Web Search Engine Project

Chung-Ling Yao

Objective:

The objective of this project is to build a better search result clustering system.

The query keywords sometimes have multiple meanings, and the web search results may exist several clusters with distinct contents. The current working clustering search engine such as "yippy" have problem because the "cluster labeling" doesn't choose labels good enough to represent the clusters.

Furthermore, clustering the search results would also help users to figure out the different meaning of his/her query keywords. The user can narrow down the search result by adding more keywords corresponding to the meaning he/she is looking for.

In this project, I try to find out a better way to find the cluster labels for web search result clusters.

Architecture:

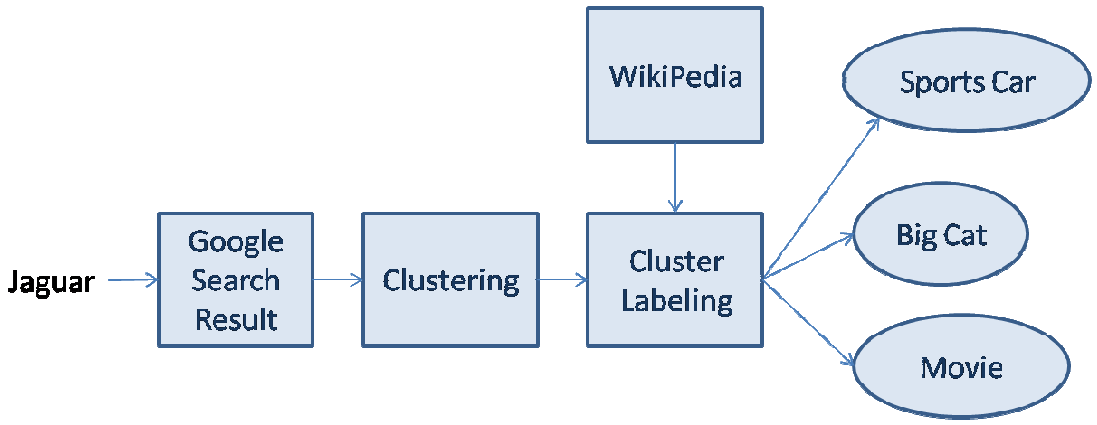


Figure 1. Data handle process

There are several problems to be solved:

1. Search index

Google is well-known for it very good indexing and ranking system. I use Google search result as my starting point. This system executes a Google query and downloads the first 20 or 40 pages from Google search result.

2. Clustering

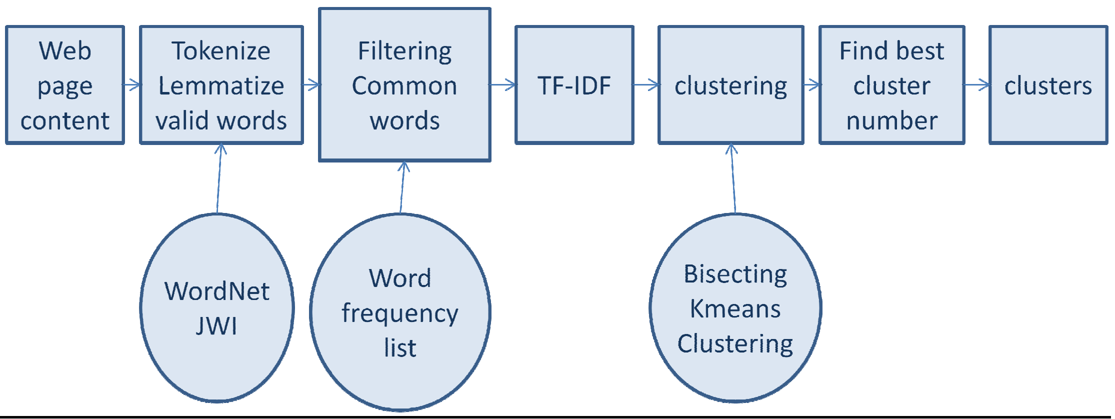


Figure 2. Process of calculating word vectors and finding clusters

According to [1], bisecting K-means may be a good algorithm to find clusters from web search result.

