Information Retrieval

COMP 479 Project 1
Text preprocessing with NLTK Report

Lin Ling (40153877)

A report submitted in partial fulfilment of the requirements of Comp479.

Concordia University

INDEX

1. First Module: Read and Extract Raw Text

- Source Code: read_extract_reuters.py
- Output files: COMP479Project1/mod1_data

Read the “sgm” files in data folder: reuters21578 which download from http://www.daviddlewis.com/resources/testcollections/reuters21578/ , split the content of the file to articles and extract the raw text from the each article. And then write the raw text to each raw text file which includes in mod1_data folders for next module to handle.

2. Second Module: Tokenize

- Source Code: tokenize_text.py
- Output files: COMP479Project1/mod2_data

Read the files in data folder: mod1_data, use word_tokenize method from nltk. tokenize package to split the raw text to word and punctuation. And then write the tokens to a file which includes in mod2_data folders for next module to handle.

3. Third Module: Make all text lowercase

- Source Code: lowercase_text.py
- Output files: COMP479Project1/mod3_data

Read the files in data folder: mod2_data, use lower method to make each word of raw text lowercase. And then write the tokens to a file which includes in mod3_data folders for next module to handle.

4. Fourth Module: Apply Porter stemmer

- Source Code: porter_stemmer.py
- Output files: COMP479Project1/mod4_data

Read the files in data folder: mod3_data, use stem method in PorterStemmer class to stem each word of raw text. And then write the word stem to a file which includes in mod4_data folders for next module to handle.

5. Fifth Module: Remove stop words

- Source Code: remove_sw.py
- Output files: COMP479Project1/mod5_data

Read the file in data folder: mod3_data, use list comprehension to filter the stop words read from a stop_word.txt out of the words stem. And then write the rest of words stem to a file which includes in mod5_data folders.

6. Test and Driver

- Source Code: test.py
  main.py

Test Module is aimed at doing the unit test for each module by each function. The shortage of this part is the test cases could be taken more coverage to make the system more robust. The Main Module is aimed at demonstrating the whole process of the system which is presented in the demo report.