Computer Science & Engineering, Santa Clara University

Refine search engine results with patterns

COEN493 Directed Research Report

Shouchun He (SCU ID: 1008350)

Table of Contents

Refine search engine results with patterns

1. Background

1.1 Search engine

Web search engines, like Google, Baidu, Bing, are powerful software systems which allow users to search for information on the World Wide Web. Users submit the key words and then the search engine process and feedback several rows of results. The search engine system provides us big convenience on information retrieving and problem solving.

1.2 Problems

But sometimes the search engine systems are not so perfect:

Firstly, they return thousands or even millions of results, and the users need to manually filter out the information that they really want. Sometimes this filter process takes very long time.

Secondly, many results are similar or even identical to each other. Though they came from different resources or websites, the contents are almost the same. This problem can be frequently found in the results of Baidu.com because many Chinese Internet users prefer to copy and paste information from other resources rather than create innovative information by themselves.

Thirdly, since there are too many results returned, the most useful and optimal result may be hidden in the middle of the results but the users may only check the first ten to one hundred results. Additionally, most users have no patience to check all the results, the search engine wasted computing resource, energy, storage, and network bandwidth on preparing those results which will never be seen and used by the end users.

1.3 Cases study

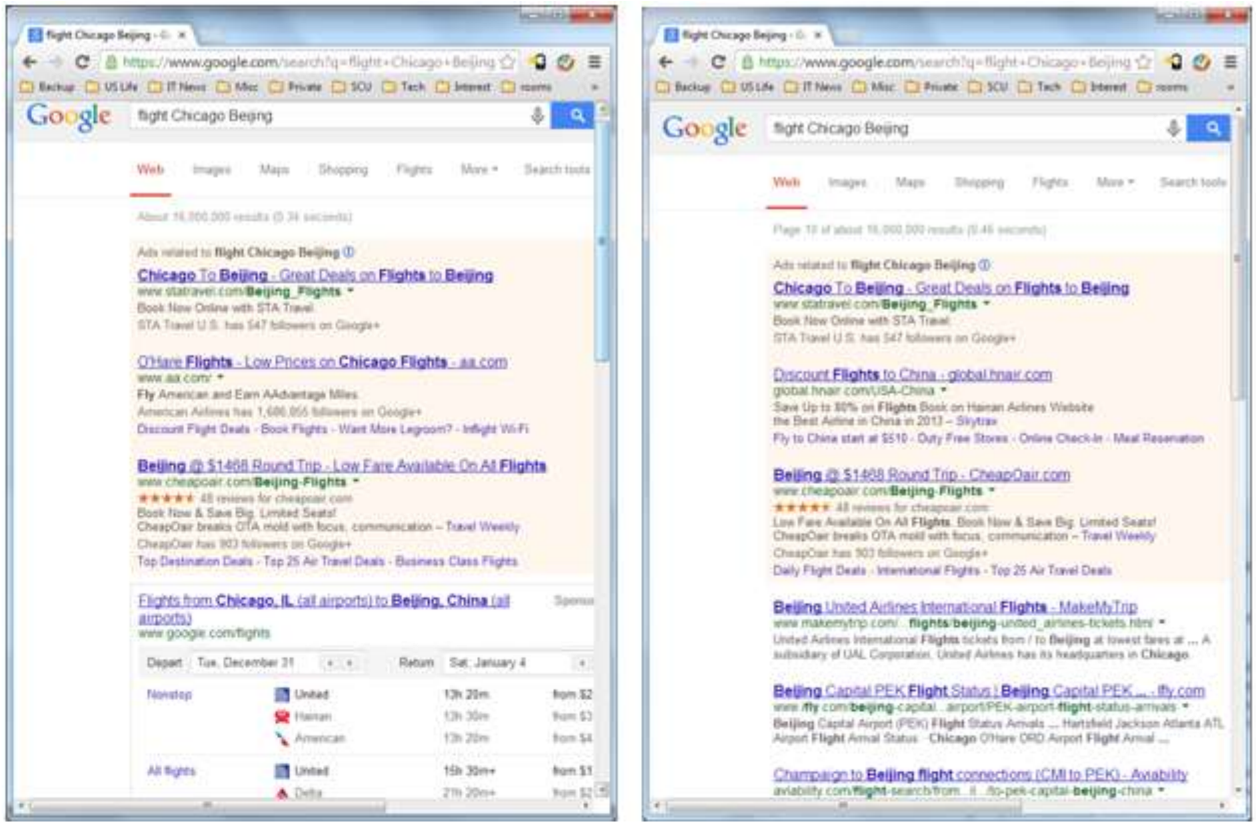
Below are examples of some typical keywords:

flight Chicago Beijing

thai food SCU

incluencing up book

For "flight <city A> <city B>", the user just wants to know the information about the flight between the <city A> and <city B>, such as, air lines, fare, schedule, and some recent news.



flight Chicago Beijing

https://www.google.com/search?q=flight+Chicago+Beijing

Backup · US Life · IT News · Misc · Private · SCU · Tech · Interest · rooms

Google · flight Chicago Beijing

Web · Images · Maps · Shopping · Flights · More · Search tools

About 16,000,000 results (0.34 seconds)

