IR Final Project: GossipIR

Jen-Chieh,Yang
B05902134
signature
楊仁傑
contribution:Data preparation, Inverted File construction,VSM Design, Recommendation System Design
b05902134@ntu.edu.tw

Zi-Yuan, Huang
B05902050
signature
黃子源
contribution: Front end interface, UI optimization, Model output and interface connection
b05902050@ntu.edu.tw

Sung-Ping,Chang
B05902010
signature
張頌平
contribution:Data preparation, Inverted File construction,VSM Design, Recommendation System Design
b05902010@ntu.edu.tw

Yu-Chun,Chen
B05902116
signature
陳昱鈞
contribution: Front end interface, UI optimization, Model output and interface connection
b05902116@ntu.edu.tw

1 INTRODUCTION

Everyone loves gossip. However, the information in PTT Gossiping is quite unstructured and messy. In order to fulfill our needs, we want to build a retrieval system for news in PTT Gossiping by IR models we learned in class. This retrieval system is able to search what we want to know fast and accurate.

2 RELATED WORK

Our reference model is what we implemented in HW1. The difference between these two system is that GossipIr not only aim on precision but also on efficiency as well. The similarity between these models is we construct the model based on vector space model(VSM). In order to success our goal, we changed some details of the original model and implemented some technique to make the system more user friendly and similar to searching engine. We also try to label some relevance term for some topic to implement query expansion to make the system able to retrieve documents more suit for recent news.
We also reference some mainstream recommendation such as recommendation in Netfix to design the user interface.
While implementing system, we also construct the inverted file on our dataset. We defined some criteria to drop some useless data because those data might hurt the performance while we implementing some technique such as normalization.

3 METHODOLOGY

We used django to demonstrate our project in a user friendly interface. We will introduce our IR methodology first and then our django data structure.