(1) Tokenize the web page content and build a terms vector:
I used JWI API [11] to tokenize and lemmatize the web pages by querying in wordNet [8] and to check whether this token is a valid word. Filter out all the invalid words.

(2) Remove common words. Some keywords are used too frequently to represent the topic of web pages, so I load a Map to store the k most common keywords. Then use it to remove the most common words. However, it's hard to decide the value of k.

(3) Calculate TF-IDF [12]

(4) Execute Bisecting K-Means clustering. I used S-Space API [10] to calculate the clustering.

(5) Find best cluster number: Use the jumpMethod in [13]. Calculate clustering using center from 1 to N, and find the best cluster number. N is 7 in current system setting.

3. <u>Cluster labels</u>

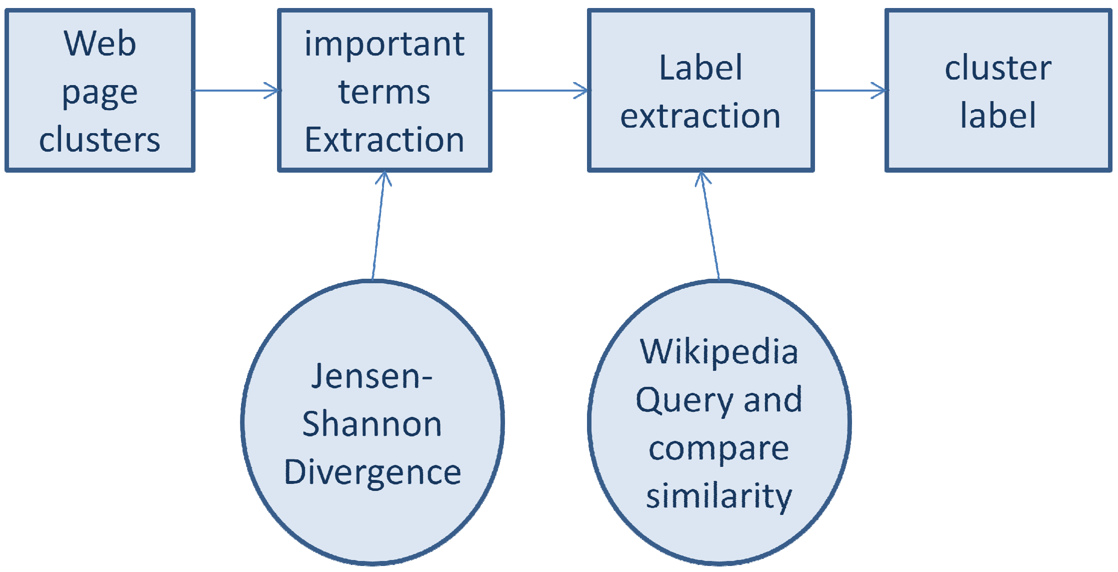

Web page clusters
important terms Extraction
Label extraction
cluster label
Jensen-Shannon Divergence
Wikipedia Query and compare similarity

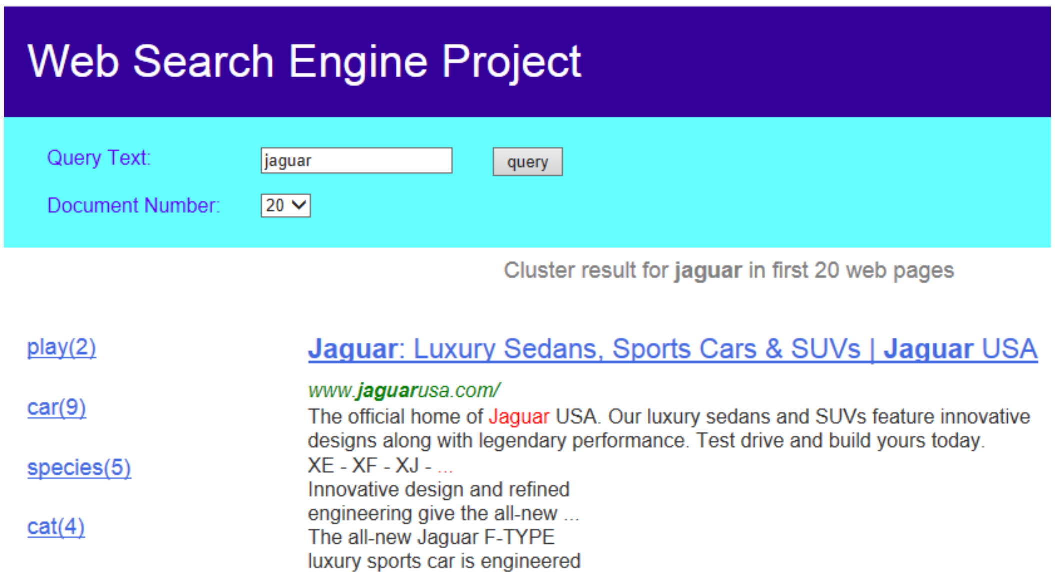
Figure 3. Process of finding cluster labels

