Journal of Management and Service Science, 2022,
Vol. 02, Iss. 02, S. No. 020, pp. 1-20
ISSN (Online): 2583-1798

Query Recommendation System in Social Networks

Anuradha Pillai

JC BOSE University of Science and Technology, YMCA Faridabad, India
anuangra@yahoo.com

How to cite this paper: A. Pillai, "Query Recommendation System in Social Networks," *Journal of Management and Service Science (JMSS)*, Vol. 02, Iss. 02, S. No. 020, pp. 1-20, 2022.

https://doi.org/10.54060/jmss.2022.20

Received: 05/09/2022
Accepted: 10/10/2022
Published: 25/11/2022

Abstract

Recommender Systems are software which provides suggestions to the user according to his or her interest. These suggestions are related to supporting users in making their decisions, for example what to search, what to buy, what to listen, etc. Recommender systems are very important in online stores where there are a lot of items to buy. These recommender systems help user to find things according to their interest and buy them. There are a lot of techniques proposed for recommendation and used in commercial environments. People are thought to trust suggestions from friends more than those from websites that are similar to them [2]. As a result, it is helpful to feed a recommender system with the friends' ratings. However, social media sharing websites' recommender systems have several difficulties, such as ranking the information from the user's friends as well, finding information from other sources in comparison to the user's immediate friends, and using metadata and context links for suggestion. In this research, an architecture based on profile-based crawling of social media sharing websites is proposed for query recommendation.

Keywords: *Text Mining, Indexing, Stemming, Recommendation System.*

1. Introduction

Social media sharing services, such as Flickr, YouTube, Amazon, and others, are growing in popularity in the current day. The primary factor for these websites' widespread popularity is their capacity to facilitate social interaction between users and their friends, as well as information sharing with the global community. The context information on these websites is abundant. There are two basic kinds of information on these sites. One is in the form of multimedia, rich text, and tag data that the uploader uploads to these websites and shares. In the uploader's profile, there are key details. We can determine the type of person the uploader is based on their interests and those of their friends.

Scanning these websites has gained popularity. The focused crawler that is still employed by traditional search engines consists of three basic components: the distiller, the classifier, and the crawler. These crawlers only allow certain subject searches. These days, these crawlers are made more effective by using the tags and profile data found on these websites; this process is known as profile-based crawling.

For these websites, a query suggestion system is suggested in this study. In order to extend the crawling subject and provide a consistent set of tags for a specific topic for effective crawling, this system initially employed co-tagging. It then used the uploader's profile for crawling. Finally, it suggests similar searches to the consumers.

For example, in flickr, each uploaded photo is tagged with different data. We use this information for our query recommendation purpose. Firstly, we have to make topic discovery for the corresponding topic. Suppose you search for flowers then it should include white, yellow, lily, etc this is known as co-tagging. To achieve this goal, we have to do page classification. Now focused crawling is done on this co-tagged data. The results of this focused crawling may have some irrelevant results. To remove such irrelevant results, an uploader’s profile is used to estimate whether corresponding link belongs to targeted topic or not and this type of crawling is called profile based focused crawling. DOM based page classification is used to classify the different list pages, detail pages and profile pages in both cases automatic co-tagging as well as profile based focused crawling. After returning the results to the user for targeted query, this system recommends queries to the users. These queries are recommended on the basis of the similarities in co-tagged data corresponding to each query and resulting URLs corresponding to each query. This query recommendation helps the users in searching.

Query recommendation helps the user in searching. Query recommender recommends the queries related to his interest. Suppose you are searching for lotus then on the basis of similarity between co-tagging data of this query and recently fired queries it finds out the similarity between queries and recommend the user such related queries like flower, lily, etc.

This work provides a novel query recommendation system for social media platforms. The proposed approach uses collaborative and content-based filtering. The complete framework contains a page classifier to classify the pages as per the content. Focused crawler is used to crawl the pages as per the focus area. Topic discovery system suggests topics. Similarity analyzer determines the similarity of the page content and focus topic. The favored query finder constructs the queries which are finally recommended by the recommendation system.

Section 2 encompasses extensive literature review. The subsections cover tagging and mining of social media, page classification and filtering techniques. The proposed framework is discussed in section 3. Section 4 exhibits implementation of proposed framework and its performance. Section 5 concludes the paper.

2. Literature Review

2.1. Tagging and folksonomies in social media

Websites that share content on social media are abundant in context-specific content. Tags and folksonomies are linked to context type information. These folksonomies are the descriptive words or free-form tags that are often linked to a particular resource, such as a document, video, or universal resource locator. Metadata of this kind is often used in social networks. Users may easily arrange information on social bookmarking sites like Delicious2 and photo sharing sites like flickr, which save this information in bulk. Users on social networks can distribute this information. For instance, social media sharing services like flickr enable users to interact socially with their peers in addition to sharing material on networks.

Heymann et al. [28] talk on the numerous uses of social bookmarking in Web search and how successful tagging is. Social media tagging and bookmarking have shown to be an expanding trend.

Given that Delicious has a far smaller index of URLs than the whole internet, it may be determined that these significant websites can be included in these systems. User-generated tag phenomena have been examined, and the usefulness of tagging has been assessed, by Brooks and Montanez [26]. The study also shows that the efficacy of tagging on these bookmarking sites is actually declining as more people join them and annotate more items with free-form tags.

The easiest approach to collectively organize and exchange information is by tagging, which is also the most effective method of information organization. A steady consensus develops despite stark variations in the reasons for labeling and in how tags are used [27]. One example of the use of tags is on social media sharing website like flickr is in extracting images. In this site, images are extracted on the basis of the tags. When a user searches for animals, only the images tags as animal will be retrieved. Also, an image can have multiple tagged data. For example, an image can be tagged with flower, flowers, lotus, red, etc. in this way tags have very important place in social media.

2.2. Mining techniques for social media

Because these sites include large amounts of multimedia data, it is essential to properly mine them in order to extract only pertinent data. Mining cannot be done using a structure as a criterion since every media asset has a different structure [23]. For instance, pattern recognition from a vast image collection is a need in image mining. These websites' content and contextual information can be mined for important information [21] [23].

In the case of picture retrieval, content information is connected to the visual and semantic qualities. Color, texture, form, and spatial information in the photos are examples of visual content [21]. One can characterize these visual characteristics locally or globally. While local visual characteristics are established for certain areas of the picture, global visual features are defined for the whole image. Either textual annotation or intricate inference processes based on visual components are used to provide semantic content [23].

Multimedia items and context objects can be linked to obtain context link information [1][21][24]. These context items are those that users have either directly or indirectly contributed. These context objects are the tagged data that are associated with each photograph on flickr, which might include title or caption data. Every photograph on flickr is properly annotated with metadata, either by the original uploader or by a user approved by the original uploader. Additional instances of tags include labeling URLs on Delicious, using the Hash tag on Twitter, annotating images on Facebook and Orkut, and labeling news and reviews on several platforms. We will talk about a few of these systems in the next section.

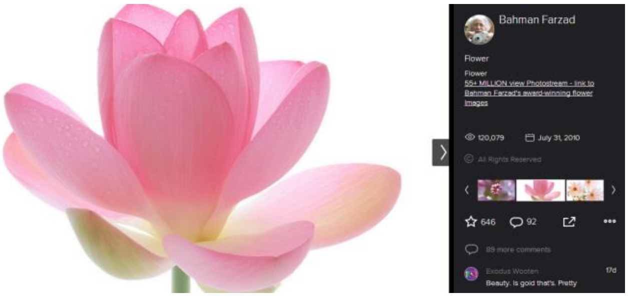
2.3. Structure of Social Media Sharing Websites

All the social media sharing websites like flickr, YouTube are generally similar in their structure. There are generally three types of pages:

2.3.1. List page: A list page is the page having a number of images or videos. A simple example of list page is the page on flickr which you get when you search anything on flickr. For example, when you search for animals, you get a list of images with their uploader’s name or small description. Crawling starts from this page therefore lst page is also called crawling hub page. A list page has many outlinks to detail pages.