Ads related to **flight Chicago Beijing**

Chicago To **Beijing** - Great Deals on **Flights** to **Beijing**
www.statravel.com/**Beijing_Flights**
Book Now Online with STA Travel
STA Travel U.S. has 547 followers on Google+

O'Hare **Flights** - Low Prices on **Chicago Flights** - aa.com
www.aa.com/
Fly American and Earn AAdvantage Miles
American Airlines has 1,686,055 followers on Google+
Discount Flight Deals - Book Flights - Want More Legroom? - Inflight Wi-Fi

Beijing @ $1468 Round Trip - Low Fare Available On All **Flights**
www.cheapoair.com/**Beijing-Flights**
★★★★★ 48 reviews for cheapoair.com
Book Now & Save Big. Limited Seats!
CheapOair breaks OTA mold with focus, communication – Travel Weekly
CheapOair has 903 followers on Google+
Top Destination Deals - Top 25 Air Travel Deals - Business Class Flights

Flights from **Chicago, IL** (all airports) to **Beijing, China** (all airports)
www.google.com/flights

Depart: Tue, December 31 · Return: Sat, January 4

Nonstop	United	13h 20m	from $2
	Hainan	13h 30m	from $3
	American	13h 20m	from $4
All flights	United	15h 30m+	from $1
	Delta	21h 20m+	from $2

flight Chicago Beijing

https://www.google.com/search?q=flight+Chicago+Beijing

Google · flight Chicago Beijing

Web · Images · Maps · Shopping · Flights · More · Search tools

Page 10 of about 16,000,000 results (0.46 seconds)

Ads related to **flight Chicago Beijing**

Chicago To **Beijing** - Great Deals on **Flights** to **Beijing**
www.statravel.com/**Beijing_Flights**
Book Now Online with STA Travel
STA Travel U.S. has 547 followers on Google+

Discount **Flights** to China - global.hnair.com
global.hnair.com/USA-China
Save Up to 80% on **Flights** Book on Hainan Airlines Website
the Best Airline in China in 2013 – Skytrax
Fly to China start at $510 - Duty Free Stores - Online Check-In - Meal Reservation

Beijing @ $1408 Round Trip - CheapOair.com
www.cheapoair.com/**Beijing-Flights**
★★★★★ 48 reviews for cheapoair.com
Low Fare Available On All **Flights**. Book Now & Save Big. Limited Seats!
CheapOair breaks OTA mold with focus, communication – Travel Weekly
CheapOair has 903 followers on Google+
Daily Flight Deals - International Flights - Top 25 Air Travel Deals

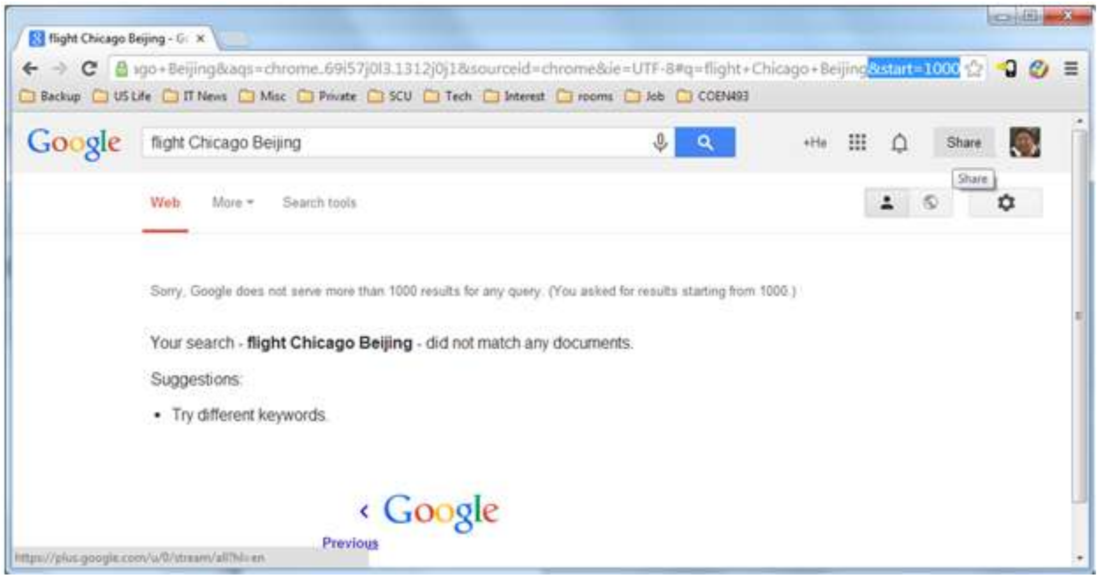
Beijing United Airlines International **Flights** - MakeMyTrip
www.makemytrip.com/.../**flights/beijing**-united_airlines-tickets.html
United Airlines International **Flights** tickets from / to **Beijing** at lowest fares at ... A subsidiary of UAL Corporation. United Airlines has its headquarters in **Chicago**.

