ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

□□□□□

BÁO CÁO MÔN HỌC
ĐỀ TÀI: XÂY DỰNG MÔ HÌNH TRUY XUẤT THÔNG TIN TRÊN BỘ DỮ LIỆU CRANFIELD

GVHD: Th.S. Nguyễn Trọng Chỉnh

Sinh viên thực hiện:		
STT	Họ tên	MSSV
1	Dương Văn Bình	18520505
2	Hà Như Chiến	18520527

TP. HỒ CHÍ MINH – 12/2020

1. GIỚI THIỆU

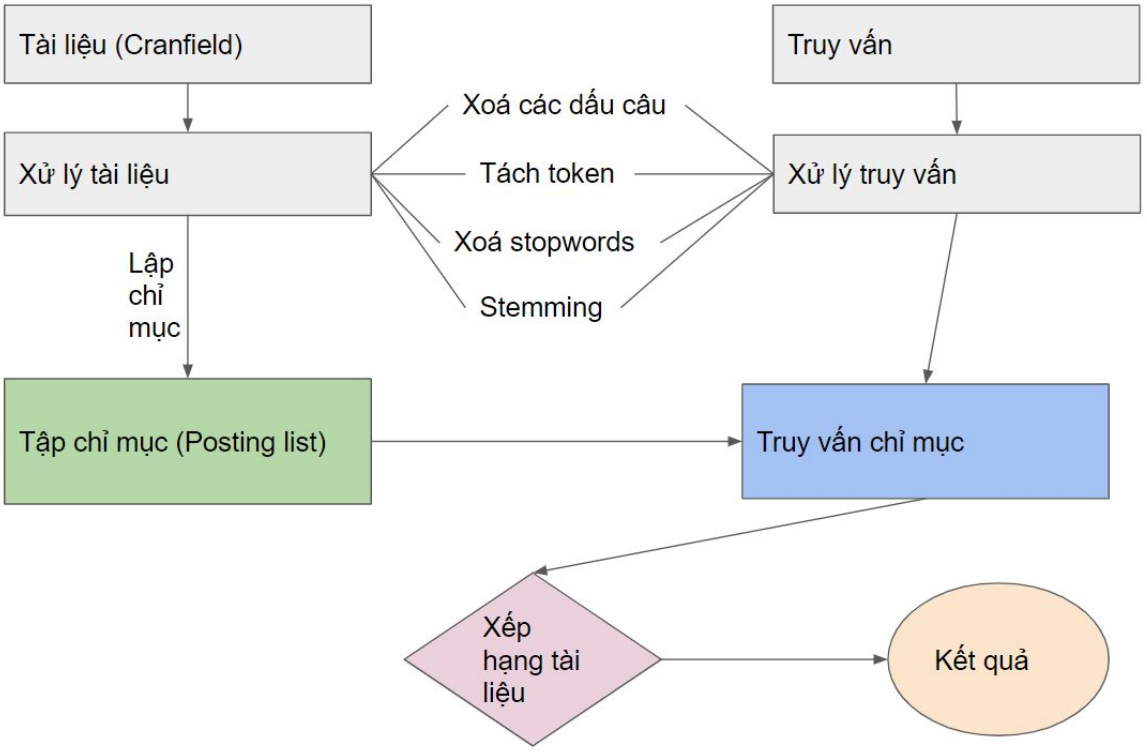
Trong phạm vi đồ án môn học, nhóm sẽ xây dựng mô hình truy xuất thông tin dựa trên mô hình Vector Space trên bộ dữ liệu Cranfield.

Mô hình sẽ được sử dụng để đánh giá xếp hạng các tài liệu liên quan trong bộ dữ liệu với các truy vấn.

Mô hình Vector Space có khả năng xếp hạng tài liệu theo độ liên quan, được sử dụng rộng rãi trong truy xuất thông tin, dựa vào cơ sở toán học nên dễ tính toán các độ đo tương đồng.

2. NỘI DUNG

Quy trình thực hiện:



2.1. Hệ thống mô hình Vector Space

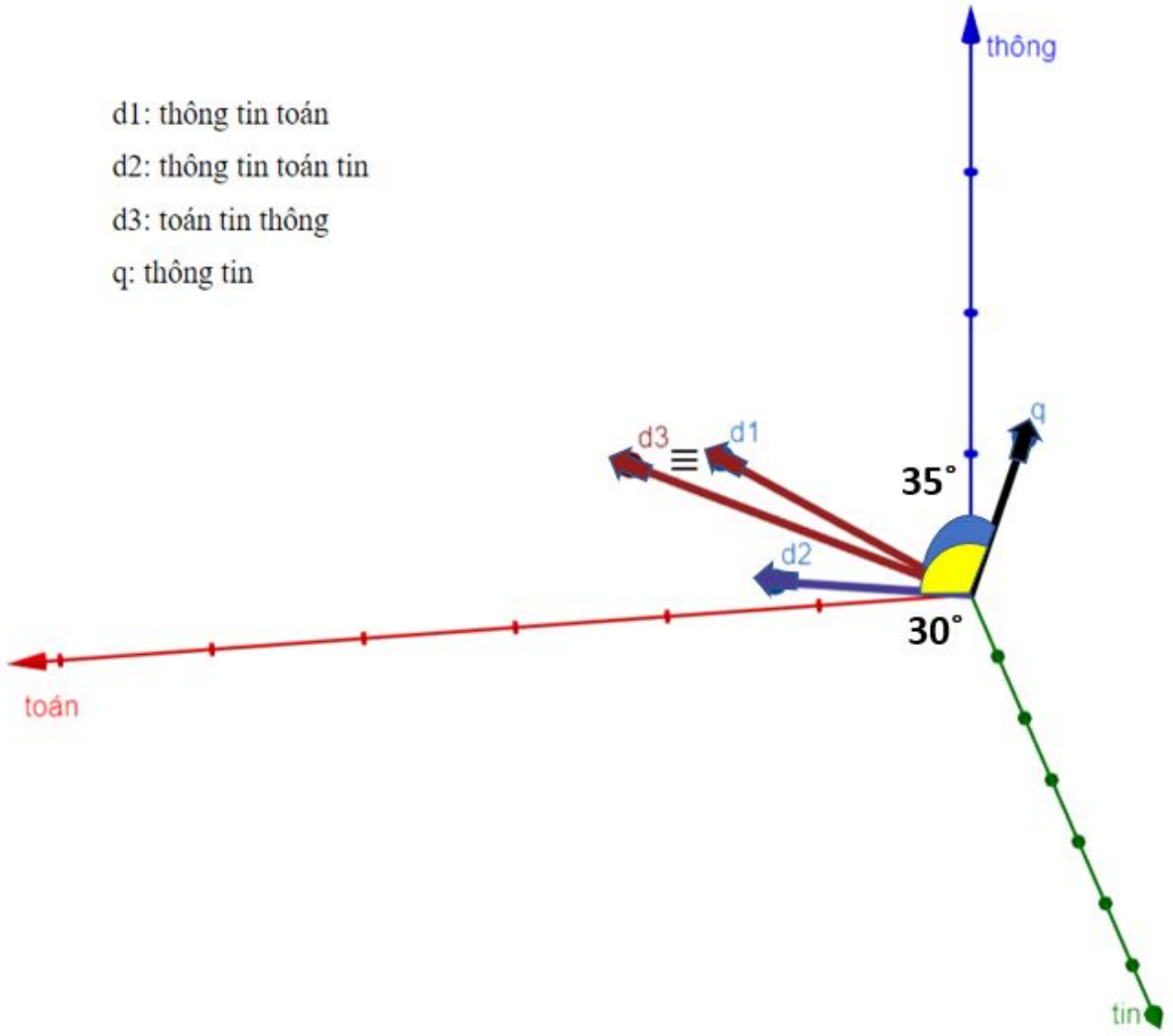
Biểu diễn văn bản và câu truy vấn bằng một vector đa chiều: $vector(d_i) = \{ w_{i,1}, w_{i,2}, ..., w_{i,3} \}$ với $w_{i,j}$ là trọng số của term j trong tài liệu thứ i.

Mỗi chiều của không gian là một từ ngữ được chọn để biểu diễn tài liệu truy vấn (Bag of words).

Nhờ vào những đặc điểm trên mô hình Vector Space đã chuyển các term từ ký tự chuỗi thành dạng số học theo nhưng vẫn giữ được độ quan trọng của từng chuỗi với mỗi tài liệu.

Độ liên quan giữa 2 vector tài liệu d và vector truy vấn q được tính toán thông qua độ đo khoảng cách giữa các vector.

$$sim(d_i,\ q) = cos = \frac{d_i * q}{|d_i||q|}$$



Như kết quả trong hình bên trên thì truy vấn q khớp với tài liệu d2 nhất vì góc cosine giữa chúng là nhỏ nhất.

Việc dựa vào cơ sở toán học, chỉ quan tâm đến các kết quả tính toán và biểu diễn từ theo dạng Bag of words làm cho mô hình có một nhược điểm là không có hiểu được ngữ nghĩa trong câu. Như 2 câu: ‘thông tin toán’ và ‘toán tin thông’ có nghĩa hoàn

toàn khác nhau nhưng mô hình vẫn hiểu chúng là một vì số lần xuất hiện của các term trong 2 câu là như nhau.

2.2. Bộ dữ liệu

Tên bộ dữ liệu: Cranfield.

Số lượng điểm dữ liệu: 1400 tài liệu là các phần abstract của các bài báo.

Định dạng lưu trữ: <số thứ tự tài liệu>.txt. (1.txt, 1000.txt, ...).