Figure 1. Example of a List page on flickr

2.3.2. Detail page: A detail page generally has a large image or video with a detailed explanation about that image like uploader name, comments and tagged data. This detail page is the crawling target page. Crawling starts from the list page and ends with detail page. Detail page has set of keywords for which that image is tagged. This information is used for focused crawling.



Bahman Farzad

Flower

Flower
55+ MILLION view Photostream - link to Bahman Farzad's award-winning flower Images

120,079 July 31, 2010

All Rights Reserved

646 92

89 more comments

Exodus Wooten 17d
Beauty. Is gold that's. Pretty

Figure 2. Example of a detail page on flickr

3) Profile Page: Profile page has the information about the uploaders. It has various sections like sets, favorites, photo streams, etc. and uploader's friends and this information is used to make focused crawling efficient. These pages have two types of information. This information can be divided into two properties: inner properties and inter properties. Inner properties define the uploader's own interest such as uploader's set, photo stream, favorites. Inter properties is related to uploader's friends and their interest. This information is used to make focused crawling efficient.

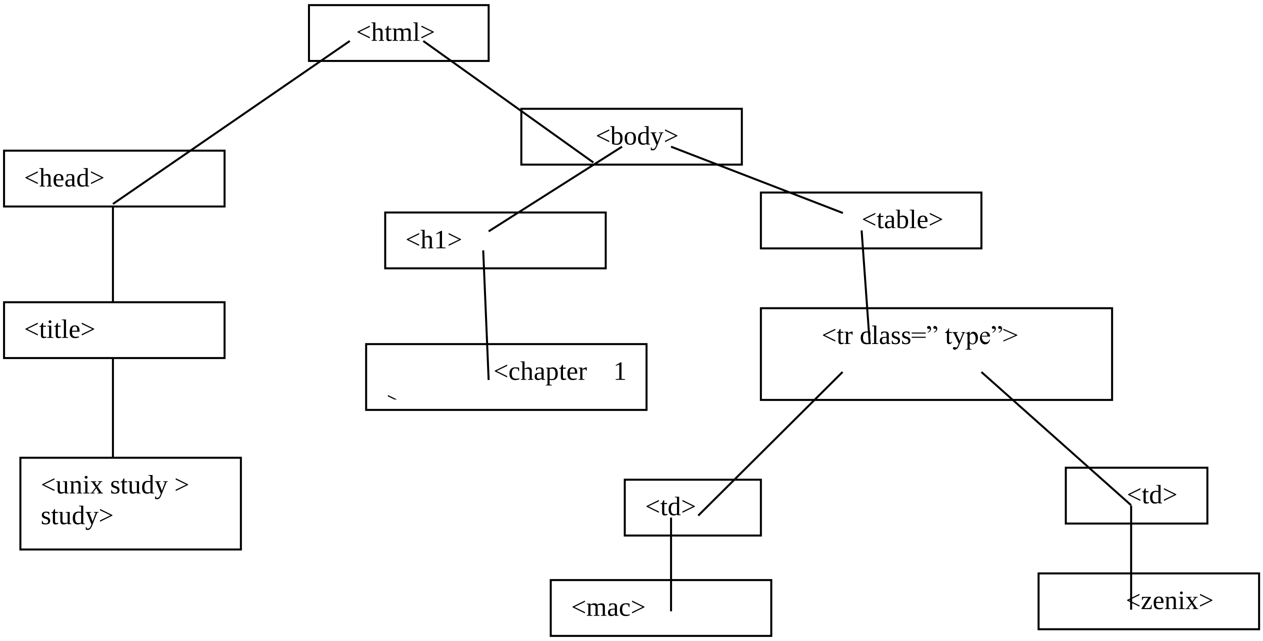


Figure 3. Dom Tree representation

2.4. Page Classification

To perform the crawling on these sites, these different pages on these sites need to be classified. To classify these pages, we use DOM path string based strategy.

How to create path strings from node tree: Concatenating strings from a node's immediate parent to the tree's root is known as the path string for that node. In addition, this string contains node properties. Property value and property name are concatenated using the characters "-" and "/".

Path strings for comparable sorts of pages are similar. There will be some common route strings across all list pages. In a similar vein, certain route strings will be shared by all detail pages. Since list pages contain a list of photographs together with the uploader's name, path strings pertaining to the uploader's name will be present on every list page. Path strings pertaining to tags will be shown on every detail page. Similar path strings are seen on profile pages for sets, favorites, etc. in this way, on the sites like flickr only one path string is enough to find the type of the pages.

There are also schema path strings. These path strings are not needed to classify different types of pages. These path strings are found in approximately all the pages. Some examples of these path strings are “Copyright”, “Terms of Use”. We can remove these path strings for proper classification.

2.5. Co-tagging Topic Discovery

The focused crawler must be fed the necessary crawler subject in order for it to begin crawling [1]. This crawling subject is too limited to meet all the standards if we consider it to be just one tag. Suppose we give the crawling topic as bird if we feed the crawler with this topic bird then it will return the images I which bird will be tagged. Suppose there is an image which is tagged as parrot then it should also return this image as output since the user who is searching for birds of course wants the images for parrots also. As a result, this crawling system should incorporate itself, as well as parrots, swans, crows, bats, etc., and crawl in accordance with each of these tags. In order to enable this one, we must use the tags connected to each image to expand the crawling subject to a certain point. Let's say we begin with the first issue, the bird. Links to the appropriate urls will be provided by the focused crawler. The picture that corresponds to these URLs will now additionally have additional tags applied to it. For example, an image of a swan may include tags for both swan and bird. Based on a vote method, we now choose whether or not to include these tags in our crawling subjects. We consider one vote for T1 if a picture has both our subject tag (t) and one tag T1. This tag T1 is also included in the crawling subject with T if we receive votes for it overall in all photos that are more than a certain threshold value. The focused crawler will then once more apply to this new crawling topic.

2.6. Profile Based Focused Crawling

Following the identification of co-tagging subjects, a targeted crawler searches the web for results related to newly discovered crawling topics. Sometimes a general-focused crawler will produce irrelevant results. For instance, a bus may be marked as plants on flickr even if it is exclusively used to deliver plants. Therefore, these kinds of links need to receive a lower ranking if a visitor searches for plants. An uploader might even designate his daughter as a flower in this way. The uploader's profile may be utilized to extract just pertinent data pages, increasing efficiency. We examine the profile page of the person who uploaded the image for every detailed page that was pulled out by a focused crawler. Whether or not the rank of the related detail link exceeds a threshold value depends on the uploader's profile. To ascertain the uploader's interest, we go through their likes and photosteam. It is predicted that an uploader would share the most bird-related photographs if he has the most bird-related images in his favorites and photostream and if he has friends who are equally interested in this topic. By doing this, we may determine the rank based on the uploader profile and disregard links with a rank below a certain threshold. The inner profile and the inter profile are the two components that make up an uploader's profile.

Ranking from inner profile: The inner profile is used to find the uploader’s interest. The inner profile comes from media uploaded by that uploader and identifies the type of uploader. A nature lover will upload maximum photos related to nature and animal lover will upload maximum photos related to animals like cat, dog, camel, etc. inner profile rank can be calculated by similarity between topic tags and terms in profile.

Ranking from inter profile: Inter profile is related to uploader's friend's profile. In social media sharing websites, an uploader fan of a particular topic tends to be socialized with uploader's fan of similar topic. The inner profile of an uploader can be calculated with the accumulation of inner profile of his friends.

Combining inner and inter rank: Inner and inter rank are combined to make the crawling more efficient by checking whether this profile is useful for topic document.

2.7. Profile Based Document Specific Crawling

Uploader's profile gives a rough idea about uploader's interest bit sometimes we want recent data or data of particular author. In these cases, metadata like date, author and topics can be taken as criteria. These metadata can be divided into two types:

a) Descriptive metadata: these include date, size, title, type, etc.

b) Semantic metadata: these include topic, organization, title, etc.