Beijing Capital PEK **Flight** Status | **Beijing** Capital PEK ... - ifly.com
www.**ifly**.com/**beijing**-capital.../airport/PEK-airport-**flight**-status-arrivals
Beijing Capital Airport (PEK) **Flight** Status Arrivals ... Hartsfield Jackson Atlanta ATL Airport **Flight** Arrival Status · **Chicago** O'Hare ORD Airport **Flight** Arrival ...

Champaign to **Beijing flight** connections (CMI to PEK) - Aviability
aviability.com/**flight**-search/from.../il.../to-pek-capital-**beijing**-china

Figure 1-1 (a) The first page of searching result (b) The 10th page of searching result

But Google prepared about 16,000,000 results for this search. (See figure 1-1 a) Is there anyone who has willing to read all the results? Or even has time to check 1/10000 of these results? Even Google knows no one care about those much results so when you are going to check the 1001th result it displays the page like figure 1-2.


flight Chicago Beijing - G
ıgo+Beijing&aqs=chrome..69i57j0l3.1312j0j1&sourceid=chrome&ie=UTF-8#q=flight+Chicago+Beijing&start=1000
Backup
US Life
IT News
Misc
Private
SCU
Tech
Interest
rooms
Job
COEN493
flight Chicago Beijing
+He
Share
Share
Web
More
Search tools
Previous
https://plus.google.com/u/0/stream/all?hl=en

Sorry, Google does not serve more than 1000 results for any query. (You asked for results starting from 1000.)

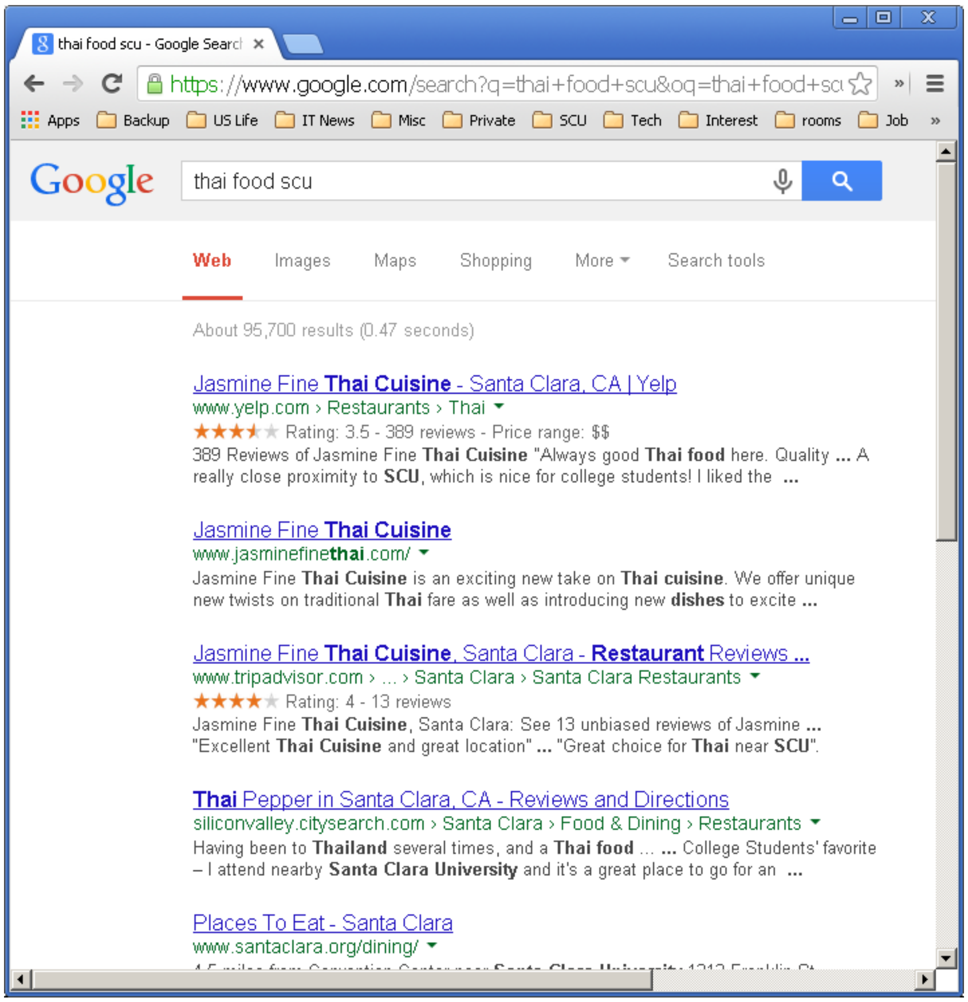
Your search - **flight Chicago Beijing** - did not match any documents.

Suggestions:

- Try different keywords.