Lưu trữ dữ liệu: Vì dữ liệu gồm rất nhiều tập tin dẫn đến việc đọc dữ liệu sẽ mất nhiều thời gian và việc tên tập tin cũng là số thứ tự của các tài liệu vì vậy nếu không chú ý cách đọc sẽ dẫn đến nhầm lẫn số thứ tự của tài liệu. Do đó, nhóm quyết định lưu trữ lại tài liệu vào dataframe gồm một cột là *'text'* gồm 1400 dòng tương ứng với nội dung của các tài liệu từ 1.txt tới 1400.txt. Sau đó dataframe này sẽ được lưu lại thành *'Cranfield.csv'* phục vụ cho các bước tiếp theo.

Lý do thực hiện việc lưu trữ lại dữ liệu là vì khi đọc một tập tin *name.csv* chỉ 1400 dòng thì sẽ nhanh hơn rất nhiều so với đọc 1400 tập tin *.txt.*

2.3. Tiền xử lý bộ dữ liệu

Loại bỏ các dấu câu, số và chuyển tài liệu thành toàn bộ chữ viết thường: vì mục tiêu của ta là truy xuất thông tin dựa vào nội dung của tài liệu nên các dấu câu sẽ không có nhiều tác dụng và các chữ giống nhau khi viết hoa sẽ bị nhận là các chữ khác nhau nên ta sẽ chuyển về chữ viết thường cho toàn bộ tài liệu đồng thời nếu để nguyên chúng sẽ làm tốn tài nguyên lưu trữ nên ta sẽ phải xử lý.

Tách token (tokenize): sau khi đã có dữ liệu sạch ta sẽ tiến hành việc tách các token, mục đích của việc này là giúp việc tính toán các trọng số trong bước lập chỉ mục tài liệu và các bước stemming, xoá bỏ stopwords được thực hiện dễ dàng.

Chiều dài bộ từ vựng

- trước khi loại bỏ dấu câu, số và chuyển chữ viết thường: 8360
- sau trước khi loại bỏ dấu câu và chuyển chữ viết thường: 7229

Loại bỏ stopwords: vì đây là bộ dữ liệu Tiếng Anh nên ta sẽ dùng bộ stopwords đã được xây dựng sẵn cho ngôn ngữ này.

Trong Tiếng Anh các từ như: 'and', 'is', 'the', ... thường xuất hiện với tần suất rất lớn trong một dữ liệu vì vậy chúng không chỉ ảnh hưởng tới tài nguyên lưu trữ mà còn tới việc tính toán các trọng số cho các từ này là không có nhiều ý nghĩa vì gần như tài liệu

nào cũng có sự xuất hiện của chúng.

Chiều dài bộ từ vựng

- trước khi loại bỏ stopwords: 7229
- sau khi loại bỏ stopwords: 7110

Kết quả cho thấy việc loại bỏ stopword giúp giảm đáng kể không gian lưu trữ và các chi phí tính toán sau này.

Stemming: Trong Tiếng Anh có những quy tắc ngữ pháp đặc trưng riêng dẫn đến việc một từ có thể có nhiều cách biểu diễn khác nhau (ví dụ: come, coming, comes, ...) vì vậy stemming là phương pháp đưa các từ có biến thể về chung một dạng của ban đầu của chúng.

Nhóm có thử nghiệm hai phương pháp PorterStemmer và WordNetLemmatizer.

Chiều dài bộ từ vựng

- trước khi stemming: 7110
- sau khi dùng PorterStemmer: 4410
- sau khi dùng WordNetLemmatizer: 6268

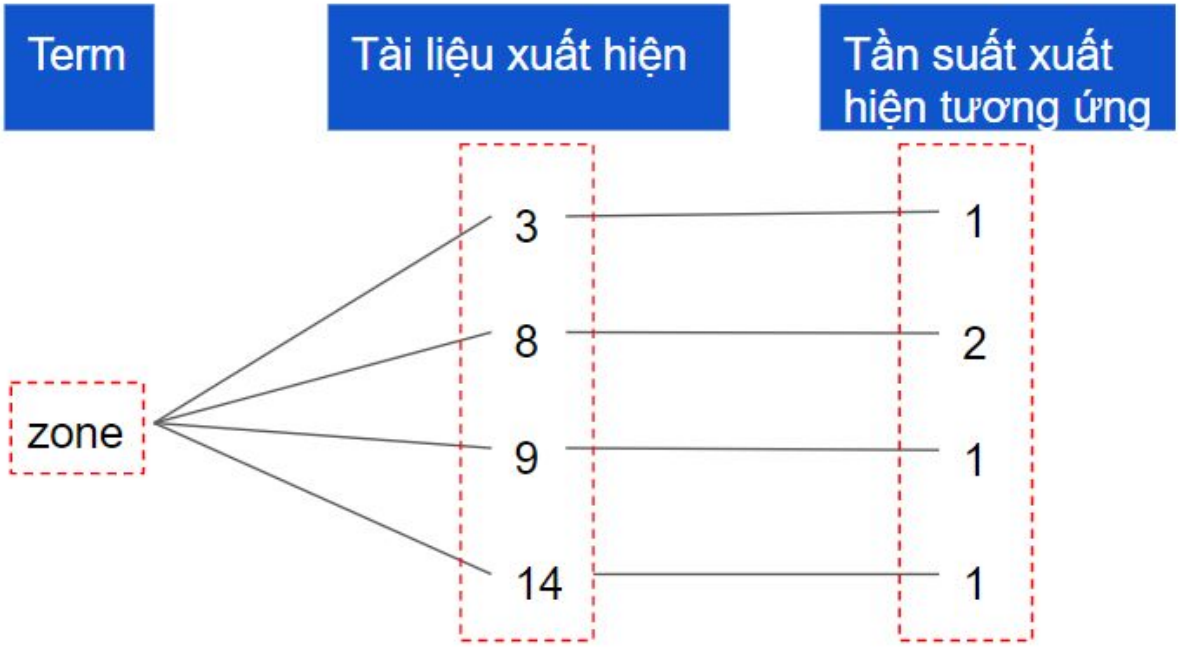
Kết quả đạt được khi sử dụng PorterStemmer có chiều dài của tập từ vựng đã giảm đi nhiều hơn so với WordNetLemmatizer vì vậy trong đồ án này nhóm sẽ sử dụng PorterStemmer là thư viện stemming chính.