(1) Important terms extraction:
I use "Jensen-Shannon Divergence (JSD)" distance (use MALLET API [6]) to find the important terms that contribute to the JSD distance between the cluster and the whole collection.

(2) Label extraction:
For each important term in a cluster, query it from Wikipedia to find corresponding Wikipedia page. Compare the similarity between the Wikipedia page and the center of cluster, and choose the most similar word as label.

Other System Features

1. The project is written by Java
2. The user interface looks like Google, with a text input box.
3. The query result interface looks like Yippy. The cluster results are shown in the left as links. When the user clicks on the cluster link, the system will show the pages of this particular cluster.



Web Search Engine Project

Query Text: jaguar

query

Document Number: 20

Cluster result for **jaguar** in first 20 web pages

play(2)

car(9)

species(5)

cat(4)

Jaguar: Luxury Sedans, Sports Cars & SUVs | Jaguar USA

www.jaguarusa.com/

The official home of Jaguar USA. Our luxury sedans and SUVs feature innovative designs along with legendary performance. Test drive and build yours today.

XE - XF - XJ - ...

Innovative design and refined engineering give the all-new ...

The all-new Jaguar F-TYPE luxury sports car is engineered

Figure 4. System layout.

Cluster result for “jaguar” with first 20 Google search result pages.

Achievements:

1. Acceptable clustering result

   Figure 4 is one of the clustering result using “jaguar” as query term with the first 20 Google search results. Please note that the search result will be slightly different each time because there are random factors in Bisecting K-mean clustering algorithm.

Label	Correct	Incorrect	Possible reasons for error occur
Play	0	2	These two web pages are youTube pages, and there are few words to represent their topic.
Car	9	0	-
Species	4	1	The error clustered page is a map selector with almost no words about its topic. See Figure 5.
Cat	4	0	-
Total	85%	15%	-

