Search Engine Query Recommendation
Using SNA over Query Logs with User Profiles

Lule Ahmedi and Dardan Shabani
Faculty of Electrical and Computer Engineering, University of Prishtina, Prishtina, Republic of Kosovo
lule.ahmedi@uni-pr.edu, dardan.shabani@uni-pr.edu

Keywords: Query Recommendation, Social Network Analysis, Information Retrieval.

Abstract: Recommending the adequate query in search engine for a specific user on the web is still a challenge even for recommender systems today with social networks incorporated. In this paper we present a query recommender that in addition to relying on similarity of the actual query posted by current user to queries in a query log in search engine, it also bases on social network analysis (SNA) to first find most similar users to the current user based on their profiles, and then recommend their most similar queries to current user. Calculation of the similarity of users follows an existing approach for Points of Interest (POIs) recommendation, which applies certain SNA ranking algorithms over concurrent users based on their social profiles in the login session.

1 INTRODUCTION

Most of recommender systems use collaborative filtering as key technique to find similar items or users in order to recommend items that they liked (searched) to their similar users. Nowadays search engines are characterized with opportunity to search quite easy and unambiguous, entering some keywords for the things user is looking for and list of queries will be shown from earlier search. However, not always what users require is listed in the top questionnaires listed. Very often it happens that a specific user becomes the first one who asks for what makes a new query for search engine. With the rapid growth of users in social networks, recommender systems are already integrated in every search query of social network's users. Social network impact is influencing many fields, so we decided to embrace the usage of social network analysis (SNA) as matter of fact that computing based in SNA is gaining in popularity when dealing with computational problems in general (Wasserman & Faust, 1994). Usage of SNA for Points of Interest (POIs) recommendation in our previous work (Ahmedi, et al., 2012) motivated us to involve usage of SNA to another domain, that of query recommendation. Most known search engines are adding social element to their core process, like Google[1] with social network Google[2] or Facebook[3]. Regarding these two indicators we decided to try a novel approach, making query recommendation based on SNA, specifically calculating similarity of personal attributes of users in social network to find most similar users and then recommend their queries to current user. Our method

[1] http://www.google.com
[2] http://plus.google.com
[3] http://www.facebook.com/places

for query recommendation consist of four steps: matrix similarity generation, query classification, concurrent users ranking, and finally recommendation of most similar queries using Jaccard method.

The rest of this paper is organized as follows. Section II discusses related work. Our approach is introduced in Section III. Section IV discusses evalution of developed algorithm for this approach.

2 RELATED WORK

In **different application domains**, a number of diverse **social network-based recommendation** approaches have been proposed in recent years to exploit the user generated contents available in the Social Web, such as social network data, tagging, and ratings (He & Chu, 2010) (Konstas, et al., 2009). Authors in (Carrer-Neto, et al., 2012) prove that the combination of social and collaborative algorithms into hybrid recommendation approaches overcomes this limitation in coverage inherited by social algorithms, benefiting in the same time from the accuracy of social-based recommendations not sufficiently supported by collaborative filtering methods. Similarly, the work introduced in (He & Chu, 2010) show that the collaborative recommendation system benefits from the social annotations and friendships established among users, items and tags. Only approaches presented in (Kang, et al., 2013) (Sohn, et al., 2013) use **degree centrality as an SNA measurement** along content-based filtering with FOAF (Friend of a Friend) ontology to compute centrality of each tag, respectively degree of importance of the particular user, and that way **recommend content**.

In (Shokouhi, 2013), a personalized **auto-completion** ranker is presented which takes into consideration **demographic-based features**, i.e., age, gender and location extracted from Microsoft Live profiles of users when searching via Bing. Results on the effectiveness of the ranker before and after personalization (re-ranking) show that demographic features significantly improve ranking when compared to the (no-reranking) baseline. Utilizing **user-specific data** for improved query suggestion by re-ranking the original results obtained by traditional ranking approaches is not new and has been approached by several studies already. Authors in (Wu, et al., 2015) employ user generated ratings and comments of books in Amazon as helpful metadata when suggesting social books while searching. Further in (Cheng & Cantú-Paz, 2010), a framework for the personalization of click models in sponsored search is presented which bases on **user-specific and demographic-based** features that reflect the click behavior of individuals and groups.

To the best of our knowledge, none of these existing systems considers users acting as nodes in a unimodal graph and their analysis with SNA techniques in a collaborative filtering (CF) approach to recommend query to a given user.

3 OUR APPROACH

Our SNA-based approach of query recommendation takes into account some personal attributes of users, like home city and gender, as well as their query topic or categories (e.g., politics, or sports). Social network analysis (SNA) metrics are applied over the generated uni-modal user-user network in order to generate the similarity matrix.