2.4. Tạo tập chỉ mục (Posting list)

Tạo tập từ vựng: Từ tập các token của các tài liệu ta sẽ tiến hành lập ra bộ từ vựng của toàn bộ tập dữ liệu.

Tập từ vựng này sẽ gồm 4410 từ khác nhau, mỗi từ sẽ xuất hiện trong các tập tài liệu với tần suất khác nhau.

Tính toán tần suất xuất hiện của mỗi từ vựng (term): Ở bước này ta cần nhận được kết quả là thông tin như tài liệu xuất hiện và tần số xuất hiện tương ứng với mỗi term.



Tổng số lần xuất hiện (tf) của term trong bộ dữ liệu chính là tổng của cột Tần suất xuất hiện tương ứng.

Tính toán idf cho term

Ta sử dụng công thức $idf_{term} = log(\frac{N}{N_{term}})$ trong đó N là tổng số tài liệu trong tập dữ liệu (1400), N_{term} là tổng số tài liệu chứ term.

Hàm logarithm ở đây là cơ số 10.

Ta có ví dụ: idf(zone) = log(1400/4) = 2.54

Các trường hợp giá trị phân số bên trong hàm log làm cho idf không tính được hoặc bằng 0:

- tử số bằng 0.
- mẫu số và tử số bằng nhau.

Để giải quyết các vấn đề này ta sẽ cộng thêm vào tử của phân số với 1, và cộng thêm vào mẫu giá trị rất nhỏ 0.0001: $idf_{term} = log(\frac{N+1}{N_{term}+0.0001})$.

Như vậy giá trị idf(zone) = log(1401/4.0001) = 2.54, không có thay đổi so với công thức ban đầu.

Tính toán trọng số cho term (weighting)

Ta có công thức sau: $w_{doc\_id,\ term} = tf_{doc\_id,\ term} * idf_{term})$ trọng số w này được tính cho mỗi term tương ứng với mỗi tài liệu mà term này xuất hiện.

Ví dụ: tính trọng số cho zone với tài liệu số 1: $w_{1,\,zone} = 1 * 2.54 = 2.54$

tính trọng số cho zone với tài liệu số 8: $w_{1,\,zone} = 2 * 2.54 = 5.08$

Chuẩn hoá các trọng số

Ta có thể nhận thấy từ 'zone' có trọng số cao hơn trong tài liệu 8 như vậy có thể kết luận tài liệu số 8 sẽ phù hợp hơn tài liệu 1 với truy vấn là từ 'zone'. Khẳng định trên có thể đúng nếu chiều dài của hai tài liệu này là bằng nhau.

Để giải quyết trường hợp độ dài của các tài liệu làm ảnh hưởng tới việc tính toán các trọng số ta sẽ tiến hành chuẩn hóa các trọng số.

Công thức: $w_{doc\_id,\,term} = \frac{w_{doc\_id,\,term}}{norm}$

với norm = sqrt($w_{doc\_id,\,term\_1}{}^2 + w_{doc\_id,\,term\_2}{}^2 + ... + w_{doc\_id,\,term\_N}{}^2$) tức là căn bậc 2 của tổng các trọng số của từng term trong một tài liệu bình phương.

Có rất nhiều phương pháp chuẩn hóa trọng số, nhưng nhóm chọn công thức trên là vì độ đo đánh giá độ tương đồng giữa tài liệu và truy vấn là độ đo cosine nên việc chọn norm như trên sẽ giảm bớt một bước tính toán khi tính độ tương đồng.

