FTEC4005 Course Project Task 1: Find Similar Articles with LSH

1. Background

We discussed searching for "similar" items in class. This kind of problem exists widely in real life, such as checking mirror pages on the Internet, and checking duplicate documents, and can also be used to quickly find similar users or similar items in the business system.

In this task, given a data set containing text, LSH for cosine distance is required to be applied to quickly find **all similar text pairs** within it. **A similar pair** is defined as two text strings with a **Cosine distance greater than 0.8** after word-based.

For example, given two strings "Tom eats an apple today." and "Tom eats an orange today." (without quotes). The Cosine distance of the two sets is 0.8.

2. Data Set Information

This data set contains one file:

1. all_articles.txt
    - There are 10000 articles in this dataset. Each line of the dataset is an article.
    - Each line begins with t_id (note that the t_id is not in order), followed by the specific content of the text, Format is as follows (<article_content> is a string, and you can look at the data set for more information):

```
t132 <article_content>
t212 <article_content>
...
```

2. test_articles.txt
    - This is a small data set for you to debug. There are 100 articles in this test dataset. Each line of the dataset is an article. In test_ans.txt, we show which pairs on the test dataset are similar. You can refine your code on small datasets and then run it on all article datasets.
    - This file has the same format as all_articles.txt.
3. test_ans.txt
    - This shows which articles are similar to the test article dataset. The Format is as follows:

```
t1088 t5015
t1297 t4638
t1768 t5248
...
```

3. Goal

You should implement an algorithm to find all similar articles in all_articles.txt. Save **all similar pairs** you find as **result.txt** (format is the same as test_ans.txt). These results need not be in order.

In your final report, you should **describe your method in detail** and show the following points:

- The **running time** of your algorithm (report the **time spent processing the data** and the **time spent searching**, respectively).
- The score of this task will depend on the time overhead of your method and the precision of the pairs you found.
- You **must** use an LSH method to solve the problem. (refer to page 21 of FTEC4005-Lecs3-4-LSH)
- We will run your method in our machine. Please provide a detailed readme to reproduce your experimental result.

PS: Remember to upload your **code** (with the necessary comments). We'll check and run your code (make sure your code runs and outputs the run time and results).