3.1 System Architecture

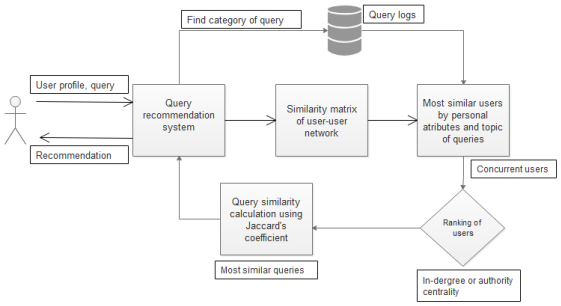

Figure 1: System architecture.

In Figure 1, the architecture of our proposed SNA-based system of query recommendation system is depicted. At the input, the system is supplied with the following type of data: the user's social profile data (e.g., its gender, and home city) and the query posted by the user. Based on input data, a similarity matrix is generated which serves to find the most similar user to the current user. After this step, if there is more than one concurrent user, ranking of users using SNA metrics, either degree or authority centrality is next performed. Final step is searching in query log for queries with most similar keywords to those submitted by concurrent users. Regarding query of current user filtering of queries is made using Jaccard similarity coefficient (Phillips, 2013). Two datasets

have been used in our proposed system. First dataset contains data from AOL search engine during three months of 2006. It consists of data about the user id in anonym form such as AnnonID (which expected to be replaced by real User ID in a future), the posted query itself, as well as the query time field and the rank field. Second dataset comprises of data gathered from Text Retrieval Conference (TREC), published during 2001-2014. Web queries retrieved from TREC dataset contain topic of the query along with the co-clicked query, the actual query, and the clicked URL. Data from two datasets have been merged into a single collection using the matching keyword criteria. From AOL dataset one of six available user's collection of queries have been used in our scenario, it contained 3013956 queries, while TREC dataset contains 5980 queries belonging to 350 distinct topics. Topics from TREC dataset have been further categorized into 8 categories, according to Google Trend Search for a better grouping purposes and due to inappropriate grouping of topics from AOL datasets. For instance some of topics from AOL dataset were: hunger, Chevrolet Trucks and deer, so it was necessary to merge these topics (queries) in one of eight categories (Lifestyle, Travel & Leisure and Nature & Science). As the result of merging the respective AOL and TREC input datasets using keyword matching, three groups of queries have been generated. First group contains queries that are matched 100% (12713 queries) from both datasets, second group contains queries matched more than 50% (56981 queries), and the last group queries matched 30% (141240 queries), always regarding keyword matching.

3.2 Modelling Similarity Matrix

Table 1: An example of user similarity calculation.

User	Gender	Home city	Topic of queries	Weight
U_1	M	London	P,S,O,E	
U_2	M	NewYork	S,N	
U_1,U_2	1	0	1	2

Similarity matrix (Algorithm A1) is comprised of all users collected from dataset. A matrix M_{ij} is a matrix whose dimensions describe a user (i) and another user (j) different form (i). Each element in M_{ij} means that i-th user and j-th are similar if value is not 0 (also when comparing a user with himself), which means they have common gender, home city or searched for queries with same topics (categories). Similarity matrix is used to find similarity of active user to the rest of users in system. An example of similarity calculation of users, say U_1 and U_2, is given in Table 1. When gender of user U_1 and U_2 matches, weight value increases for one. Also if one of categories of queries in logs match category of current search, weight increases on more time, but this time weight of interest while before was weight of personal attribute, summing up the total weight to two. In cases when a user turns to have more than one concurrent users in the matrix with the highest value of similarity to him/her, ranking of concurrent users follows. Ranking of user is provided using SNA metrics such as degree centrality or authority centrality. Each user in database is compared to current user for personal attributes matching such as gender or home city and for his topics of search queries earlier regarding current query topic. Summed weight is equal with sum of personal attributes weight plus sum of query's topic weight. Once weight's values are calculated for each user, we have a final network of users regarding to user comparing, represented as a |Ui| x |Uj| matrix.

```
   ALGORITHM A1.
(SIMILARITYWEIGHT (U, P (U)): U w → U

INPUT: User U in a set {U}, and set of
attributes of user expressed as P (U):
P (U) = Per (U) U Int (U), where

/* Per (U) stands for personal
attributes of user U, like gender Per1
(U), or location Per2 (U), etc.
Per (U) = {Per1 (U), Per_i (U), Per_m
(U)}*/

/* Int (U) stands for interest of user
in one of query categories, like sport
(Int1 (U) =2), or science (Int2 (U)
=0), etc. */
Int (U) = {Int1 (U), Int_j (U), Int_n (U)}

OUTPUT:  Similarity weight w of U_w → U
for a given pair U x U of users
/* Calculation of query category of
user */

Initialize win_t =0
/* For a same category of queries, e.g.
"science" as Int1 */
FOREACH Intj (j=1 to n)
   IF (Int_j (U) == Int_j (U_x)) // e.g.,
for U & U_x have same category of
queries
   THEN wint = wint + Int_j (U)
   IF (w_int == 0)
   THEN RETURN 0        // w = 0
   ELSE
```

```
/* then similarity weight of personal
attributes is calculated: */
   Initialize wper =0
/* for every personal attribute, e.g.
Per5 */
   FOREACH Peri (i=1 to m)
   IF (Peri (U) == Peri (Ux))
   THEN wper ++
/* Similarity weight is sum of two
distinct weights, personal and category
interest of query (wper and wint) */
   w = wint + wper
```