Kết quả xử lý trên tập tài liệu Cranfield:

	terms	doc_id	freq_in_doc	idf	w	normed_w
0	ab	[744, 924]	[1, 1]	2.85	[2.85, 2.85]	[0.18, 0.13]
1	abbrevi	[122]	[1]	3.15	[3.15]	[0.16]
2	abil	[51, 77, 738]	[1, 1, 1]	2.67	[2.67, 2.67, 2.67]	[0.11, 0.08, 0.12]
3	abl	[99, 132, 536, 581, 695, 763, 908, 914, 986, 1...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 2]	2.15	[2.15, 2.15, 2.15, 2.15, 2.15, 2.15, 2.15, 2.1...	[0.08, 0.07, 0.07, 0.19, 0.05, 0.2, 0.08, 0.09...
4	ablat	[82, 274, 553, 587, 1065, 1096, 1097, 1098, 10...	[2, 2, 7, 1, 3, 2, 7, 3, 4, 4, 3, 3, 3, 1]	2.00	[4.0, 4.0, 14.0, 2.0, 6.0, 4.0, 14.0, 6.0, 8.0...	[0.09, 0.14, 0.68, 0.16, 0.38, 0.25, 0.51, 0.2...
...	...	...	...	...	...	...
4405	zhukhovitskii	[270]	[1]	3.15	[3.15]	[0.12]
4406	zone	[14, 126, 167, 218, 243, 455, 828, 960, 1072, ...	[2, 2, 1, 1, 3, 1, 1, 2, 3, 1, 1, 1, 1]	2.03	[4.06, 4.06, 2.03, 2.03, 6.09, 2.03, 2.03, 4.0...	[0.15, 0.32, 0.17, 0.07, 0.46, 0.09, 0.1, 0.27...
4407	zoom	[374]	[2]	3.15	[6.3]	[0.38]
4408	zuk	[890]	[1]	3.15	[3.15]	[0.18]
4409	zurich	[792, 1137]	[1, 1]	2.85	[2.85, 2.85]	[0.1, 0.13]

4410 rows × 6 columns

2.5. Xử lý truy vấn

Các truy vấn sẽ được thực hiện tiền xử lý như với tập tài liệu.

Tính trọng số cho các term của câu truy vấn và chuẩn hóa trọng số.

Tính toán độ tương đồng với tài liệu.

Tổng hợp độ tương đồng.

Sắp xếp độ tương đồng giảm dần tương ứng với tài liệu và đưa ra danh sách các tài liệu liên quan.

Ví dụ với query: 'a b a a'

- Qua bước tiền xử lý ta có token: [a, b].
- Với tần suất xuất hiện sẽ được tính trên query: tf(a) = 3, tf(b) = 1.
- Idf của a và b sẽ được lấy ra từ phần tạo tập chỉ mục của tập tài liệu.
- Tính trọng số cho query:
  - $w^a_{query} = tf(a) * idf(a)$
  - $w^b_{query} = tf(b) * idf(b)$
- Chuẩn hóa trọng số:
  - $Norm_{query} = \sqrt{(w^a_{query})^2 + (w^b_{query})^2}$
  - $w^a_{query} = \frac{w^a_{query}}{Norm_{query}}$, $w^b_{query} = \frac{w^b_{query}}{Norm_{query}}$
- Từ posting list ta có posting list của:
  - a: $[id_1, id_2, id_3, \dots][w_1, w_2, w_3, \dots]$
  - b: $[id_1, id_2, id_3, \dots][w_1, w_2, w_3, \dots]$
- Nhân các trọng số của query cho danh sách các trọng số của posting list tương ứng:
  - a: $[id_1, id_2, id_3, \dots][w_1, w_2, w_3, \dots] * w^a_{query}$
  - b: $[id_1, id_2, id_3, \dots][w_1, w_2, w_3, \dots] * w^b_{query}$

Việc thực hiện phép nhân này chính là ta đang thực hiện nhân các phần tử của 2 vector cho nhau (tử số của phép tính độ tương đồng) và ta không cần chia cho tích độ dài 2 vector vì ở bước trên độ dài đã được chuẩn hóa về 1.

- Giả sử sau các bước trên ta có bảng trọng số cho của tài liệu và query như sau:

	a	b
id_1	$w_1 * w^a_{query} = w^a_1$	$w_1 * w^b_{query} = \mathbf{0}$
id_2	$w_2 * w^a_{query} = \mathbf{0}$	$w_2 * w^b_{query} = w^b_2$
id_3	$w_3 * w^a_{query} = w^a_3$	$w_3 * w^b_{query} = w^b_3$

- Tổng hợp độ tương đồng và đưa ra kết quả:
  - a: $id_1\ (w_1^a)$, $id_3\ (w_3^a)$
  - b: $id_2\ (w_2^b)$, $id_3\ (w_3^b)$

⇨ $id_1\ (w_1^a)$, $id_2\ (w_2^b)$, $id_3\ (w_3^a + w_3^b)$ ta sẽ sắp xếp danh sách này theo trọng số w theo thứ tự giảm dần từ đó các id tương ứng chính là các tài liệu liên quan nhất với query.

2.6. Đánh giá kết quả mô hình

Để đánh giá mô hình, độ đo MAP (Mean Average Precision) được sử dụng.