Figure 1-2 Want to see the results after the 1000th

For the 2nd keyword, any human can understand that the submitter wants to know the Thai food around a place called SCU. (Sometimes the user even did not specify any space, but the search engine can acquire the location information of the query submitter from the browser of the client.)


thai food scu - Google Search

https://www.google.com/search?q=thai+food+scu&oq=thai+food+scu

Apps | Backup | US Life | IT News | Misc | Private | SCU | Tech | Interest | rooms | Job

thai food scu

Web | Images | Maps | Shopping | More | Search tools

About 95,700 results (0.47 seconds)

[Jasmine Fine **Thai Cuisine** - Santa Clara, CA | Yelp]
www.yelp.com › Restaurants › Thai
★★★½ Rating: 3.5 - 389 reviews - Price range: $$
389 Reviews of Jasmine Fine **Thai Cuisine** "Always good **Thai food** here. Quality **...** A really close proximity to **SCU**, which is nice for college students! I liked the **...**

[Jasmine Fine **Thai Cuisine**]
www.jasminefine**thai**.com/
Jasmine Fine **Thai Cuisine** is an exciting new take on **Thai cuisine**. We offer unique new twists on traditional **Thai** fare as well as introducing new **dishes** to excite **...**

[Jasmine Fine **Thai Cuisine**, Santa Clara - **Restaurant** Reviews ...]
www.tripadvisor.com › ... › Santa Clara › Santa Clara Restaurants
★★★★ Rating: 4 - 13 reviews
Jasmine Fine **Thai Cuisine**, Santa Clara: See 13 unbiased reviews of Jasmine **...** "Excellent **Thai Cuisine** and great location" **...** "Great choice for **Thai** near **SCU**".

[**Thai** Pepper in Santa Clara, CA - Reviews and Directions]
siliconvalley.citysearch.com › Santa Clara › Food & Dining › Restaurants
Having been to **Thailand** several times, and a **Thai food** ... **...** College Students' favorite – I attend nearby **Santa Clara University** and it's a great place to go for an **...**

[Places To Eat - Santa Clara]
www.santaclara.org/dining/

Figure 1-3 Example of searching food information

We can see that Google replied with about 100,000 results. Apparently the first 10 results are enough for the end users. The search engine need not provide so much results – User may never read them and it waste storage, CPU time, and other resources.

For the 3rd keyword, it is obviously that the user wants to search information about book “influening up”. But there are some other results have nothing to do with the book are provided by Google.

1.4 Possibility of improvements

In most cases, the end users may only want to quickly get the answer they want from search engine. 3 – 15 highly related and focused results are enough for them to reach what they want.

So, if we can rate and refine the results and choose the three to twenty most closely related results and present them to the end users, the user experience will be significantly improved. For most keywords, it is quite easy for any human beings to do the judgment based on their knowledge, but to computer systems, they are less intelligent and could not check how close the relationship between a motley variety of key words and searching results.

However, fortunately, for some keywords, almost all the results have the similar results and almost all the results have equal relationship with the key words. So if the pre-processor can detect the pattern of these key words, they can tell the search engine system that it only needs to process the search request in a simple way and return only a limited amount of results can satisfy the submitter of the query.

1.5 Goal and approaches of this research

My research is focusing on how to solve the problems that the existing search engines have as I mentioned above: Too many results; duplicated contents in the results; sometimes rely on the human to review the results and pick for useful information. And the ultimate goal of this research is to provide a new search engine system which:

- Can smartly identify the patterns of the keywords
- Use the identified patterns to get more closely matched search results
- Remove the duplicated results (If several results from different resource are identical in contents, only show one of them)
- Return a limited amount (only 3 to 15) of results per search, to save the bandwidth, and the filtering time of end users.

This research will focus on the following approaches:

- Design the flow and algorithm to check the user input and match the patterns
- Apply the information from the pattern to the refining of the searching results

- Implement the ideas above, and compare it with existing search engines and evaluate for its usability, accuracy and relativity of the searching results.
- Identify the value of this research and potential business model.

1.6 Chapters introduction

Chapter 2 introduces the pattern matching flow, and how to use the matched patterns to refine the search results in concept level.

Chapter 3 gives detail introduction of algorithms and procedures for the pattern matching and how to perform searching based on pattern information.

Chapter 4 introduces how I implement my ideas based on the Google search.

Chapter 5 gives a brief market value analysis for this research, just for attracting some funding and VC to help me to do the research in depth and utilize it for the real business.

2. Patterns handling flow

2.1 Patterns in key words

As we discussed in chapter one, most of the keywords user input fully or partially match some patterns. So it is possible to handle those keywords with special measures and offer small amount but more accurate results to the end users to satisfy their requirements. Those patterns can be:

- [From] {<location A>} [to] {<location B>} [<budget>] {<vehicle| <transportation>}
- {<food style>} {<location>|<location got from device>}
- {<gift>} {<price>} [<location>]
- {<name of a book>} {book} [<author>]
- ….

Note about the syntax:

<word>: Something matches the domains described by the "word".

{…}: Mandatory parts

[…]: Optional parts

If the search engine recognizes that the keywords input by the users match one or more patterns, it can use some special routine to handle it rather than handle it with a generic routine. In later sections I will discuss this in detail.

2.2 Overall introduction of patterns handling flow

Below is the flow chart of how to handle the search by patterns matching.

