Assignment 3

CS 734: Introduction to Information Retrieval

Fall 2017

Grant Atkins

Finished on November 9, 2017

1

Question

```
6.1. Using the Wikipedia collection provided at the book website, create
a sample of stem clusters by the following process:
    1. Index the collection without stemming.
    2. Identify the first 1,000 words (in alphabetical order) in the index.
    3. Create stem classes by stemming these 1,000 words and recording
    which words become the same stem.
    4. Compute association measures (Dice?s coefficient) between all
    pairs of stems in each stem class. Compute co-occurrence at the
    document level.
    5. Create stem clusters by thresholding the association measure.
    All terms that are still connected to each other form the clusters.

Compare the stem clusters to the stem classes in terms of size and the
quality (in your opinion) of the groupings.
```

Answer

For this problem as well as other problems that used Galago it should be noted that I used Galago version 3.12 [4]. I also used the wiki-small html files as the input for the Galago search engine.

To handle the first two parts of this problem I decided to take advantage of the Galago search engine for indexing and sorting the index created by it. I then used galago to dump the unique keys from its inverted index using the following command:

$ galago dump-keys index/postings > dump-keys.csv

Galago creates an inverted index and it should be noted that from the keys dumped there were sequences of different characters but not all of them were actually words but rather numbers or misspellings. To compensate for this I used a python library called Pyenchant [5] to go through each and see if the the words were actually english and spelled correctly. The code to complete this is in **spellcheck.py** shown in Listing 1. This however still had numbers inside of it. To get to the english words starting with the letter 'a' I skipped to line 8578 and took the next 1000 words saved this to a file named **top-1k-words.txt**.

```
import enchant
import csv


def spellcheck():
    with open('./data/dump-keys.csv', 'r') as f, \
            open('./data/spell-checked-keys.csv', 'w') as out:
        reader = csv.reader(f)
        writer = csv.writer(out)
        d = enchant.Dict("en_US")
        for row in reader:
            word = row[0]
            if d.check(word):
                writer.writerow([word])


if __name__ == "__main__":
    spellcheck()
```

Listing 1: Spellcheck code in python

To stem each of the words in my list I used the NLTK python library [3] and specifically the PorterStemmer function to stem each word. I made a dictionary where each stem was a key and the stems then had an array of words that fell into that stem class and saved this to a json file named **stems.json**. A small sample of this is shown in Figure 1. I then used Galago again to find the document ids where the words had been used using the following Galago command:

$ galago dump-index index/postings > dump.csv

I then merged the previous dictionary with the document ids found in this step so I could use it to find the co-occurrence and Dice's Coefficient as described below. The code to perform these steps is in the file named **stemmer.py**.

Dice's Coefficient is described in our textbook [2] as:

$$\frac{n_{ab}}{n_a + n_b}$$

To compute Dice's Coefficient

Figure 1: Subset of stems generated from Porter Stemmer

2

Question

Answer

3

Question

Answer

4

Question

Answer

5

Question

Answer

References

[1] Atkins, Grant. "CS734 Assignment 3 Repository" Github. N.p., 9 November 2017. Web. 9 November 2017.https://github.com/grantat/cs834-f17/tree/master/assignments/A3.

[2] B. Croft, D. Metzler, and T. Strohman. "Search Engines: Information Retrieval in Practice." Pearson, 2009. Web. 14 October 2017. ISBN 9780136072249.

[3] Bird, Steven, Edward Loper and Ewan Klein (2009). "Natural Language Processing with Python" O?Reilly Media Inc. Web. 9 November 2017.http://www.nltk.org/.

[4] "The Lemur Project - Galago". Web. 9 November 2017. https://sourceforge.net/p/lemur/wiki/Galago/.

[5] "Pyenchant". Web. 9 November 2017. http://pythonhosted.org/pyenchant/.