Với đáp án truy xuất từ bộ câu hỏi truy vấn dùng cho bộ dữ liệu Cranfield kết quả được trả về từ mô hình sẽ được lấy 35 tài liệu liên quan nhất, vì trong tập tài liệu kết quả dành cho các câu truy vấn số lượng tài liệu liên quan cao nhất cho một câu truy vấn là 33 (của câu truy vấn số 23).

```
6       what theoretical and experimental guides do we have as to turbulent couette flow behaviour .
7       is it possible to relate the available pressure distributions for an ogive forebody at zero angle of attack to the lower surface
pressures of an equivalent ogive forebody at angle of attack .
8       what methods -dash exact or approximate -dash are presently available for predicting body pressures at angle of attack.
9       papers on internal /slip flow/ heat transfer studies .
10      are real-gas transport properties for air available over a wide range of enthalpies and densities .
11      is it possible to find an analytical,  similar solution of the strong blast wave problem in the newtonian approximation .
12      how can the aerodynamic performance of channel flow ground effect machines be calculated .
13      what is the basic mechanism of the transonic aileron buzz .
14      papers on shock-sound wave interaction .
15      material properties of photoelastic materials .
16      can the transverse potential flow about a body of revolution be calculated efficiently by an electronic computer .
17      can the three-dimensional problem of a transverse potential flow about a body of revolution be reduced to a two-dimensional
problem .
18      are experimental pressure distributions on bodies of revolution at angle of attack available .
19      does there exist a good basic treatment of the dynamics of re-entry combining consideration of realistic effects with relative
simplicity of results .
20      has anyone formally determined the influence of joule heating,  produced by the induced current,  in magnetohydrodynamic free
convection flows under general conditions .
21      why does the compressibility transformation fail to correlate the high speed data for helium and air .
22      did anyone else discover that the turbulent skin friction is not over sensitive to the nature of the variation of the viscosity
with temperature .
23      what progress has been made in research on unsteady aerodynamics .
24      what are the factors which influence the time required to invert large structural matrices .
25      does a practical flow follow the theoretical concepts for the interaction between adjacent blade rows of a supersonic cascade .
```

Việc lựa chọn số lượng tài liệu trả về cho mô hình như vậy là hợp lý vì sẽ đảm bảo trả về tài liệu phù hợp nhất cho từng câu truy vấn, đáp ứng việc tính toán MAP cho toàn bộ câu truy vấn và giảm tài nguyên lưu trữ chỉ số của những tài liệu không cần thật sự liên quan.

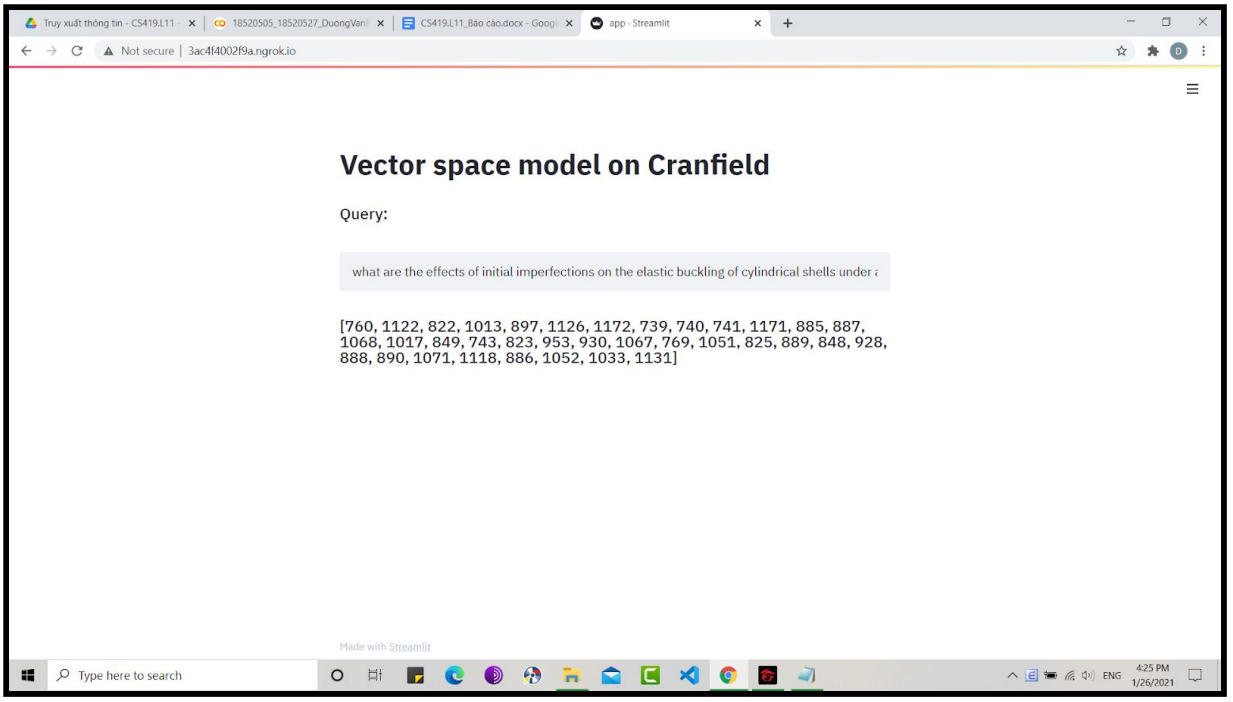
Kết quả MAP đạt được của mô hình với 100 câu truy vấn là 31.