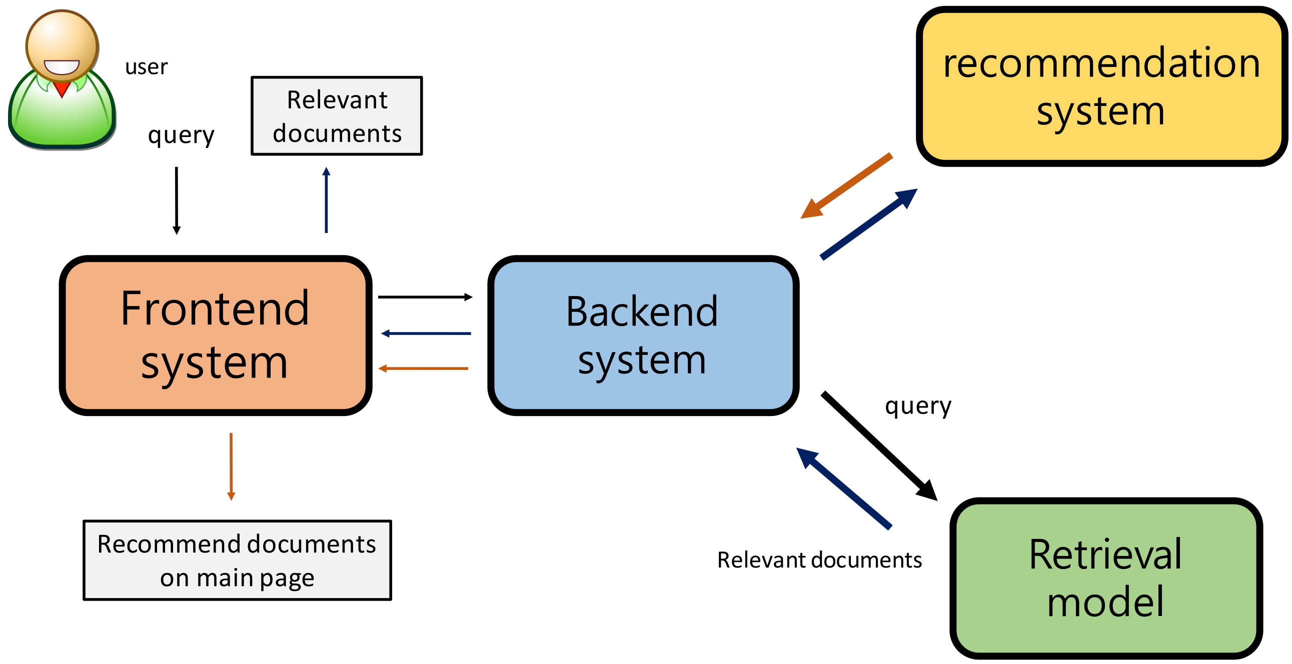

Figure 1. The illustration of our proposed methodology.

As for retrieval system, we implement a VSM(vector space model). We constructed document vectors for articles we clawed from PTT. Using experience in HW1 and combining with some techniques, we try to enhance the performance for the system. Because of the fact that our purpose is to design a retrieval system more convenient for users, we have to consider not only precision of the outcome, but also the efficiency. As a result, we shorten the run-time but still maintain the performance of the model.

3.1 Retrieval model

Our dataset (PTT gossiping articles), which is known for containing lots of so called ”meaningless” articles, is hard to maintain the performance, that is, hard for model to retrieve relevance documents that containing the exact articles. Another topic we aim to solve is the efficiency for retrieval process. Thus, we removed the articles that lack of information. Moreover, we discarded some low frequency terms.After those data pre-processing, we construct inverted file on document for implement the retrieval system. we implement VSM model with cosine-similarity as the measure to select documents. We also obtain okapi-25 for term frequency, document normalization. Due to the fact our system runs as client sever system, we also take seriously consideration on time limit. We implemented clustering method that could speed up the retrieval process. Splitting document into several cluster, the system could simply reduce the number of calculating cosine similarity by only choose the cluster that have highest similarity value of centroid among all cluster as candidates of the final result. We also try to defined some relevant terms for some queries for query expansion by human labeling, our concept is that users are more interested in current information than which in the past. Therefore, our criteria during labeling was based on popular searching terms on PTT.

3.2 recommendation

Assume our sever have collect enough information after running for a period of time, we could construct the data of the format as HW2 like user and document each user query for. As a result, we could implement recommendation by matrix factorization and BPR model easily by transform data we collected to the format as HW2. However, we can’t collect user preference of document in this scenario since limited time and lack of users. Thus, we implemented our recommendation system through rule-based method by user’s query history, which is similar to searching history like other searching engine. After we collect enough history in the database, we could connect the model to our system and switch our recommend strategy to BPR model.

3.3 Backend System

In our backend system, we design 3 data sheet to record user behaviors and document features, which is user, document,

Query topic	柯文哲
VSM(1st result)	Title: [新聞] 賴清德：我比蔡英文、柯文哲更適合當總統
+ okapi/bm25	Title: [新聞] 深夜才談六四被網友酸爆柯: 幕僚叫我寫的
+ K-means clustering	Title: [新聞] 柯文哲：兩岸問題不回答就是最好回答
+ Both	Title: [新聞] 柯文哲：守護台灣是每個人的責任，就算乞丐也要守護台灣
Query topic	籃球
VSM(1st result)	Title: [問卦] AND1?
+ okapi/bm25	Title: [問卦] AND1?
+ K-means clustering	Title: [問卦] 急 玩躲避球推薦戰術或陣形?
+ Both	Title: [問卦] 今年籃球冠軍會是林家拿下的嗎??
Query topic	館長
VSM(1st result)	Title: [新聞] 中職》兄弟水祭 3 連戰蔡阿嘎、舒子晨、館長接力開球
+ okapi/bm25	Title: [新聞] 女友狂誇館長很帥很壯童仲彥吃醋想...
+ K-means clustering	Title: [新聞] 苗縣議員鄭聚然嗆館長給臭錢
+ Both	Title: [新聞] 女友狂誇館長很帥很壯童仲彥吃醋想..
Query topic	台灣大學生
VSM(1st result)	Title: Re: [新聞] 總統：六四與美麗島是關鍵盼助中國邁向民主
+ okapi/bm25	Title: Re: [問卦] 64 到底關我們台灣人屁事啊?
+ K-means clustering	Title: Re: [問卦] 六四到底死了多少人有正確統計呢?
+ Both	Title: [新聞] 大學生嘆亡國感，陳建仁：要使台灣亡國，除非......