2.8. Recommender System

When there was no recommendation system, people tended to ask their friends or expert for guidance. Today online recommendation systems provide technological methods for social recommendation process [30] where these systems are used to find whether a user will like a particular item or not and to identify top N items of user's interest.

Recommender systems [29] are used in various applications like web stores, online communities and music players. Today recommender systems are mainly associated with e-commerce sites where these systems are used to provide suggestions to customers about products and help them to buy products.

Recommender systems technology has been applied in a number of domains, such as online stores (Linden et al., 2003), movies (Herlocker et al., 1999), music (Celma, 2008), Web pages (Joachims et al., 1997), e-mail (Goldberg et al., 1992), books (Mooney and Roy, 2000), news articles (Das et al., 2007), scientific articles (Budzik and Hammond, 1999), and even jokes (Goldberg et al., 2001). There are mainly three approaches for recommendations [41]. They are content based filtering and collaborative filtering and hybrid approach.

2.8.1. Content based filtering

Items are chosen in content-based filtering based on the relationship between the user's priorities and the content. A suggested music recommendation has one such architectural [22]. Content Based Music Recommender: [22] presented a prototype for Myusic, which uses social media to recommend music to consumers. This technique uses social networking sites like Facebook to determine the consumers' preferred music. The Myusic system uses data filtered from user searches on websites like Amazon to identify user preferences and make tailored recommendations based on the user's interests. The four components that make up this platform are the crawler, extractor, profiler, and recommender. The list of accessible artists is found using a crawler and is then locally saved on a medium so that it may be utilized for recommendations in the future. The extractor retrieves the user's preferred artist. In order to do this, it establishes a connection with Facebook and determines the user's favorites by utilizing their music preferences from their profile, posts, and likes. Currently, the list of the user's favorite artists is created using this information. To make recommendations easier to utilize, this information is also added to the user's personal profile.

Profiler creates the user's profile. Whether or if users like an artist's profile or submit a song about them, that artist's weight is determined. Consider that if a user likes a post about that artist made by one of his friends, we can offer a score of

2 out of 5 and if a user posts a music connected to that artist, we can provide a score of 3 out of 5.A person can provide a score of four to the artist's Facebook profile. Additionally, a user may receive a score of five out of five if they like an artist's Facebook page and share a link to them. Based on this profile, recommendations can then be made. Using vector representations of the artist and user that are similarity measured, Recommender generates a prioritized list of artists.

2.8.2. Collaborative filtering

In collaborative filtering system items are chosen on the basis of correlation between people with similar preferences.

The targeted client receives recommendations from the system for goods or individuals that have been rated positively by other users whose ratings are comparable to the targeted user's [37][38]. This user's profile is used for recommendations, and in order to create it, the user has to log into the system. A user's profile can be shown as a vector of products with ratings next to each item. As the user ranks the goods, the vector is continuously updated [39]. The rating may be on a broad scale, or it may have a Boolean value based on the user's likes and dislikes of the product.

This process of collaborative recommendation consists of mainly two phases:

- Firstly, it searches for the users similar to target user for which recommendation is made. In the traditional collaborative filtering systems similarity between different users is estimated on the basis of user's personal ratings [39][40]. In this rating assigned by the targeted user is compared with the rating provided by other users to similar items and the users with similar ratings are find out. There are also other ways to find out similarities.
- In this phase the items which were highly graded by the users identified in first phase are recommended. One simple method of recommendation is to recommend items with higher ratings, or the items bought frequently.

Advantages

- This method of recommendation is very effective although it is the oldest method.
- In this method it is not needed to represent the object in the easily readable form by computer [39].

Disadvantages

- The problem is related with new users and products not rated yet [40]

Many people need to rate the product before the system is effective [38]

- Data sparseness problem which occurs when there are so many items to rate, the set of items changes frequently, and the number of customers is very small [38][40]. It creates a problem in finding users similar to targeted users.
- There may be Difficulty in spotting the unpredictable users with rare preferences and having unusual opinions about the products.
- In traditional systems, as the number of users and the items is increased, the amount of work to be done also increased. Computation is done offline because it is very complex process.

2.8.3. Hybrid approach

Many recommender systems combine collaborative and content-based methods into one method known as the hybrid approach. By using this method, the constraints of collaborative filtering and content filtering are eliminated.

Approaches to build a hybrid approach.

Combining separate recommender systems: This method applies collaborative and content-based approaches independently, then combines their predictions [42]. This can be achieved by choosing the best individual recommender system after evaluating the quality of both systems, and by integrating the ratings from each recommender system into a final recommendation system.

Adding some content-based characteristics or techniques to the collaborative approach: This approach results in the addition of some content-based techniques in collaborative approach. In this case, similarity between two users is estimated with content based profiles and items which are not commonly rated.[43] explains that with this approach sparsity related problems of a proper collaborative approach is overcome.

Adding some collaborative characteristics or techniques to the content-based approach: This approach results in the addition of some collaborative based techniques in content-based approach. Easiest method is to make a collaborative view of a set of user profiles.

[42] provides the differences between various methods of recommendation: collaborative method, content-based method, demographic, utility based and knowledge-based techniques.

2.9. Recommendation in Social Network

In this section the functional requirements of a recommendation system for social networks are discussed. The recommendation process providing the suggestions to targeted user x, should have the following elements:

- These systems should be able to monitor the user's behavior on social networks.
- Data gathering and Data preparation
- Should be able to calculate the ranking list for member x and this list should be recalculated frequently.
- Finding k–nearest neighbors for user x
- Providing recommendation to user x
- Weight based on the user's feedback should be recalculated periodically.

These are the identical actions for every member of the community. It takes longer for some of these processes than for others. This indicates that, as figure 4 illustrates, the amount of time required to complete a given work might be split across online and offline activities.

As shown in the figure, monitoring behavior of users and the delivery of recommendation should be done online. All other tasks are done offline due to their calculation complexity.

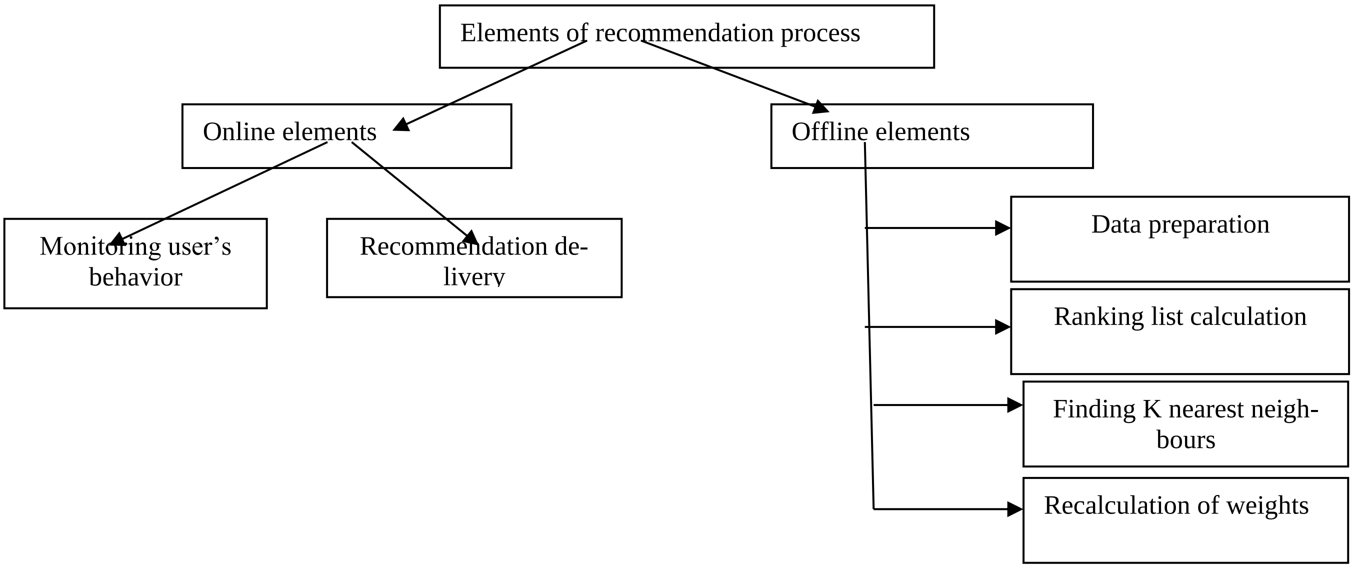


Figure 4. Elements of recommendation process

2.10. Query Recommendation in Search Engines

Search engines help the users in finding data of their interest. They must have a mechanism through which they can find the user's interest with respect to fired queries and can optimize the results according to user's interest. For this purpose, we need to track the user's activities on search engines. This goal can be achieved with the help of query logs maintained by the search engines. These query logs analyze how a search engine is used and what are user's interests. Use web log mining to improve search engine's performance by utilizing the mined information.