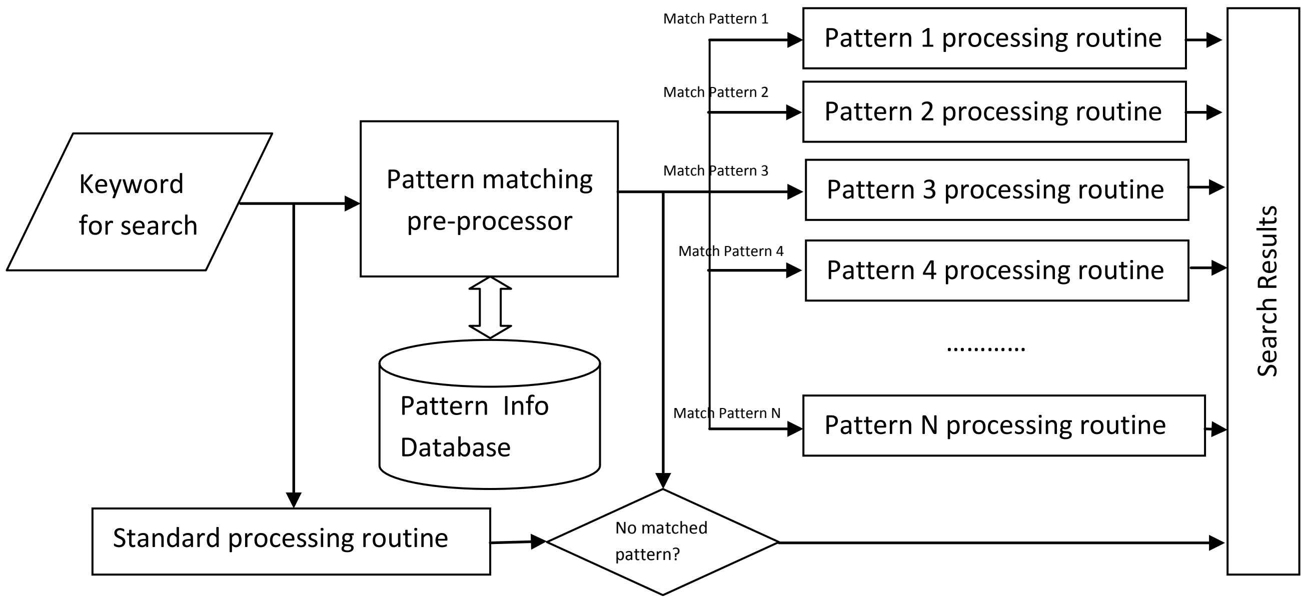

Keyword for search
Pattern matching pre-processor
Pattern Info Database
Match Pattern 1
Pattern 1 processing routine
Match Pattern 2
Pattern 2 processing routine
Match Pattern 3
Pattern 3 processing routine
Match Pattern 4
Pattern 4 processing routine
…………
Match Pattern N
Pattern N processing routine
Search Results
Standard processing routine
No matched pattern?

Figure 2-1 Overall flow of pattern matching and searching with patterns

When the search engine receives a keyword input by end user, it starts the pattern match checking. Since there are lots of pre-defined patterns, the search engine system can run the pattern match checking for all the patterns in parallel with multi-threading, multi-processing, or other parallel computing technologies. The matching for each pattern can be run in a separate thread or process. At the same time, the system starts another thread or process to do the regular search processing in case of that the key word cannot meet any existing pattern.

2.3 Patterns matching

2.3.1 Database of the patterns

To identify the patterns, there should have a database which stores key information about the patterns that the search engine supports. For example, for the traffic query pattern:

[From] {location A} [to] {location B} [{budget} {vehicle}| {transportation}|…]

The search engine must have a database table which stores all the locations, such as, cities, countries, islands, famous place (the pyramids in Egypt, the Yosemite national park, etc.), and so on.

For the food service pattern, the search engine must prepare the typical food styles, names, such as, “Thai food”, “sushi”, “pizza”, etc.

However, some pattern data, like the book names, hotel names, flight and train schedules, are huge in size and changed dynamically. So our search engine pattern database cannot store all of them, but need to retrieve them from the external databases, such as the public library databases, third party databases, travel information systems, etc.

2.3.2 Semantics analysis in the pattern matching

In the implementation of generic search engine, the system always break the keyword user input into individual words, and then search them in the back-end indexed database, and then feedback the results sorted by the ratio of relativity.

But in this research, I tried to check whether the keyword input by end users match the patterns first. If yes, then the keyword will go into a special processing routine for such pattern which can provide only 3 to 15 results which are more accurate. And my personalized search engine does the pattern matching in the following steps:

1. Break the keyword into semantics units (English word, or combination of words), and remove those meaningless information;
2. Create a tree of semantics units. In each node of the tree, there are some attributes, such as, categories (location, food style, number, etc.), length, and ratio of length in the whole keyword, etc. These attributes are used for further analysis of which patterns this keyword belongs to and the matching weight.
3. Search the semantics tree nodes in the pattern information databases (include internal database and external resources, such as third party resources), and update the category attribute the node. Each node may belong to zero, one, or more categories.

The search of semantics tree nodes in pattern database can run in parallel and the system will keep synchronization of the all the searching results and attributes update.

