CS418: INTRODUCTION to NLP
Lab 01B – Extracting keywords by TF*IDF

1 Problem

Your goal in this assigment is to write a program using TF*IDF to extract keywords in BBC news documents.

1.1 Preprocessing

Text preprocessing are needed for transferring text from human language to machine-readable format for further processing. In our assignment, the preprocessing includes:

- converting all letters to lower case,
- removing numbers,
- removing punctuations,
- removing white spaces,
- tokenization,
- removing stop words,
- stemming or lemmatization.

All text processing have to be done in `def preprocess(document)` function. You can use regular expression or `nltk` library.

1.2 Extracting keywords using TF*IDF

TF*IDF, short for term frequency — inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in searches of information retrieval, text mining, and user modeling.

1. Term frequency: the number of times a term (word/token) occurs in a document or $f_{t,d}$

$$tf(t,d) = log(1 + f_{t,d})$$

2. Inverse document frequency: is the factor that diminishes the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely.

$$idf(t, D) = log \frac{N}{1 + |\{d \in D : t \in d\}|}$$

with N: total number of documents in the corpus, $N = |D|$
$|\{d \in D : t \in d\}|$: number of documents where the term t appears

There are three functions involved:

- `def calculate_idf(corpus)`: given the corpus, your function calculates `idf` score for every words.
- `def calculate_tf(word, words_in_document)`: given all words in a document, calculate tf for word.
- `def process(corpus, top_key)`: given the corpus, calculate top_k (here $k = 5$) keywords for each document that have the best tf*idf score.

2 Implement

We used BBC dataset for this task. The dataset consists of 2225 documents from the BBC news website corresponding to stories in five topical areas from 2004-2005. However, because TF*IDF (and our task) is the super naive method, we used only 401 documents in tech class (see in `bbc/tech/` folder) to simplify the problem.

To execute your program:

```
>> cd pa02
>> python keyword.py bbc/tech/ output.csv
```

The output.csv file will include 2 columns: the first column contains the name of each text file, the other column is a list of keywords extracted from the corresponding text.