ASSIGNMENT

Online Forum Data Processing

CZ4045 Natural Language Processing

2017/2018 Semester 1

NANYANG TECHNOLOGICAL UNIVERSITY

1 Objective

The objective of this assignment is to let you getting familiar with the main components in an end-to-end NLP application, the challenges faced by each component and the solutions. Through this assignment, you shall also get hands on experiences on various packages available for NLP 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.
3. You may use ANY programming language of your choice, *e.g.,* Java, Python, C#, C++.
4. You may use any NLP and Machine Learning library/software as long as the license allows free use for education and/or research purpose. Example packages are listed below.
    - All-in-one library: NLTK (Python), LingPipe (Java), GATE (Java), Stanford NLP(Java), OpenNLP (Java)
    - HTML/XML processor: jsoup (Java), Python HTML/XML tools (Python)
    - Conditional Random Fields (CRF) library: CRF++ (C++), CRFsuite (C++, Python)
    - Machine learning library: Weka (Java), CMU Rainbow (C), SVM^{light} (C), libsvm (C++, Java), milk (Python)

3 Assignment (100 marks)

The assignment consists of the following components: Dataset Collection (10 marks), Dataset Analysis and Annotation (30 marks), Development of a Tokenizer (30 marks), Further Analysis (20 marks), and Application (10 marks).

3.1 Dataset Collection (10 marks)

We will use data from Stack Overflow `http://stackoverflow.com`, an online forum dedicated to programming related questions and answers. You may get a data dump from `https://archive.org/details/stackexchange` and then select a subset of questions with answers. You may also write your own crawler to collect questions/answers from the website. Your data set shall satisfy all the following three conditions:

- The dataset shall contain at least 500 threads of discussion.
- All the selected threads are on *one particular Programming Language*, *e.g.,* Java, Python, JavaScript, PHP, or another language of your choice.
- Each selected thread should have at least two posts. A post can be either a question or an answer.

Your report shall document how the dataset was collected and justify it satisfies the three conditions. Your report shall also show the statistics of the dataset, including number of questions, number of answers, and the distribution of the answers *e.g.,* what is the percentage of questions having one, two, or more answers, and the length of the posts in number of words (using an off-the-shelf tokenizer).

3.2 Dataset Analysis and Annotation (30 marks)

Stemming. Process the dataset using an off-the-shelf toolkit. From the dataset, identify the top-20 most frequent words (excluding the stop words) before and after performing stemming. Stop words are the words that are commonly used but do not carry much semantic meaning such as *a*, *the*, *of*, *and*. For each of the top-20 most frequent stems, list their original words from which the stem is reached. Discuss your results.

POS Tagging. Randomly select 10 sentences from the dataset, and apply POS tagging. Show and discuss the tagging results.

Token Definition and Annotation. Define what a token is, in the context of Stack Overflow discussion. As an example, you may consider whether each of the following is a single token:

- String
- java.lang.String
- String ( )
- String()
- String (byte[ ] bytes)
- indexOf(int ch)
- int indexOf(int ch)

Randomly select at least 100 posts, and then annotate the sentences based on your token definition. The annotated dataset will be used as ground truth in the next section. You may use `http://brat.nlplab.org/` for your annotation task.

3.3 Tokenizer (30 marks)

Your task is to tokenize all sentences in your dataset. Your group may design different techniques to tokenize the data based on your definition of "token". You may consider different techniques like regular expression and Conditional Random Field (CRF). Note that, if you plan to use CRF, there is NO need to develop your own CRF implementation. You may use any of the CRF libraries.

To evaluate the effectiveness of tokenzier, a cross-validation is often applied if you use a learning methods (*e.g.,* not using regular expressions). In K-fold cross validation, the annotated data is partitioned into k non-overlapping portions. Among them, $k-1$ portion are used to train the model and the remaining one portion is used to evaluate the effectiveness of the model. For instance, if $k = 4$, then each time 3/4 of the annotated data are used in training, and the remaining 1/4 of the

annotated data are used for evaluation. By using each of the 4 folds as evaluation data, the model is evaluated on all annotated instances. Then the performance of the model can be evaluated by using *Precision*, *Recall*, and F_1. If you use regular expressions, you can evaluate the results based on the 100 annotated posts. Report the *Precision*, *Recall*, and F_1 of your model, and analysis the errors (*e.g.,* case studies on false positives and false negatives).

3.4 Further Analysis (20 marks)

Irregular Tokens. Tokenize the dataset using your own tokenizer. Identify the top-20 most frequent tokens which are NOT standard English words (including their morphological forms).

POS Tagging. Randomly select 10 sentences from the dataset where each sentence contains at least one irregular token. Apply POS tagging on the sentences based on your own tokenization results. Show and discuss the tagging results.

3.5 Application (10 marks)

Define and develop a simple NLP application based on the dataset. An example application is to detect sentences containing *Negation Expression* using regular expressions. 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.

4 Submission of Report and Source Code

4.1 *Final Report in Hardcopy*

- The hardcopy report must be submitted on or before **2 Nov 2017** (Thursday, Week 12), through SCSE General Office. The report shall be formatted following the ACM SIG Proceedings Templates (either MS Word or Latex), **maximum 8 pages** including appendix if any.[1] 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 of your system. 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.
- Make sure any words or pictures in the report is readable.

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

- A CZ4045.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

[1] `http://www.acm.org/sigs/publications/proceedings-templates`

    - A link to download the third-party library if you used any in your assignment
    - A link to download the datasets used in your assignment, one for the 500 posts and the other for the 100 annotated posts.
    - 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.
  - SourceCode folder shall contain all your source code. The dataset and the libraries shall **NOT** be included in the softcopy submission to minimize the file size.

- Softcopy submission deadline: ***2 Nov 2017 11:59PM***. Late submissions are allowed but will be penalized by 0.5% 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.