adjective Phrases (20 marks)

Given all reviews of a randomly selected business b_1, we would like to summarize b_1 by extracting the most indicative adjective phrases from its reviews. The indicative adjective phrases are the phrases that appear often in b_1's review, but relatively less often in other reviews. Manually go through all reviews of b_1, sample

[2] See `http://en.wikipedia.org/wiki/List_of_information_retrieval_libraries` for a list.

[3] Note, a text-based command line system is sufficient; a GUI or web-based interface to the search engine is NOT encouraged.

some other reviews in the dataset, and discuss whether the selected adjective phrases indeed reflect the unique characteristics of b_1. Discuss the results and your findings.

3.5 Application (10 marks)

Define and develop a simple NLP application based on the dataset. An example application is to detect the sentences containing *Negation Expression*. Negation is often expressed through negative words such as no, not, never, none, nobody. You may define your own application with similar (estimated) difficulty level. Note that, application here means a small tool (or piece of code) to analyze or to mine the data. Application here does not mean a web-based application or mobile app. Command line interface is sufficient, and GUI or Web-based interface does not contribute to grading.

4 Submission of Report and Source Code

4.1 *Final Report in Hardcopy*

- The hardcopy report must be submitted on or before **25 Oct 2021** (Monday, Week 11), through SCSE General Office.
- The report must use the provided cover page, and the main content shall be formatted following the ACM "sigconf" proceedings templates[4] (either MS Word or Latex). The main content of the report ***must not exceed 10 pages***, *i.e.,* excluding cover page and appendix.
- DO NOT include in your report all the source code and complete results sets. However, you must include *code snippets* which are important for the main functions for your task. You should cite all third-part libraries used in your assignment.
- The report shall be printed in double-sided format whenever possible. A plastic cover or ring-binding leads to 2% penalty.
- Before submission, please read the hardcopy of your own report. **Make sure any words or pictures in your report are readable**.

4.2 *Final Report in softcopy, Source Code, and Documentation*

- An AI6122.zip file containing the following files and folder shall be submitted: Report.PDF, Readme.txt, SourceCode.
  - Report.PDF shall be the same as the hardcopy report submitted.
  - Readme.txt shall include
    - A link to download the third-party library if you used any in your assignment.
    - An installation guide on how to setup your system, and how to use your system (*e.g.,* command lines, input format, parameters).
    - Explanations of sample output obtained from your system.

[4] `https://www.acm.org/publications/proceedings-template`

  - SourceCode folder shall contain only your source code. The dataset and the libraries shall **NOT** be included in the softcopy submission to minimize the file size.

- Softcopy submission deadline: ***25 Oct 2021 11:59PM***. Late submissions are allowed but will be penalized by 5% every calendar day (until zero). The softcopy can be submitted for at most three times, only the last submission will be graded and time-stamped.