

WSM Project 1: Ranking by Vector Space Models

1. [40 points] Vector Space Model with Different Weighting Schemes & Similarity Metrics (Please DON'T use any off-the-shelf packages or functions)

The example codes given in Week 4 demonstrate how an IR system works via Vector Space Model. Below are some steps in the codes:

1. Stemming & Removing Stop Words ([English Stop Words](#)); & Indexing
2. Transfer Queries into a Vector
3. Transfer Documents into Vectors
4. Calculate the Similarity between the Query Vector and the Document Vectors
5. Rank the Documents according to the Similarity scores

Now you are asked to develop a retrieval program that is able to retrieve the relevant news to the given query from a set of 8,000 [English News](#) collected from *reuters.com* according to different weighting schemes and similarity metrics. In the given dataset, each file is named by its News ID and contains the corresponding news title and content, as shown in below:

```
~/wsm/wsm_project1 > cat ./EnglishNews/News1.txt
Breakingviews - Corona Capital: U.S. airlines
NEW YORK/LONDON/HONG KONG (Reuters Breakingviews) - Corona Capital is a column updated throughout the day by Breakingviews c
hort, sharp pandemic-related insights.
```

There are the two combinations you're asked to implement. For each combination, please retrieve the top 10 results and scores. Here is an example result for the query "**Youtube Taiwan COVID-19**":

◦ [20/40 points] TF-IDF Weighting (Raw TF in course PPT) + Cosine Similarity

```
TF-IDF Weighting + Cosine Similarity:
NewsID      score
News7403.txt 0.446431
News1240.txt 0.416358
News668.txt  0.367137
News623.txt  0.308268
News2401.txt 0.292633
News7570.txt 0.292633
News7362.txt 0.284659
News447.txt  0.278903
News1679.txt 0.276170
News796.txt  0.270135
-----
```

◦ [20/40 points] TF-IDF Weighting (Raw TF in course PPT) + Euclidean Distance

```
TF-IDF Weighting + Euclidean Distance:
NewsID      score
News2925.txt 11.958069
News1830.txt 12.122651
News2424.txt 12.234900
News7207.txt 12.234900
News7467.txt 12.700265
News1497.txt 13.023024
News7098.txt 13.027066
News2401.txt 13.051226
News7570.txt 13.051226
News28.txt   13.890028
-----
```

2. [10 points] Relevance Feedback

Relevance Feedback is an IR technique for improving retrieved results. The simplest approach is Pseudo Feedback, the idea of which is to feed the results retrieved by the given query, and then to use the content of the fed results as supplement queries to re-score the documents.

In this work, you're asked to use the Nouns and the Verbs within the first document of the above **Method 1** (e.g. TF-IDF Weighting + Cosine Similarity) for Pseudo Feedback. The new query term weighting scheme is **[1 * original query + 0.5 * feedback query]**. Please try to use the new query to re-rank the documents.

For instance, suppose the index vector is ["network", "computer", "share", "ask", "soccer", "song"], the query is "network", and the content of the feedback document is:

Jimmy shares songs via the computer network.

Then we will get a new query vector like this:

$$1 * [1, 0, 0, 0, 0, 0] + 0.5 * [1, 1, 1, 0, 0, 1] = [1.5, 0.5, 0.5, 0, 0, 0.5]$$

In this work, you may need to use the Python NLTK package. For more details, please refer to [this link](#).

3. [20 points] Vector Space Model with Different Scheme & Similarity Metrics in Chinese and English

In this part, you are asked to retrieve the relevant news to the query from a set of 2,000 [Chinese News](#) collected from *chinatimes.com* and *setn.com* according to different weighting schemes (TF and TF-IDF) and **cosine similarity metric**.

Here is the example result of the query "烏克蘭 大選":

```
TF-IDF Weighting + Cosine Similarity:
NewsID      score
News200892.txt 0.196025
News200049.txt 0.187055
News200847.txt 0.172412
News200908.txt 0.157098
News200056.txt 0.153941
News200071.txt 0.153664
News200137.txt 0.150758
News200565.txt 0.147686
News200053.txt 0.136437
News200000.txt 0.132843
-----
```

Hint: You may use [Jieba](#) or [CKIP](#) to split the Chinese word segments.

4. [30 points] Evaluation IR system

In this part, we'll focus another [smaller dataset](#), which have 1460 documents, 76 queries and their labelled relevant documents.

You need to implement the following metrics on this dataset:

- [10 points] Recall@10
- [10 points] MAP@10
- [10 points] MRR@10

by using vector space model and trying some NLP technique e.g. stemming, remove stop word ...

Here is the example result :

```
tfidf retrieve ...
-----
tfidf      MRR@10      0.334598
tfidf      MAP@10      0.205493
tfidf      RECALL@10    0.123421
-----
```

Submission Details

- Due: 23:59, Tuesday, 25 October 2022
- What to turn in:

Electrical submission: compress all the necessary fiels and data into a zip file, and submit it via the WM5 website.

- Late policy:

In general, late homework may receive fewer points than incomplete homework. The penalty for late homework is about 20 points per day. Please DO comment and format your codes to avoid any penalty imposed by the grader.