Components mainly used in query recommender system are:

2.10.1. Query Logs

Query Logs record user activities on search results and therefore are information repositories. The performance of search engines is improved by mining these logs. Query logs generally have information including user's queries, Clicked URL corresponding to a particular query and information related to browsing activities.

The typical query logs in literature [5] of search engines include User IDs, Query issued by user, URL clicked by the user, Rank of the URL clicked by user and the time at which that query was submitted.

Mining Query Log on Click Graph: Applications for mining query logs include query to query similarity, query clustering, query recommendation, and more.

The two most crucial responsibilities for comparing various mining models are as follows:

(a) Basic query-to-query similarity analysis is one job. This may be used to gauge how well query representation models perform.

(b) The popular query recommendation job employs a graph-based random walk model to identify semantically similar inquiries for a given query.

2.10.2. Query similarity analyzer

Query Similarity Based on Keywords: If two queries have same or similar keywords, it denotes that both need same or similar information. The content similarity is measured on the basis of the number of common keywords in both queries to the union of keywords in both queries. The queries having a higher ratio is highly similar.

Query similarity based on user feedback: If two queries result in the same or similar documents, then those queries are considered to be similar. In this bipartite graph is used to measure similarity. Here similarity analyzer first creates a bipartite graph with one set of vertices representing the queries and the other representing the documents.

In this bipartite graph, a query vertex is joined with document vertex if document is clicked or accessed by a user corresponding to that query. Similarity value between two queries p and q sharing common document d is ratio of the total number of distinct clicks on common document d with respect to both queries p and q and the total number of distinct clicks on all the documents accessed for both queries p and q.

Combined similarity measure: With the help of similarity based on keywords, queries with similar composition can be grouped together. With the help of similarity based on user feedback, we can find similarity based on the user's judgments.

The combined similarity between both queries T1 and T2 can be estimated as follows:

$$sim_{combined}(T1, T2) = \alpha . sim_{tags}(T1, T2) + (1 - \alpha). sim_{urls}(T1, T2) \quad (1)$$

where 0≤ α≤1 and α is a constant. The determination of α's value might be based on the analysis and weight assigned to each element. We have chosen α to be 0.7 in our study as we are placing greater importance on labeling data. There is a greater likelihood of comparable searches if the query themes are more similar.

3. Proposed Work

This architecture has mainly two storage systems query logs and query clustering database and the following functional components:

- Page classifier
- Focused crawler
- Topic discovery system
- Profile based crawler
- Similarity Analyzer
- Favored query finder
- Query Recommender

The proposed architecture for the query recommendation is shown in figure 5 recommended result.

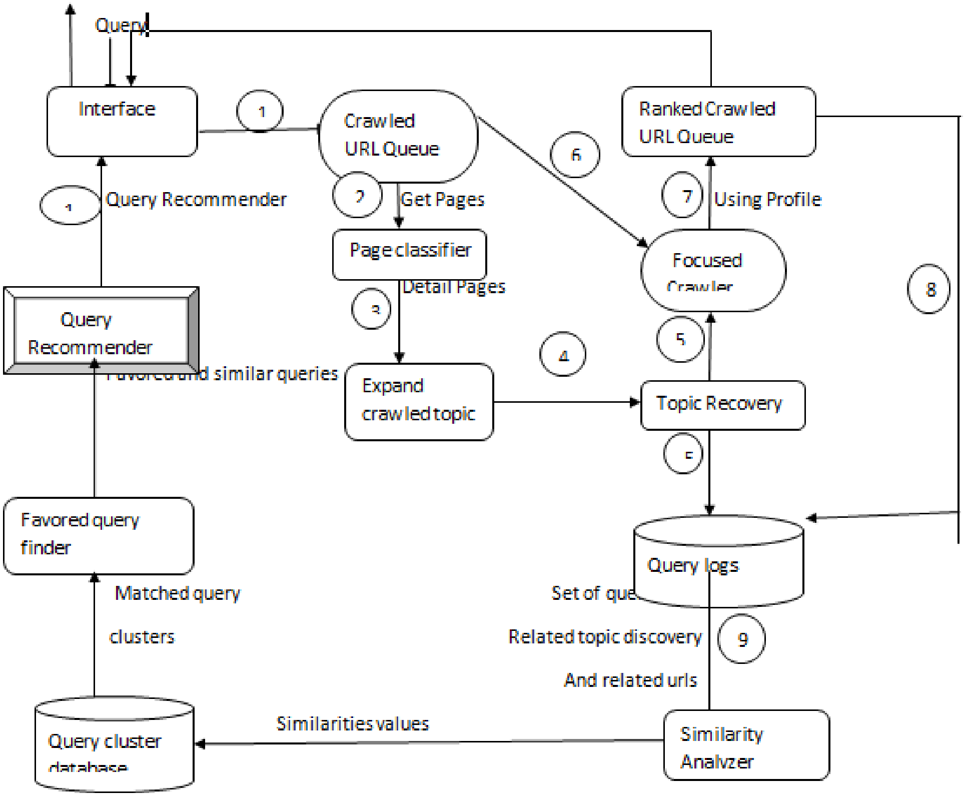


Figure 5. Query Recommendation Architecture

Different modules of proposed architecture are explained below:

3.1. Page classifier

Page classifier is used to classify the list pages, detail pages and profile pages.

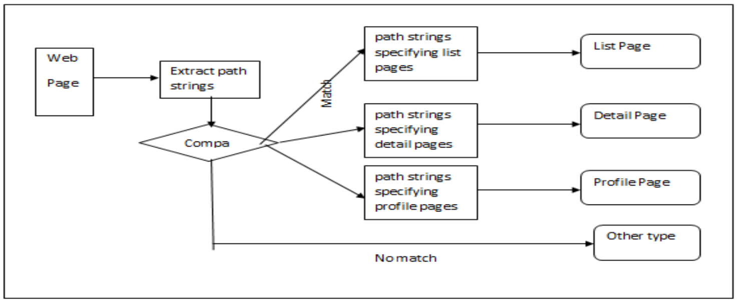

Web
Page
Extract path strings
Compa
Match
path strings specifying list pages
path strings specifying detail pages
path strings specifying profile pages
List Page
Detail Page
Profile Page
Other type
No match

Figure 6. Page classifier

Page classification is needed because different web pages on these web sites have different structures. For topic discovery we need to collect only tags which are present on detail page and for profile-based crawling we need to find out profile pages. Page classifier uses DOM path string for page classification. Web pages of similar type have same path strings. A unique group of path strings identifies a particular type of web page. All similar types of web pages has some common path strings. We use the same property for page classification. It is analyzed that only one path string is enough to classify a particular web page on photo sharing web site flickr. Every list page has a route String that corresponds to the name of the uploader. Path strings on detail pages match to tags. Similar to this, every profile page has a path string for each favorite. The algorithm for classifying pages is illustrated in Diagram 3.2.

3.2. Focused crawler

Focused crawler is the crawler retrieving the pages only related to a particular topic. When a user searches for flowers on flickr then it should return link only related to flowers. Each image on these sites has some tagged data and that data is used for this purpose. Images which are tagged as flowers are returned.