3. KẾT LUẬN

Kết quả đạt được trong việc xây dựng mô hình Vector Space trong việc truy xuất thông tin trên bộ dữ liệu Cranfield đã đạt được những kết quả cơ bản của một mô hình truy xuất thông tin.

Trong đồ án này, những kiến thức về truy xuất thông tin đã được áp dụng để đáp ứng việc xây dựng mô hình truy xuất thông tin. Đồng thời một mô hình demo cho việc truy xuất thông tin cũng được xây dựng để trực quan việc truy xuất thông tin sinh động hơn.

Phụ lục



Mô hình được deploy sử dụng thư viện Streamlit và được chạy trên môi trường của Google Colab.

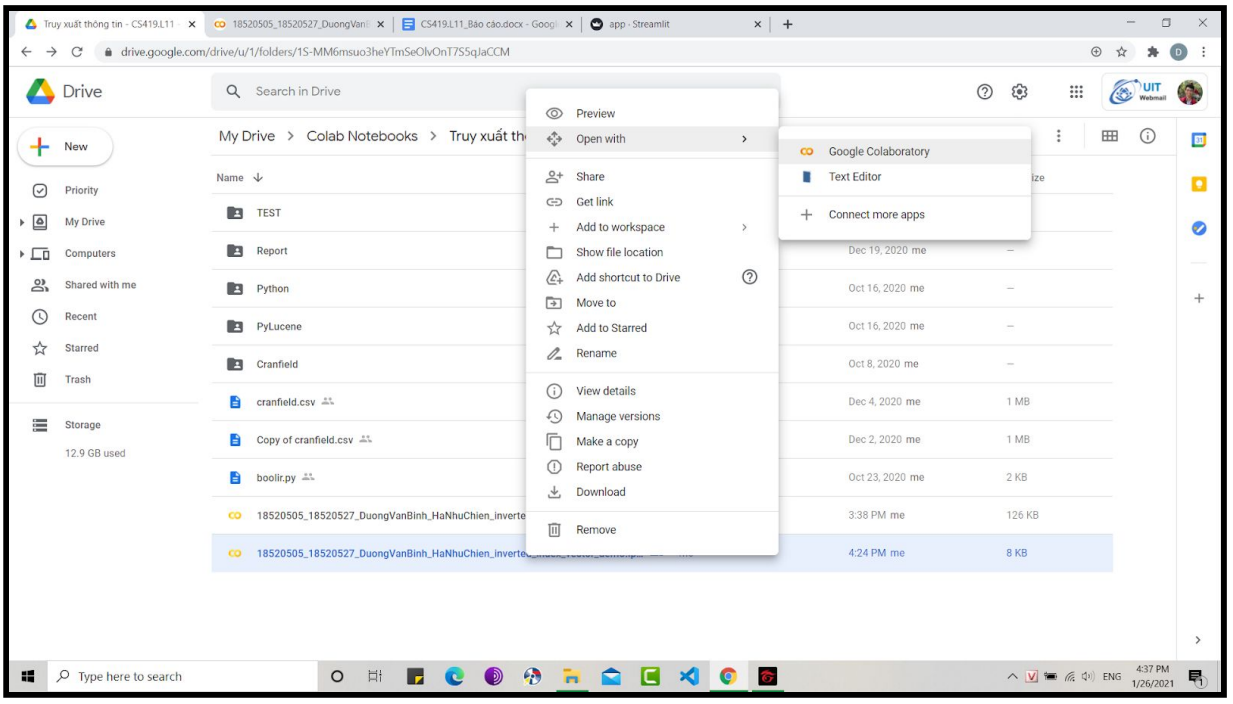
Hướng dẫn cài đặt chạy demo mô hình.

1. Cài đặt môi trường chạy:

   Để thuận tiện cho việc sử dụng Google Colab và việc load các file dữ liệu cần thiết thì toàn bộ folder báo cáo cần được đưa lên Google Drive.

2. Chạy chương trình deploy

- Sau khi folder đã được đưa lên Drive, tiến hành mở file ***18520505_18520527_DuongVanBinh_HaNhuChien_inverted_index_vector_demo.ipynb*** bằng Google Colab.



- Để chạy được chương trình cần chú ý đường dẫn của file posting list.

