20-00-0546-iv Foundations of Language Technology

Homework 3
Homework Writing structured programs

19. November 2020

Please send Python Notebook files (`.ipynb`) for programming parts and plain text or PDF for essay questions.

In case your submission consists of several files, compress these to a `zip`-file. Indicate clearly which submission corresponds to which question. Include comments in your program code to make it easier readable.

The deadline for the homework is **Thursday, 26.11.20 18:00 CET**.

3.1 Homework

Homework 3.1 *(10 points) Design an algorithm to find the "statistically improbable phrases" (SIP) of a document collection*[1][2]*. The main idea behind SIPs is that if a phrase occurs much more often in a document than this is statistically expected (hence the name "statistically improbable phrases"), it might be a good indicator of what a document is about. For example, we may look at a corpus with 1,000,000 words and see that the word "car" appears 20 times. If now the word "car" also appears 20 times in a document with 1000 words, this seems improbable.*

To formalize the counting of words a bit, we may use so called contingency tables. They are a structured way to display how often a word occurs in certain documents. As we are only interested in the counts in the current document as well as the counts in the overall collection, we will use a 2x2 table:

	Document (Foreground Corpus)	*Other Corpus (Background Corpus)*	
word	*A*	*B*	
all other words	*C - A*	*D - B*	
	C	*D*	*N = C + D*

We can now measure how "improbable" is a certain phrase in a document (also called foreground corpus) w.r.t. some corpus (called background corpus) using "Log-Likelyhood" (LL) as defined by (Rayson & Garside, 2000). In terms of the contingency table, it is defined as:

$$LL = 2 \cdot (A \cdot \log_2(\frac{A}{E_1}) + B \cdot \log_2(\frac{B}{E_2}))$$

where $E_1 = \frac{C \cdot (A+B)}{N}$ *and* $E_2 = \frac{D \cdot (A+B)}{N}$*. For simplification, you may limit phrases to bigrams.*

(a) *Use the NLTK's Brown corpus as your background corpus to represent the distribution of phrases in the language.*[3] *Access the data in an NLTK corpus with the following methods:*

```python
# Get the whole corpus as a list of words
corpus_words = nltk.corpus.corpus_name.words()

# Access ids of texts that are available in the corpus:
print("File ids: ", nltk.corpus.corpus_name.fileids())

# Retrieve a certain text by id
file_words = nltk.corpus.corpus_name.words('some_file.txt')
```

(b) *Initialize two frequency distributions of bigrams: one of 'Alice in Wonderland' ('carroll-alice.txt', in the Gutenberg corpus) as your foreground text (`fdist_fg`) and one of the Brown corpus (`fdist_bg`).*

(c) *Using the frequency distributions, determine how to get all the required counts to compute the log likelihood scores and write a function `compute_LL(phrase, fdist_fg, fdist_bg)`.*

(d) *Given the 'Alice in Wonderland' text, compute the LL score for each bigram with the `compute_LL` method and output the 10 most "improbable" phrases.*

[1] "Amazon.com's Statistically Improbable Phrases, or "SIPs", are the most distinctive phrases in the text of books in the Search Inside!™ program. To identify SIPs, our computers scan the text of all books in the Search Inside! program. If they find a phrase that occurs a large number of times in a particular book relative to all Search Inside! books, that phrase is a SIP in that book. SIPs are not necessarily improbable within a particular book, but they are improbable relative to all books in Search Inside!. For example, most SIPs for a book on taxes are tax related. But because we display SIPs in order of their improbability score, the first SIPs will be on tax topics that this book mentions more often than other tax books. For works of fiction, SIPs tend to be distinctive word combinations that often hint at important plot elements." (http://www.amazon.com/gp/search-inside/sipshelp.html, archived version)

[2] See NLTK-book page chapter 4, exercise 35, page 177

[3] You can, of course, test your function on other corpora on your own. However, for the homework submission you must use the Brown corpus.