Problem with focused crawler: As discussed above, the results are returned on the basis of the tagged data. Tags are the metadata present with each image and provided either by the uploader of that image or an authorized user. Now an uploader can upload his daughter's photo tagged as flower. When a user searches for flowers, he or she gets this daughter's image also. Since this image is irrelevant, it should be given lower rank. To solve this problem, we use the uploader's profile to estimate whether the image uploaded by the corresponding uploader is of targeted field or not.

3.3. Co-tagging Topic Discovery

Need of topic discovery

Topic discovery is very important in this system. Suppose a user wants to search for animals. We can't feed the crawler with such narrow topic only animals. If we do so, then it retrieves the images having animals as its tagged data. If an image does not have animals as its tagged data but cat, then also it will not return that image. But a user wants these results also. So these systems should be able to extend this crawling topic. These systems should give a method which will automatically extend this animal crawler topic to animals, animals, cat, dog, tiger, etc.

Steps in topic discovery process

Step 1: feed the crawler with targeted crawler topic.
Step 2: Now the focused crawler returns the links to related detail pages.
Step 3: each detail page has images with the tagged data. Now all the detail pages are analyzed for finding the co-tagged da-

ta. This co-tagged data is found on the basis of voting scheme discussed below.
Step 4: finally tags t1, t2,………, tn representing crawler tags are returned and crawling is done for all these topics.

Voting scheme for co-tagging: We determine the crawler subject by a voting-like processing technique. Votes are tallied for tag T1, say, using all of the detail pages. When subject tag T and T1 appear in a photo at the same time, one vote for T1 is taken into account. Only when the total number of such votes for tag T1 exceeds a certain threshold value is this tag T1 included in the crawling subject tags. In order to do this, we multiply the total number of images tagged by T and T1 by the ratio of images co-tagged by both tags T and T1. T1 is added to the crawling topic tags if the ratio value exceeds the predetermined threshold value; else, it is not. All of the crawling subject tags are located in this manner.

3.4. Profile Based Focused Crawler

This module returns the results corresponding to the user's query. Results returned by this module are the detail pages links corresponding to user's query. In this crawler the result of co-tagging topic discovery is fed as the crawling topic tags. Focused crawler retrieves the results corresponding to topic tags. But all of these links are not relevant as discussed in the limitation of the focused crawler. To refine these results, we use the uploader's profile. Uploader's profile has two types of properties one is inner property and other is inter property. Each link is given rank according to these properties.

Ranking according to inner profile: Inner profile comes from uploader's own photostreams, sets and favorites. From an uploader's profile we can estimate the type of image an uploader generally uploads. An animal lover will generally upload images related to animals. To find the uploader's interest we search his profile. If crawling topic terms frequently occurs in uploader's profile say photo stream, sets and favorites then it is estimated that the uploader is interested in required field and links related to that uploader are given higher ranks.

Ranking from the inter profile: Inner profile only provides the uploader's individual properties. These sites also allow users to socialize with friends. We can use uploader's social contacts to find the inter properties. Generally, it is found that an animal lover will be friends with other animal lovers on these sites. To find the inter profile of an uploader, we check the inner profile of uploader's contact and finally accumulate all these to find uploader's inter profile.

Finally, the rank is provided to a link using these both properties and the links having higher links are displayed first.

3.5. Query Logs

We must identify other fired queries that are comparable in order to suggest questions. To do this, we must keep all requests in one specific spot. We create query logs in order to accomplish this goal. Four key components are stored in query logs in relation to a query. The following four items are listed:

- IP address firing the query
- Fired query
- Co-tagged topic discovery related to that subject
- Resulting URLs matching to that query

Every query that is fired, together with its crawling subject phrases, IP address that fired it, and the resultant URLs that relate to that specific query, are stored in these query logs. This data is utilized to make additional recommendations.

3.6. Query similarity analyzer

The two guiding ideas of this similarity analyzer are the number of matched URLs in the final crawled URLs and similarity based on expanded crawling subject phrases. Query similarity analyzer links co-tagged data and resultant URLs to determine similarity by analyzing query logs.

Similarity based on extended crawler topic

In this instance, crawling topic phrases are used to determine similarities between the different requests. Two searches after co-tagging indicate the same or comparable type of information if they contain the same or similar tags to be crawled. The co-tagging topic keywords of queries lotus and flower, for instance, are comparable. The following formula may be used to determine how similar these two questions, or themes, T1 and T2, are to one another.

$$Sim_{tags}(T1, T2) = \frac{|tags(T1, T2)|}{|tags(T1)\ \cup\ tags(T2)|} \tag{2}$$

where tags (T1, T2) are the sets of common tags in the expanded topics of both queries, and tags (T1) and tags (T2) are the sets of tags in the extended crawled topic corresponding to T1 and T2, respectively.

Similarity based on the URLs retrieved

Any query's output is a list of ranked URLs, as was covered in profile-based focused crawling.The queries that yield a list of same or similar URLs are identical or comparable. Here, we apply the same idea to determine the questions' commonalities. By utilizing the provided formula to detect the similarities between the obtained URLs, one may determine the similarities between these searches.

$$Sim_{urls}(T1, T2) = \frac{|urls(T1, T2)|}{|urls(T1)\ \cup\ urls(T2)|} \tag{3}$$

where urls (T1, T2) is the set of common urls in the resulted urls list corresponding to both queries, and urls (T1) and urls (T2) are the sets of resulted urls belonging to T1 and T2, respectively.

Combined similarity measurement

Both similarity based on the expanded crawling topic and similarity based on the urls obtained are used to assess the combined similarity between the searches. It is easy to determine the combined similarity value by applying these two similarities in a linear fashion.

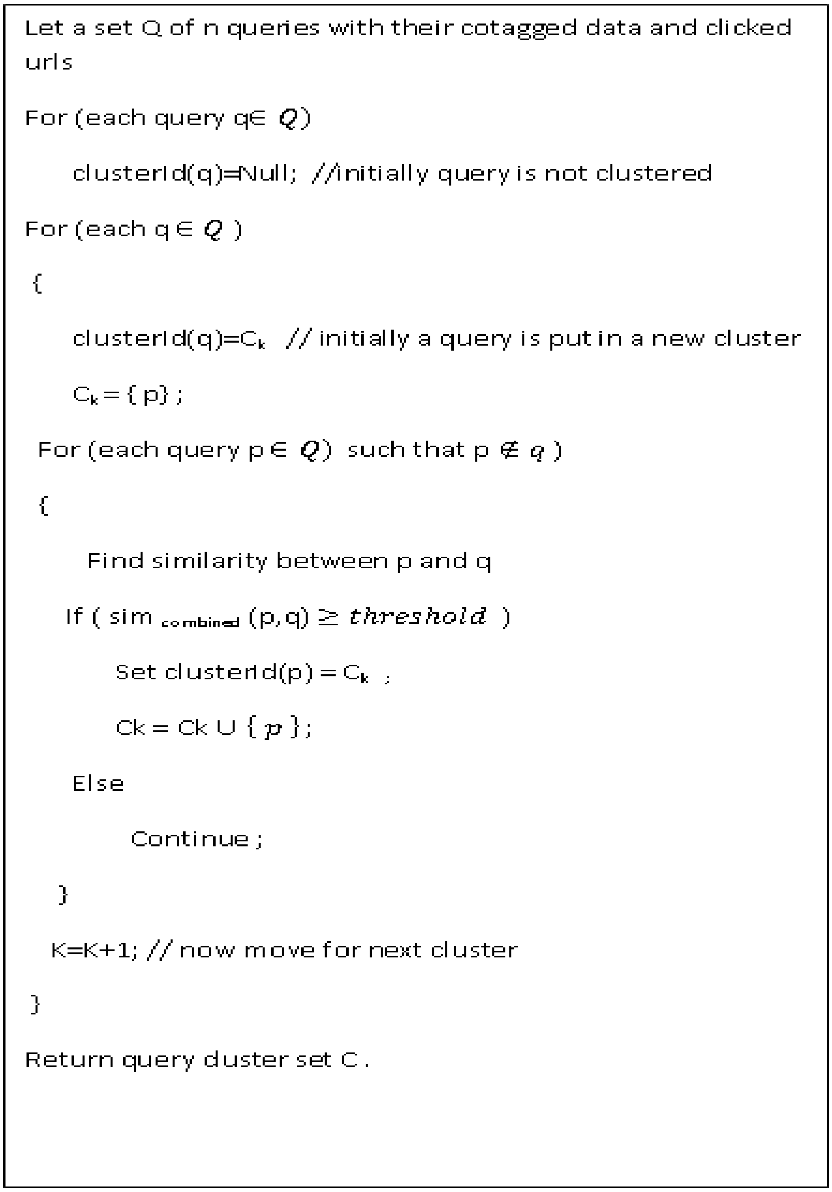
$$sim_{combined}(T1, T2) = \alpha. sim_{tags}(T1, T2) + (1 - \alpha). sim_{urls}(T1, T2) \tag{4}$$