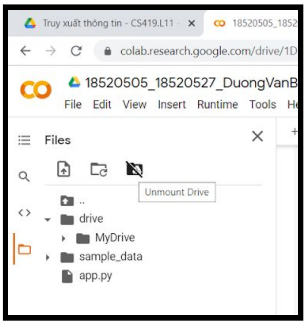
```
def main():
  st.title("Vector space model on Cranfield")
  st.subheader("Query: ")

  user_input = st.text_input("", "airplane spot for example")

  # load normed posting list  file
  pl_path = '/content/drive/MyDrive/Colab Notebooks/Truy xuất thông tin - CS419.L11/Report/norm_posting_list.plk'
  with open(pl_path, 'rb') as f:
    posting_list = joblib.load(f)

  query = [user_input]
  result = docRetrieval(query, posting_list)
  st.subheader(result)
###
if __name__ == '__main__':
  main()
```

- Mount Drive: Nhấn vào biểu tượng Mount Drive sau đó làm theo hướng dẫn của Google Colab để kết nối Drive với Colab.



Sau khi đã Mount Drive, tìm nơi lưu trữ file ***normed_posting_list.plk*** thì ta sẽ có đường dẫn để sử dụng file này.

3. Sử dụng chương trình demo

   Chạy tất cả các cell trong file demo.

   Ở cell bên dưới, sau khi đã chạy ta sẽ có link của chương trình demo như bên dưới. Chạy link này trên trình duyệt ta sẽ có giao diện chương trình demo.

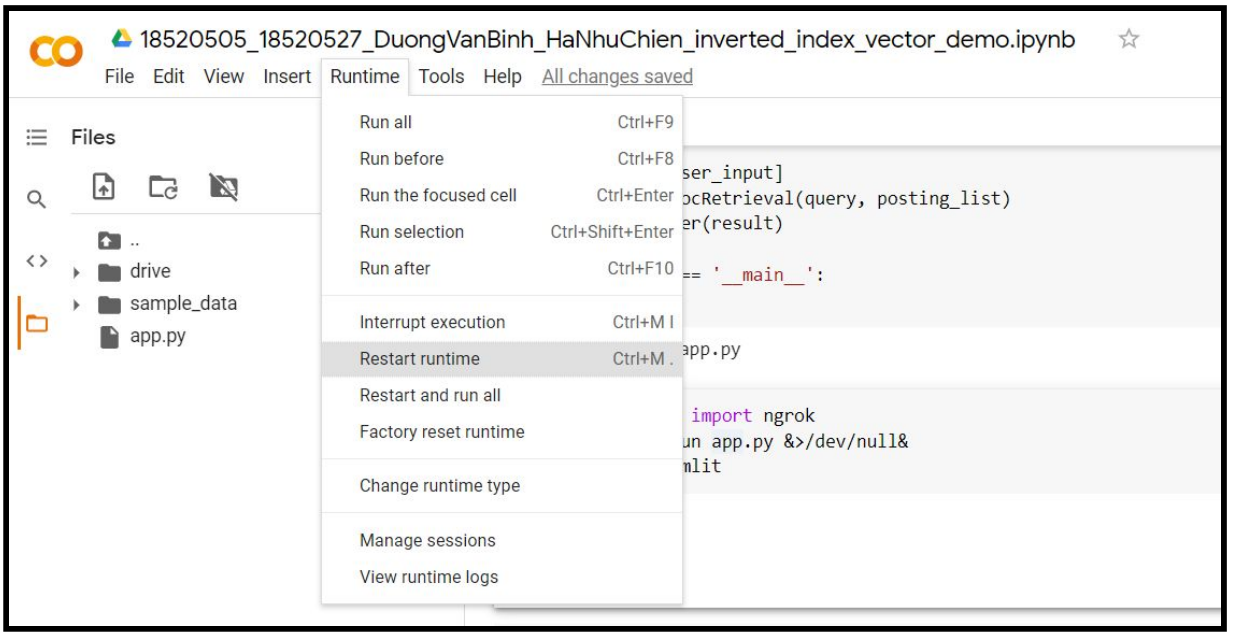
```
[4]  public_url = ngrok.connect(port='8501')
     public_url

     'http://3ac4f4002f9a.ngrok.io'
```

4. Tiến hành truy xuất

   Nhập câu truy vấn vào ô Query sau đó nhấn Enter. Kết quả trả về sẽ là một danh sách các id của tài liệu liên quan.

5. Một số lỗi có thể gặp
   - Vì chương trình chạy trực tuyến nên tốc độ Internet sẽ ảnh hưởng lớn tới việc chạy chương trình.
   - Nếu gặp lỗi trong việc lấy link chương trình demo thì có thể Restart Runtime của Google Colab rồi chạy lại các cell.



- Lỗi về đường dẫn các file: Các file cần thiết để chạy chương trình đều được gửi trong folder đồ án. Các file code đều ở dạng ***name.ipynb*** và được chạy trên Google Colab hoặc các phần mềm tương tự. Khi chạy trên Colab cần chú ý Mount Drive để có thể load các file dữ liệu cần thiết.