Table 1. Qualitative analysis for improvements of vector space model.

recommend_record respectively. data sheet user records the account and password of an user, and document records many attributes such as topic, context, tf, idf, etc. Recommend_record is a data sheet with 2 foreign keys point to user and document, means that the document is recommended to the user.

When the user login, the system will retrieve the recommended documents from database and show it on the personalized search page. The usage of database can save time calculating recommend document, since it will use thread to simultaneously calculate when user search queries.

3.4 Frontend System

The frontend system includes 5 pages: home, login, search, recommendation, document context. When user login, the website will be redirected to search page, and user can search for gossips from this page. This page will also show the recommended document for specific user. We have spent efforts to make our frontend looks user friendly.

4 EXPERIMENTS

Here are some of the settings and hyper-parameters we used to optimize our vector space model

4.1 Terms choosing criteria

We set the maximum dimension of our document vector as 25000 to avoid memory error. Therefore, we dropped the terms which appears less than 5 to 20 times and tested which settings will be better. Finally, we decided to drop the terms which appears less than 10 times and make our model can run efficiently.

4.2 k-means clustering

The search engine needs to interact with users quickly or the user may lose patience. Without clustering, it may cost up to 80 seconds to find the ten most relevant documents. It is too long for even the most patient person to wait a searching result. To diminish the searching time to 5 seconds at most, we use k-means clustering to conduct some experiments by clustering total 20000 documents from 10 to 30 clusters and discussed which setting is the best. At last, we clustered all documents to 15 groups and the range of document number in 15 clusters is from 300 to 3000. In our final demo, each searching just cost average 2.5 seconds to get the top 10 results.

4.3 okapi-25 parameters selection

We tried parameter k1 from 1.5 to 1.8 and parameter b from 0.6 to 0.8. We choose k1 as 1.5 and b as 0.75 eventually.

5 RESULTS

As shown in the Table 1,we can see that the original vector space model can already found relevant document, in some cases, okapi/bm25 helped us to find more diverse relevant documents than original vsm just like example 3 about『館長』. K-means clustering helped us to decrease lots of time of searching, but it is also possible that the documents in specific cluster doesn't have totally relevant documents just like example 2 about『籃球』, the top search result shows the article is more corresponding to ”dodge ball” than ”basketball”. To summarize, we can say that add the two methods: okapi/bm25 normalization, K-means clustering in our vector space model indeed improve our final results, not only got more relevant articles, but also accelerated the searching speed significantly.

6 CONCLUSIONS

This was the second time for as to implement a retrieval model through VSM and we feel it quite different with HW1. The main challenge was that articles on PTT gossiping board are messy and often contain useless information. In order to deal with such problem and enhance the retrieval efficiency and performance in the same time, we proposed the model combined with clustering method that could speed up the whole retrieval model.

Our project design as a searching system with login system, this setting could let us easily collect the information such as searching history and enhance performance.

In future work, we could obtain more information of the articles such as push content, time stamp...... etc. and obtain with original document vector to construct new vectors, and try to implement on neural network witch could combine these knowledge of different domain knowledge, expected to enhance performance. However, we should also cope with the time consuming of the neural network because it might slower our runtime.

7 REFERENCES

Generalized vector spaces model in information retrieval
SIGIR ’85: Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrievalJune 1985 Pages 18–25
https://doi.org/10.1145/253495.253506

Research on Decision Strategy Algorithms of Web Crawler Vector Space Model Based on Schmidt Orthogonal Optimization
2019 International Conference on Computer Information Analytics and Intelligent Systems (CIAIS 2019)

Query expansion techniques for information retrieval: A survey
Information Processing & Management Volume 56, Issue 5, September 2019, Pages 1698-1735

Document Expansion by Query Prediction