where 0≤ α≤1 and α is a constant. Based on an examination and consideration of each factor's relevance, the value of α may be determined. We have chosen α to be 0.7 in our study because we place greater importance on tagging data. There is a greater likelihood of comparable searches if the query themes are more similar.

3.7. Query Clustering Tool

Using this tool, related searches are grouped according to similarity values. Two queries are deemed comparable and placed in the same cluster if their similarity values are above a certain threshold. At first, no query is thought to belong to a cluster. Every question is compared to every other query, whether it is categorized or not. Until every query is assigned to a cluster, this procedure is repeated.

Clustering algorithm



```
Let a set Q of n queries with their cotagged data and clicked urls
For (each query q$\in$ $Q$)
    clusterId(q)=Null;  //initially query is not clustered
For (each q $\in$ $Q$ )
{
    clusterId(q)=$C_k$  // initially a query is put in a new cluster
    $C_k$ = { p} ;
 For (each query p $\in$ $Q$)  such that p $\notin$ $q$ )
 {
      Find similarity between p and q
    If ( sim $_{combined}$ (p,q) $\geq$ $threshold$  )
        Set clusterId(p) = $C_k$  ;
        Ck = Ck $\cup$ { $p$ };
    Else
          Continue ;
  }
  K=K+1; // now move for next cluster
}
Return query duster set C .
```

For clustering, we employ an incremental clustering technique. The incremental approach is used because query logs are dynamic and constantly evolving as more and more people submit searches.

3.8. Favored Query finder

The next stage after query clustering is to identify the preferred queries for every cluster. The majority of users' searches are often the ones that are marked as favorites. IP addresses that are used to fire queries can be used to identify preferred queries. Here, we will use the following equation to get the query weight in a certain cluster.

$$wt_{query} = \frac{Number\ of\ ip\ addresses\ firing\ query}{total\ number\ of\ ip\ addresses\ in\ that\ particular\ cluster} \quad (5)$$

A query is considered preferred if its weight exceeds a certain threshold. Once more, <Cluster Id, favored query> pairs are used to store this result in the cluster database.

3.9. Query recommender

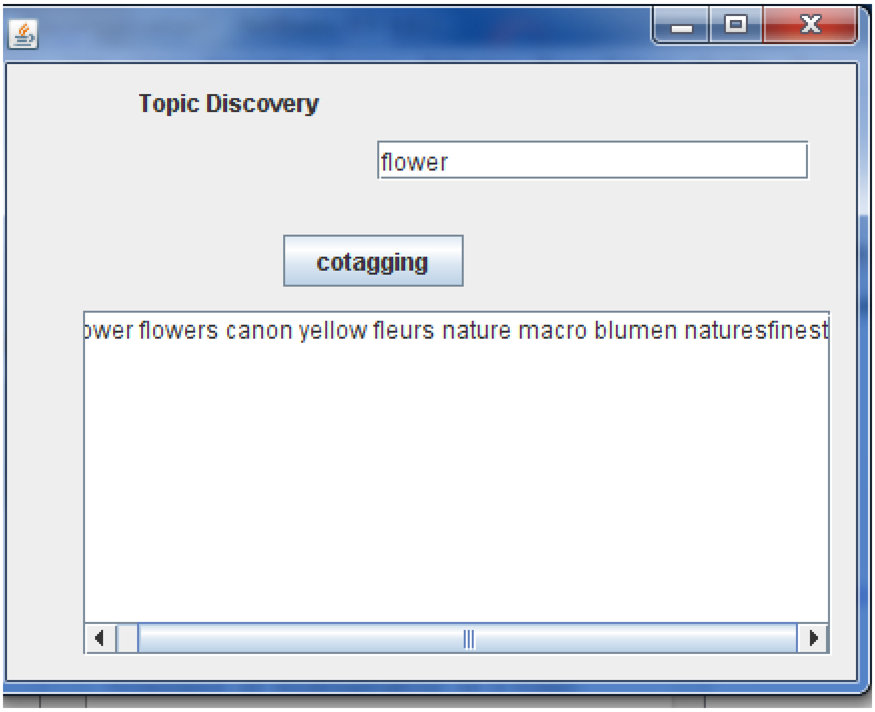
This suggests questions to users based on their intended inquiry. The suggested queries are preferred queries that are part of the same cluster and are comparable to the query that the user entered. With the aid of these suggested inquiries, customers may enhance their search and get assistance with it.

4. Results and Analysis

4.1. Results

Our methods for query suggestion provide us with rather decent outcomes. We have examined the outcomes for several searches in our experiment. Even though there is much room for improvement, our suggested architecture accomplishes a lot. Results for the three main components of our architecture will be displayed here:

Finding topics to tag
Profile-driven, targeted crawling
Question suggestion

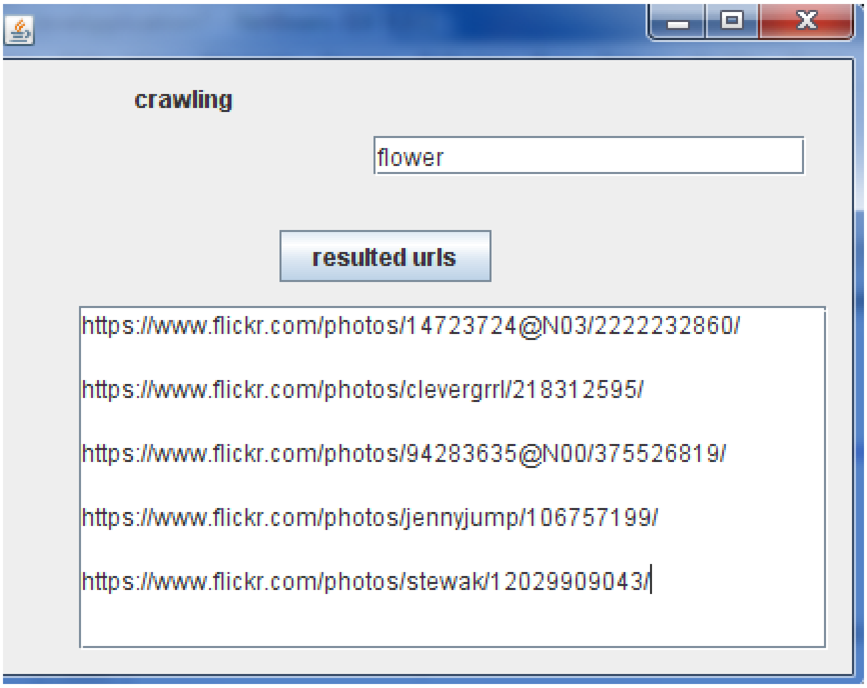


flower

ɔwer flowers canon yellow fleurs nature macro blumen naturesfinest

Figure 7. Results for co-tagging

The topic flower's co-tagging result is shown in Fig. 7. The topic categories for flowers, such as flower, flowers, canon, yellow, fluers, macro, blumen, nature, and nature best, come from our architecture.