Table 1. Correct/Incorrect rate for each cluster

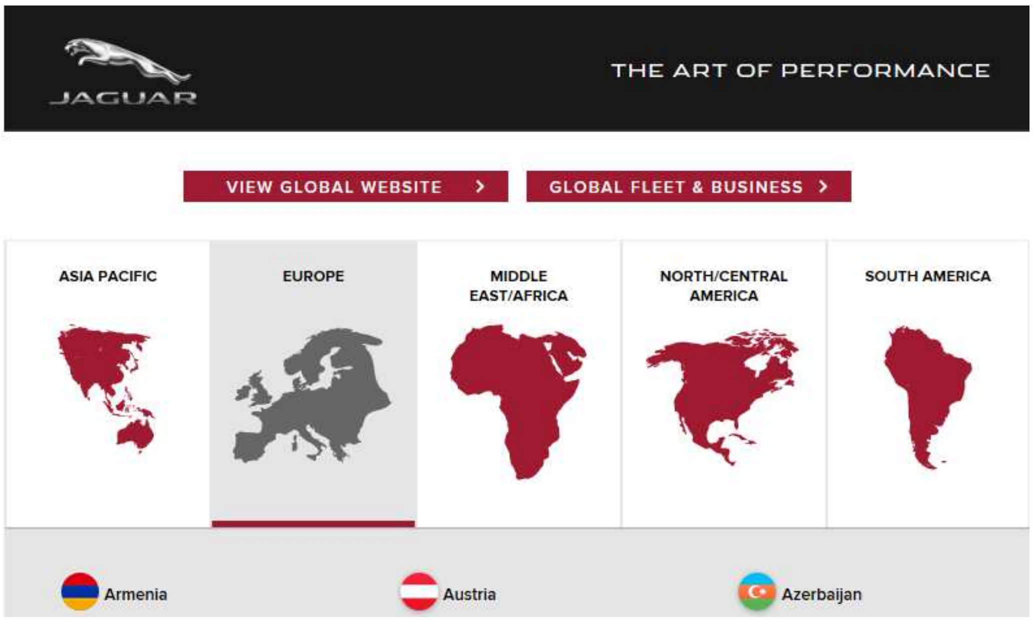


Figure 5. An Example of web page with almost no text content. (http://www.jaguar.com/market-selector.html)

2. Good label on pure clusters
   From table 1, we can see the labels for "Cat" and "Car" are very representative for their clusters. The term "species" are not as good as "Cat", but it is still acceptable. The only bad label is "play", and this caused by the lack of text contents in these pages.

Problems and difficulties:

1. Performance issue:
   The current system is too slow. It takes several minutes to show result for each query. This system relies on some web pages parsing and downloading. These web pages connections are the bottleneck of the performance of this system. There are several approaches to solve the performance issue:
   (1) Use multi-threads to download web page files in parallel.
   (2) Use local index rather than download web page from Google. However, we all know Google has best indexes, and we will need to take a lot of effort to build and update our local index files.
2. Some pages have few text content:
   See Table 1 and Figure 5 as an example. This will reduce the performance of clustering output.
3. Some pages require to login before seeing the page contents.
   Same with 2. This will also reduce the performance of clustering output.
4. Cannot find labels with two words. Currently this system can only find labels with

a single word. It cannot find labels with more than one word such as “Star War”. Here are some ideas to deal with this problem:
Use multi-gram: When we tokenize the web page content, we can also take two or three words as a whole and check in wikipedia to see whether these words are usually used together.

5. Https connection issue:
   I see several different connection errors when the application tries to download some of the web pages, such as “PKIX path building failed” and “handshake_failure”. I solved the first problem by setting the application’s http connection to trust all web page managers. For the second error, I didn’t find a simple solution so I just ignore the page.
6. Version problem:
   I first used Stanford coreNLP to tokenize and lemmatize the web content, but this API can only run in java 1.8 environment so it won’t be able to used in CIMS server. I used JWI to query wordNet instead.

Future Work:

1. Improve the query speed:
   Use multi-threading to download the web pages in parallel, or calculate different potential clusters in different threads.
2. Improve the clustering performance:
   The clustering output is not accurate enough, especially when there are pages with few text contents.
3. More stable:
   There are some http connection issues such as timeout or https handshake failure issue.

Web Resource:

Google
Wikipedia

Reference:

[1] M. Steinbach, G. Karypis, and V. Kumar. A Comparison of Document Clustering Techniques. In: KDD workshop on Text Mining (2002).
[2] D. Carmel, H. Roitman, and N. Zwerdling. Enhancing Cluster Labeling Using Wikipedia. *In SIGIR ’09*, pages 139-146, New York, NY, USA, 2009. ACM.
[3] Word frequency: https://invokeit.wordpress.com/frequency-word-lists/
[4] Wiktionary: https://en.wiktionary.org/wiki/Wiktionary:Frequency_lists

[5] Yippy: http://yippy.com/
[6] MAchine Learning For LanguagE Toolkit (MALLET)
http://mallet.cs.umass.edu/api/cc/mallet/util/package-summary.html
[7] Stanford coreNLP: http://stanfordnlp.github.io/CoreNLP/
[8] WordNet: https://wordnet.princeton.edu/
[9] Jsoup library: https://jsoup.org/
[10] S-Space package https://github.com/fozziethebeat/S-Space/wiki
[11] JWI http://projects.csail.mit.edu/jwi/
[12] TF-IDF https://en.wikipedia.org/wiki/Tf%E2%80%93idf
[13] Determining the number of clusters in a data set
https://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set