4. Check the categories attributes of the semantics tree nodes, and then calculate the matching weight of the whole keyword to each pattern.

2.3.3 Pattern matching weight

Sometimes the keywords cannot fully match the patterns. So in this research I used the weight to indicate how much degree of the matching between the key words and the patterns.

If all mandatory categories of a pattern appear in the keyword, then we can say the pattern matching weight of this keyword to that pattern is 1.0. Or, there the weight will be calculated by the ratio of mandatory categories available. Different categories in a pattern can have different weights so the overall weight is calculated by the sum of the weight of matched parts in the pattern. For example:

For the book information pattern "{<name of a book>} {book} [<author>]", let's assume the weight of "{<name of a book>} is 0.8, and the "{book}" is 0.2. If user inputs a key word "Gone with the wind", the search engine can detect that it is 80% matching the book information pattern.

If the pattern matching weight is higher than 0.8, we can assume that the keyword is almost matching that pattern and handle that query with the special routine for that pattern.

2.3.4 Match on multiple patterns

Sometimes a keyword matches, or more than 80% matches more than one pattern, then the search engine needs to handle the query with all the corresponding routines for these matched patterns. For example, the user enters a keyword:

Travel to Egypt

The user may either want a guide book about travelling in Egypt, or quotations and itineraries provided by the travel agencies.

Once the pre-processor of the search engine detects that the key word matches multiple patterns, it will follow all the corresponding routings to search and finally list all the searching results. It can either merge all the results, or separate them in groups and then present to the end users.

2.4 Process keywords which match patterns to refine the search results

2.4.1 Indices for the patterns

When the search engine system builds the indices for the web links, it can consider the pattern in the indexing. For each web link, add an additional property called "pattern IDs" to indicate that its contents contains information for those patterns. So that the search engine can further narrow the scope of results for the specified keyword and ignore those less relative results.

The process of checking the patterns matching in the page is almost the same as that process for matching keywords to the

2.4.2 Remove the duplicated results

The search engine system also should smartly check the content similarity when it builds the indices for the web links. When there are no patterns introduced, it is almost impossible for the system to compare the similarity of the pages because there are trillions of . With the patterns, the search engine can compare the contents of the results which match the same pattern. If they are identical in content or almost identical except some style and format difference, the search engine can mark those two pages as "same" and add some "reference tags" to the attribute of these pages to indicate the duplication. So that in the future the system only need to return one page if it find that those pages have the same "reference tag".

2.4.3 Return a limited amount of results for each matched pattern

For each pattern, there is a pre-defined number which indicates the maximum count of results to be returned to end user.

Just like in the generic search engine, the results should be sorted by the relativity. The results highly relate to the keyword are listed in the top, and vice versa. Since the patterns are carefully selected so that we are quite sure that if the keyword matches a pattern, the end user may only want 3 to 15 results for him/her to refer. Any more results are not necessary.

If a keyword matches multiple patterns, the search engine should return all the search results returned from different patterns and present them to the end user. The system can either merge them together in a single list, or show them separately in multiple lists. For example, display them in lists of different tables in a web page.

2.5 Standard search process running in parallel

In my design, the search engine also runs a standard search process in background in parallel for the key words. If there are no match patterns found, it returns 3 – 15 matched results to the end users. These results should came from different resource and have no close similarity to each other to ensure the diversity of the results end users can get.

The standard search process can be terminated once the search engine pre-processor recognizes that the keyword matches one or more given patterns and some refined search results are available. And it can also be terminated when there have already been 15 diversified results found.

3. Algorithm and procedures

In this chapter, I will list the detailed algorithms and procedures I introduced in chapter 2. To make it easier to understand, I describe all the algorithms and procedures in text, graphs and examples rather than code, pseudo code and flow charts.

3.1 Semantics analysis in the keywords

The search engine pre-processor needs to first do semantics analysis for the keywords. It breaks the keyword into several semantics units by the space in the keyword. If there are N spaces in the keyword, the keyword can be broken into (N+1) word units, and they can attach with the adhered word units to form combined semantics units. So the total number of semantics units can be: 1 + 2 + 3 + … + (N+1) = (N+2) * (N+1) /2 = $N^2/2 + 3N/2 + 1$.

3.1.1 Semantics analysis tree

We can build a semantics analysis tree like below. Each semantics unit is a node in the tree:

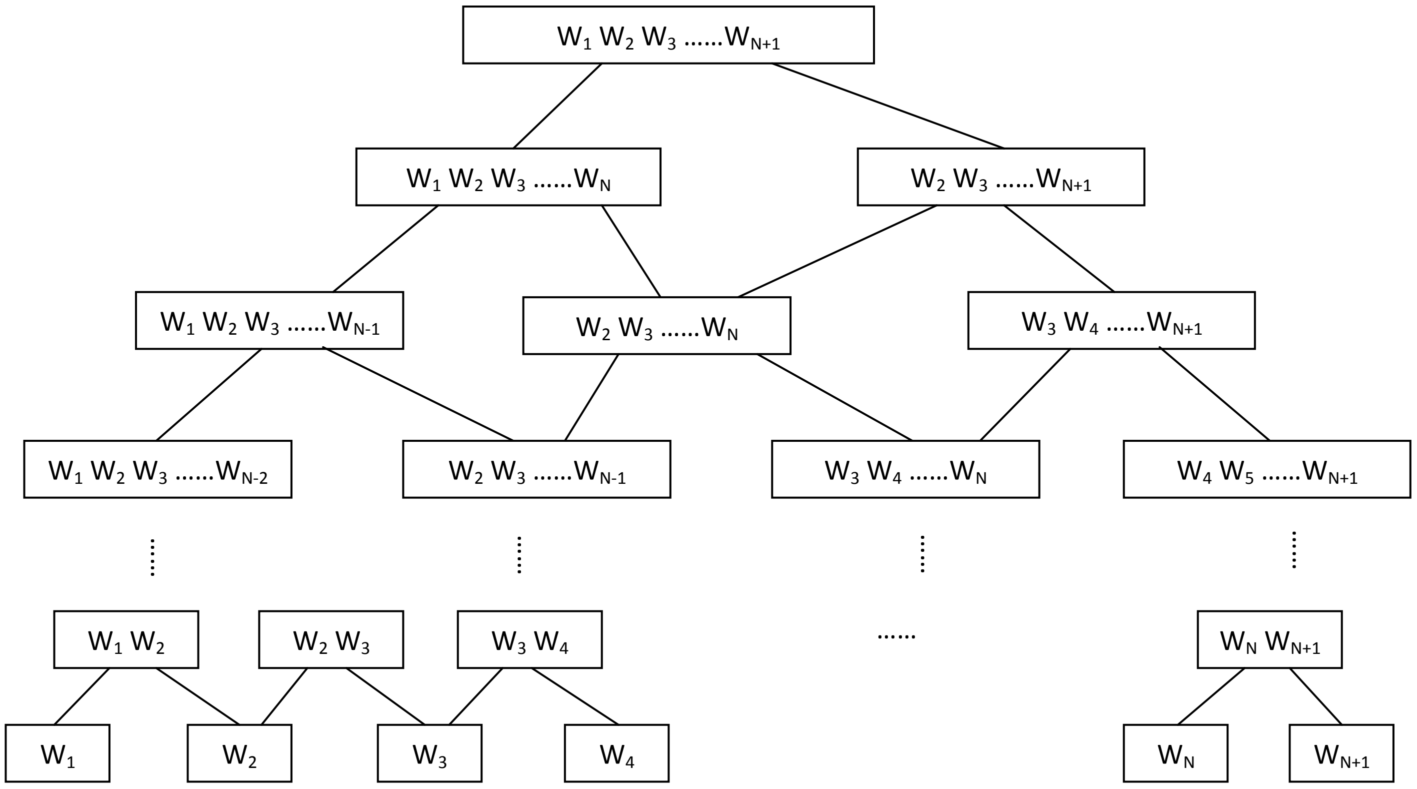


Figure 3-1 Semantics analysis tree for pattern matching

The tree has the following characteristics:

- There are (N+1) levels. In the k'th (1 <= k <= N+1) level, there are k nodes.
- Each non-leaf node has two direct children. The left child contains all its words except the right most one; the right child contains all its words except the left most one.
- Each node, except the first one and the last one in the level, has two parents.

3.1.2 Update the categories of the nodes

The pre-processor of the search engine searches the semantics units in the pattern information database to check which categories they belong to. It firstly searches the nodes at the root, then the second level, and so on, until it finishes searching the last node in the last level.

If the pre-processor finds that a node belongs to a category, then it needs not check the children and all the offspring nodes of that node in that category.

Since there are many categories which are independent to each other, the pre-processor can search each semantics unit in multiple category tables or external resources in parallel to shorten the searching time. And once a parent node has been searched in a category table or external resource, the pre-processor can start search its child nodes for that category.

3.2 Pattern matching

The pattern matching logics are pattern dependent. That means, for each given pattern, there is a special pattern matching logic. However, there are some common shared features among those logics:

- The search engine checks the categories attribute of the semantic units we got in 3.1.2, and compare them with the requirements of each pattern, and calculate the pattern matching weight.
- Each pre-defined pattern requires one or more semantics unit which belongs to specific categories, and these semantics unit should be arranged in certain order.

Example 1: The book information pattern is defined as:

{<name of a book>} {book} [<author>]

There are two mandatory fields, the <name of book> field should belong to the “book name” category, which takes 80% weight; another field should be “book”, which takes 20% weight. The order of the two fields can be random.

Example 2: The travel information pattern is defined as:

[From] {<location A>} [to] {<location B>} [<budget>] {<vehicle>}

There are three mandatory fields: <location A> and <location B> should belong to “location” category, while the third field should belong to the vehicle or transportation (air, flight, train, etc.) category. The orders are not required but always assume that the first location is the start point and the second one is the end point. The locations take 40% weight each and the vehicle takes 20% weight. So even if the vehicle field is missed in the keyword the search engine still think this keyword matches the travel information pattern because the overall pattern matching weight has reached 0.8.