flower

https://www.flickr.com/photos/14723724@N03/2222232860/

https://www.flickr.com/photos/clevergrrl/218312595/

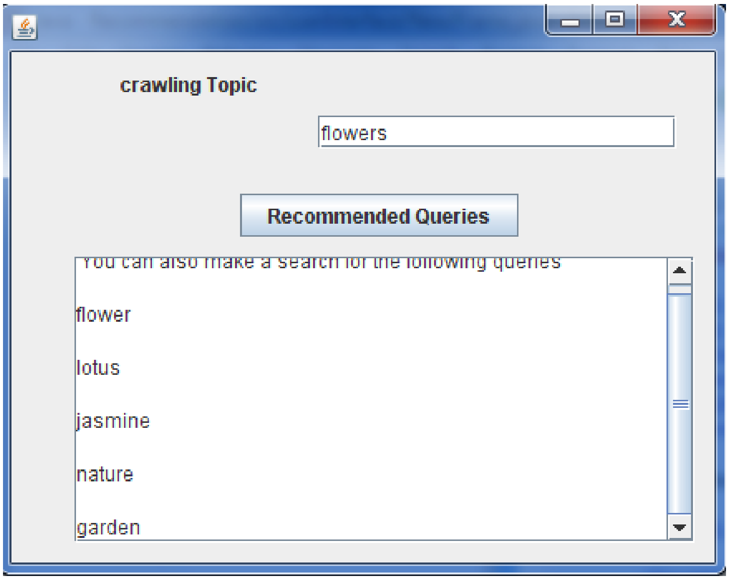
https://www.flickr.com/photos/94283635@N00/375526819/

https://www.flickr.com/photos/jennyjump/106757199/

https://www.flickr.com/photos/stewak/12029909043/

Figure 8. Result after profile based focused crawling

The outcome of profile-based targeted crawling for the topic flower is shown in Fig. 9. This crawling is connected to the co-tagging subject tags in addition to the crawling topic.



flowers

You can also make a search for the following queries

flower

lotus

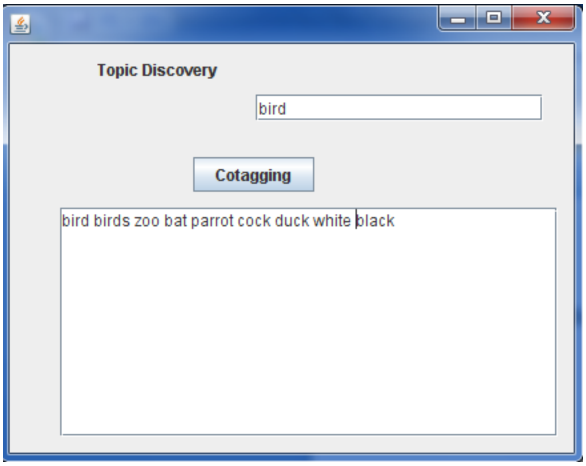
jasmine

nature

garden

Figure 9. results for the Query Recommendation

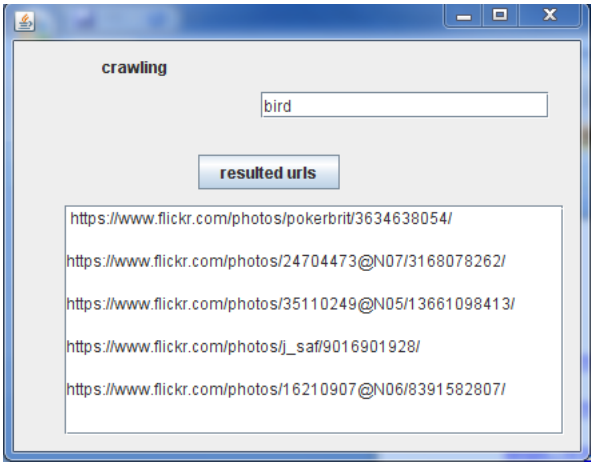
The outcomes of the crawling topic floral inquiry recommendation are displayed in Figure 10. The user may also look for similar searches like flower, lotus, jasmine, nature, and garden, as this result demonstrates.



bird

bird birds zoo bat parrot cock duck white black

Figure 10. results of co-tagging for bird



bird

https://www.flickr.com/photos/pokerbrit/3634638054/

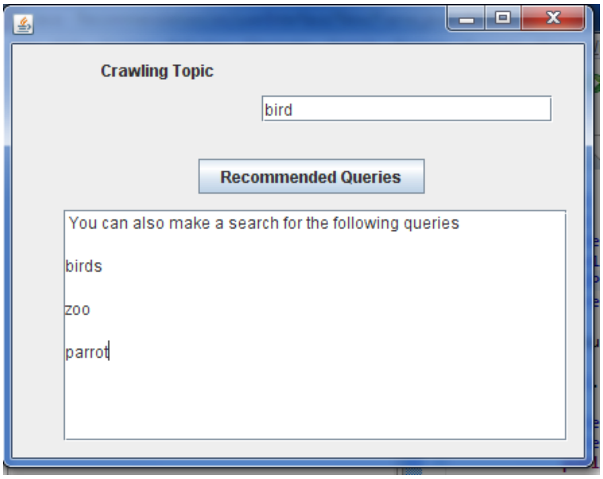
https://www.flickr.com/photos/24704473@N07/3168078262/

https://www.flickr.com/photos/35110249@N05/13661098413/

https://www.flickr.com/photos/j_saf/9016901928/

https://www.flickr.com/photos/16210907@N06/8391582807/

Figure 11. results of profile based focused crawling for bird



bird

You can also make a search for the following queries

birds

zoo

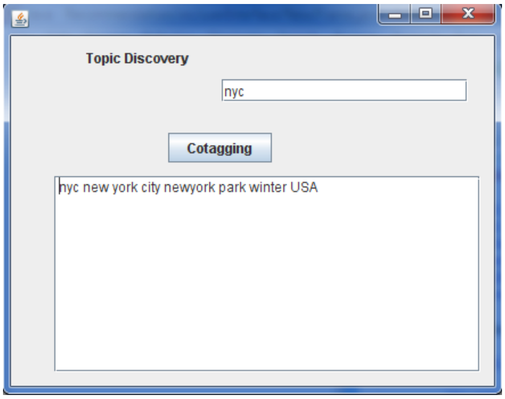
parrot

Figure 12. Results of Query Recommendation

Fig 10 shows the result for co-tagging for topic query bird. After co-tagging topic tag birds extends to topic tags as bird, birds, zoo, white, black, bat, parrot, cock, and duck. Fig. 11 shows the result for profile based focused crawling for topic tag bird. As discussed in the case of flowers, this result is for tags after co-tagging. Fig. 12 shows the result for recommendation for query bird. In this, it recommends the queries birds, Zoo, parrot. Now users can also search for these related queries.

4.2. Analysis of Results

For some questions, our suggested architecture provides correct responses. The findings for the bird and flowers in the preceding figures are good, however the co-tagging results for the search for NYC are displayed in the figure.



nyc

```
nyc new york city newyork park winter USA
```

Figure 13. co-tagging results for nyc

The result indicates that nyc causes the topic tags to be tagged as nyc, new york, city, park, winter, USA. We may claim that our architecture has to be improved because these tags don't make sense.

5. Conclusion and Future Scope

5.1. Conclusion

When it comes to providing query recommendations for various questions, this query recommendation system for social media sharing platforms outperforms its competitors. During crawling, this technology gives consumers access to more relevant media assets and suggests related and highly desired inquiries to them. Only favorite queries are recommended since they belong to the same cluster as the targeted query and are often fired by users. The user finds this tip useful in their search.

The important points concluded from this thesis are:

- Co-tagging topic discovery automatically extends the crawling topic which provides the results highly closer to user's interest.
- Uploader's profile is very efficient way to rank the crawling's results. Inner and Inter properties of uploader play an important role in ranking.
- The similarity in co-tagging topic tags and resulting URLs between different queries is an efficient measurement for query recommendation.