Since the patterns are independent to each other, the pre-processor of the search engine should run the pattern matching processes in parallel. Once it finds a matched pattern for this keyword, it should start the pattern specific search process and need not wait for the finish of other pattern matching processes.

3.3 Searching with patterns

3.3.1 Narrow the searching scope with patterns

The patterns can help the search engine to narrow the searching scope for a given keyword. For example, if a keyword matches the book information pattern, the search engine only need to search it in the book related websites, such as Amazon.com, books.google.com, etc. If another keyword matches the food information pattern, the search engine may focus on the web pages from Yelp.com, foursquare.com, Tapingo.com, etc. So that the response time can be much quicker, the results can be more accurate, and the results are more favorable to the end users.

3.3.2 Generate web link to call web APIs provided by specialized websites

For most patterns, the search engine can simply generate a web link to call the web APIs of the famous websites, such as, Amazon.com, Yelp.com, eBay.com, kayak.com, expedia.com, etc. Providing web links is much easier than using web crawlers to retrieve data, store and index them, and prepare the search results for the queries.

Example: User inputs keyword "fly San Francisco Las Vegas"

The search engine recognizes the keyword matches the flight information pattern:

<vehicle> = "flight", <location A> = "San Francisco", <location B>="Las Vegas"

Then it can generate a link to call the expedia.com API:

http://www.expedia.com/Flights-Search?leg1=from%3Asan+francisco%2Cto%3ALas+Vegas

And the search engine can even call the API and parse the feedbacks to show more user friendly results to the end users.

The other features, such as removing duplicated results and return a limited number of highly relative results, are introduced in section 2.4.

4. An implementation

4.1 Brief introduction

To support this research, I wrote lots of front-end and back-end code in HTML, AJAX and Java to prove the feasibility of my approaches. During the whole research my ideas continuously evolved and my code has been updated from time to time for several versions. In this chapter I will briefly introduce my latest version of the implementation.

Since I worked on this project by myself, I could not build a powerful standard engine from the start, nor could I collect enough data to fill in the pattern information database. To reduce my workload, I implemented my personal search engine based on the public search engine provided by Google. My program used the Google search engine as the standard search engine, and I acquired most of data of my pattern information database through Google searching results.

4.2 Flow chart

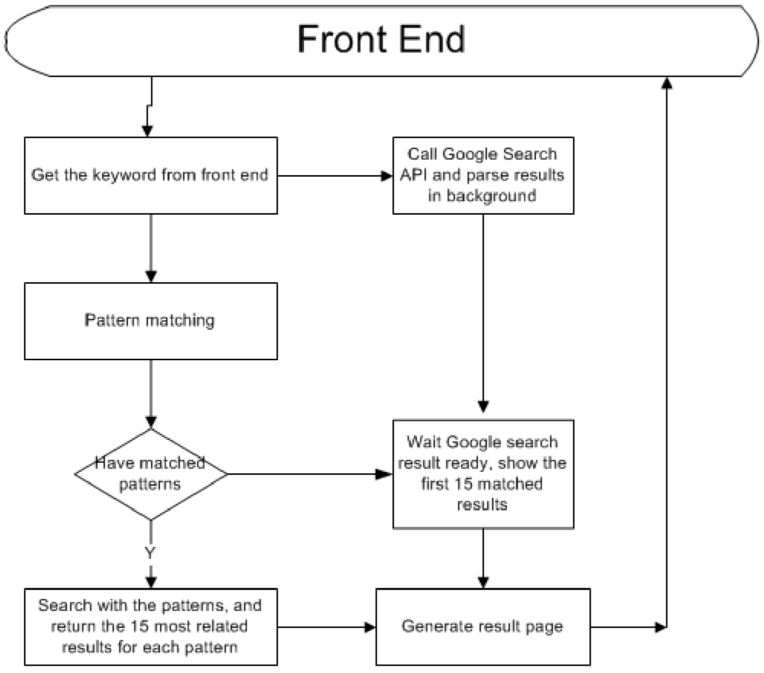

Front End
Get the keyword from front end
Call Google Search API and parse results in background
Pattern matching
Have matched patterns
Y
Wait Google search result ready, show the first 15 matched results
Search with the patterns, and return the 15 most related results for each pattern
Generate result page

Figure 4-1 Flow chart of my implementation

5. Evaluation of this research and potential business value

5.1 Evaluation

This research tried new approaches to solve the problems of existing search engines by adding a pre-processor to match keywords with pre-defined patterns. The patterns can narrow the key word searching scope, simplify the query, speed up the response time, and return highly relative results. Additionally, the patterns also enable the search engine to remove duplicated contents in the results, and reduce the amount of results to 3 to 15 records. The user experience can be significantly improved.

However, the code implementation is still in the prototype stage and relies on the generic search engine. So the actual effect of this research still to be verified by defining more patterns and ask more users to try.

5.2 Potential business value

This research designed search engine systems which may significantly improve the user experience by high relative, less amount, and no diversified results. So it may quickly attract a big amount of end users who wants simple and easy life.

When there are enough end users, it is possible for the operator of the search engine system to sign contract with vendors to add their links into the searching results and get some revenue.