From the above discussion, it is clear that this query recommendation system is very efficient providing the users options for searching.

References

[1.] Z. Zhang and O. Nasraoui, "Profile-Based Focused Crawler for Social Media-Sharing Websites," 2008 20th IEEE International Conference on Tools with Artificial Intelligence, Dayton, OH, USA, 2008, pp. 317-324, doi: 10.1109/ICTAI.2008.119.

[2.] R. Sinha and K. Swearingen, "Comparing recommendations made by online systems and friends," *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*, vol. 106, 2001.

[3.] T. Berners-Lee, *Information Management: A Proposal. CERN. World Wide Web Consor- tium (W3C)*. 1989.

[4.] "Architecture of the world wide web, volume one," Www.w3.org. [Online]. Available: http://www.w3.org/TR/webarch. [Accessed: 12-Nov-2021]..

[5.] T. Berners-Lee, R. T. Fielding, and H. Frystyk, *RFC 1945: Hypertext Transfer Protocol - HTTP/1.0*. 1996.

[6.] T. Berners-Lee, L. Masinter, and M. McChahill, "RFC 1738: Univorm resource locators (URL)" 1994.

[7.] T. Bray, J. Paoli, C. M. Sperberg-Mcqueen, E. Maler, and F. Yergeau, *Extensible Markup Language (XML) 1.0 (Fourth Edition) - Origin and Goals*. World Wide Web Consortium., http://www.w3.org/TR/2006/REC-xml-20060816, 2006.

[8.] D. W. Connolly and L. Masinter., "RFC 2854: The 'text/html' Media Type" 2000.

[9.] J. L. Herlocker, *Understanding and Improving Automated Collaborative Filtering Systems*.

[10.] Archive.org. [Online]. Available: http://archive.org. [Accessed: 12-Nov-2021].

[11.] GigaAlert, "Giga Alert - Professional Web Alerts," Gigaalert.com. [Online]. Available: http://www.gigaalert.com. [Accessed: 12-Nov-2021].

[12.] S. Chakrabarti, M. van den Berg, and B. Dom, "Focused crawling: a new approach to topic-specific Web resource discovery," *Comput. Netw.*, vol. 31, no. 11–16, pp. 1623–1640, 1999.

[13.] S. Chakrabarti, B. Dom, and P. Indyk, "Enhanced hypertext categorization using hyperlinks," in *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, 1998.

[14.] F. Menczer, G. Pant, and P. Srinivasan, "Topical web crawlers: Evaluating adaptive algorithms," *ACM Trans. Inter. Tech*, vol. 4, no. 4, pp. 378–419, 2004.

[15.] G. Pant and P. Srinivasan, "Learning to crawl: Comparing classification schemes," *ACM Trans. Inf. Syst*, vol. 23, no. 4, pp. 430–462, 2005.

[16.] C. C. Aggarwal, F. Al-Garawi, and P. S. Yu, "Intelligent crawling on the World Wide Web with arbitrary predicates," in *Proceedings of the 10th international conference on World Wide Web*, 2001.

[17.] C. C. Aggarwal, F. Al-Garawi, and P. S. Yu, "On the design of a learning crawler for topical resource discovery," *ACM Trans. Inf. Syst.*, vol. 19, no. 3, pp. 286–309, 2001.

[18.] M. Diligenti, F. Coetzee, S. Lawrence, C. L. Giles, and M. Gori, "Focused crawling using context graphs," in *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, San Francisco, CA, USA, 2000, pp. 527–534.

[19.] C. C. Hsu and F. Wu. "Topic-specific crawling on the web with the measurements of the relevancy context graph", In! Syst., vol. 31(4) pp. 232-246, 2006.

[20.] M. L. A. Vidal, A. S. Silva, and E. S. De Moura, "Jo• Structure-driven crawler generation by example," in *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006, pp. 292–299.

[21.] G. J. Qi, C. Aggarwal, Q. Tian, H. Ji, and T. S. Huang, "Exploring Context and Content Links in Social Media: A Latent Space Method "," *IEEE Transactions on Pattern Recognition and Machine Intelligence*.

[22.] "Cataldo Musto, Fedelucio Narducci, Giovanni Semeraro, Pasquale Lops, and Marco de Gemmis. IIR," *Content-based Music Recommender System based on eVSM and Social Media*, vol. 964, pp. 65–72, 2013.

[23.] N. Mishra, S. Silakari, "Image Mining in the Context of Content Based Image Retrieval: A Perspective," *IJCSI International Journal of Computer Science Issues*, vol. 9, no. 3, pp. 69-76, 2012.

[24.] S.-H. Hung, P.-H. Chen, J.-S. Hong, S. Cruz-Lara, and S. Cruz, "Context-based image retrieval: A case study in background image access for Multimedia presentations," Hal.science. "IADIS International Conference WWW/Internet, 2007. [Online]. Available: https://inria.hal.science/inria-00192463/PDF/IADIS-context-image-final.pdf. [Accessed: 10-Nov-2020].

[25.] Z. Zhang, "Roelof Van Zwol "Exploiting Tags and Social Profiles to Improve Focused Crawling," in *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, 2009.

[26.] H. Christopher and N. Brooks, *Improved annotation of the blogosphere via autotagging and hierarchical clustering*. In WWW, 2006.

[27.] S. A. Golder and B. A. Huberman, "Usage patterns of collaborative tagging systems," *J. Inf. Sci.*, vol. 32, no. 2, pp. 198–208, 2006.

[28.] P. Heymann, G. Koutrika, and H. Garcia-Molina, "Can social bookmarking improve web search?," in *Proceedings of the international conference on Web search and web data mining - WSDM '08*, 2008.

[29.] P. Resnick and H. R. Varian, "Recommender systems," Commun. ACM, vol. 40, no. 3, pp. 56–58, 1997.

[30.] K. Swearingen and S. Rashmi, “Interaction design for recommender systems,” in *Designing Interactive Systems*, citeseer.ist.psu.edu/swearingen02interaction.html, 2002.

[31.] Y. Zhang, J. X. Yu, and J. Hou, *Web Communities: Analysis and Construction*. Berlin Hei-delberg: Springer, 2006.

[32.] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2007.

[33.] B. Mobasher, et al., “Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization”, *Data Mining and Knowledge Discovery*, 2002.

[34.] J. Hou and Y. Zhang, “Utilizing Hyperlink Transitivity to Improve Web Page Clustering,” in *Proceedings of the 14th Australasian Database Conferences (ADC2003)*, 2003.

[35.] E. Han, “Hypergraph Based Clustering in High-Dimensional Data Sets: A Summary of Results,” *IEEE Data Engineering Bulletin*, 1998.

[36.] J. Xiao, Y. Zhang, X. Jia, and T. Li, “Measuring similarity of interests for clustering Web-users,” in *Proceedings 12th Australasian Database Conference. ADC 2001*, 2002.

[37.] S. H. Ha, “Helping online customers decide through Web personalization,” *IEEE Intell. Syst.*, vol. 17, no. 6, pp. 34–43, 2002.

[38.] P. Kazienko and M. Kiewra, “Personalized Recommendation of Web Pages,” in*telligent Technologies for Inconsistent Knowledge Processing*, Adelaide, South Australia, 2004, pp. 163–183.

[39.] R. Burke, “Hybrid Recommender Systems: Survey and Experiments,” *User Modeling and User Adapted Interaction*, vol. 12, no. 4, pp. 331–370, 2002.

[40.] M. Montaner, B. Lopez, and J. L. De La Rosa, “A Taxonomy of Recommender Agents on the Internet,” *Artificial Intelligence Review*, vol. 19, pp. 285–330, 2003.

[41.] G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions,” *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, 2005.

[42.] P. Paulson and A. Tzanavari, “Combining collaborative and content-based filtering using conceptual graphs,” in *Lecture Notes in Computer Science*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 168–185.

[43.] J. Michael, “A Framework for Collaborative, Content-based and Demographic Filtering,” *Artificial Intelligence Review*, vol. 13, no. 5, pp. 393–408, 1999.