3.3 Rank before Query Recommendation

#	User	Rank
1	1047685	14985
2	1016497	14710
3	10437912	14708
4	1077807	14410
5	1016002	14207

Figure 2: User ranking process.

submitted query a comparison process should be done, otherwise for some given keywords recommended queries could be some queries not related to topic of submitted query. For a given set of keywords of submitted query Q and set of keywords of candidate query to compare Q1 the result of comparison Jaccard similarity coefficient calculated as intersection of sets Q and Q_1 divided by union of Q and Q_1.. Top N queries with the highest value of Jaccard similarity coefficient in ascending order. For instance keyword "cheap" in particular query submitted by a user, using Jaccard similarity coefficient process of filtering starts with calculation of Jaccard similarity coefficient value. For set Q denoted as set of submitted query which contains keywords {cheap, air} and set Q1 denoted as set of keywords which are compared with Q, Q1 = {cheap, air, fair} then result is J=2/3=0.66. If Q1 set would contain keywords as {cheap, airline, tickets} then J=1/3 = 0.33. Following this rule all candidate queries are compared to submitted query and at the end they are ordered in ascending order as recommended list of queries. This kind of similarity is similarity based on keywords, not in phrases. In future work we could extend current similarity algorithm to take into account also the phrases, which could improve accuracy of queries similarity calculation (Wen, et al.,

After similarity matrix is composed and weight of every pair of users is known, if there is more than one candidate user (most similar users) compared to active user, ranking of users should be done. Ranking of users is made using SNA metrics such as in-degree centrality or authority centrality (Algorithm A2).

```
ALGORITHM A2.
UsersRanking(Nu, Uc): Ranked(Uc)
INPUT.  A user network: Nu
Rank Uj by
     Authority Centrality or
     In-Degree Centrality
RETURN Ranked(Uc)
```

Queries of top N ranked users are retrieved from database respecting the order of ranking, such as first we take top N queries of most ranked user, if the list of queries is not filled with N required queries, queries of second ranked user are taken in account to recommend. In Figure 2 is shown ranking of users using in-degree centrality, which means User 1047685 is connected (similar) with 14985 users.

3.4 Filtering using Jaccard Similarity

In order to get only queries that are similar to 2001). For example, if phrase "the game of chess" could be recognized by our algorithm in query "the game of chess van huys", accuracy between query "the game of chess van huys" and "the game of chess van huygel" would be 0.5 instead of 0.4 which comes from similarity calculation based on keywords. Figure 3 represent live scenario of an example with same query "cheap air". The proposed system was developed in .Net and based in SQL Server database.

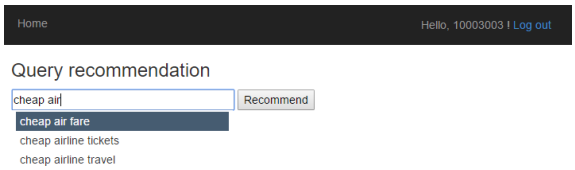

Figure 3: An example of proposed query recommendation system.

4 EVALUATION

The proposed recommendation system was evaluated using some random keywords and results compared to some of most popular search engine like Google and Bing[4] are as shown in Table 2. As result of dataset that have been used AOL and TREC in proposed recommended system, which are based on year 2006, some of recommended queries in Google and Bing are not included in our proposed system as a matter of time, for instance “flappy bird” did not exist in 2006, also “deadpool” (the movie) which are related to recent years and it was impossible to be included in recommended queries by our proposed system. Another issue to discuss is that we have been limited on datasets which means we had only 12713 queries to check for matching with submitted query, comparing these two huge datasets like Google and Bing. Our proposed recommended system experience the phenomenon of “cold start” as result of sparse data in dataset, not for every single word exist a keyword of a phrase in our database. In a future work we will try to experiment with data from local search engines in order to get more comparative results and enrich experiment results regarding our proposed approach.

4.1 User Acceptance

User study was helped by a group of 67 participants, forty-two of them were male while 25 female, including ages from 18 to 60, dominated by an average age of 30, because most of them were students while others were volunteers. The process was organized in two stages, first stage targeted the group of students, second stage our friends (friends of friends). They were asked to submit up to five queries (not mandatory) and for every submitted query they were asked to evaluate the result of recommended queries with five-star option if the intended query matched one of recommended queries. We doubt that some of user could not understand the question which intended to evaluate the result of recommendation. This may be one reason that from all participants our average feedback is 2.14, while 0 as lowest user rating and 5 as highest user rating. Based on certain statistics[5] reported in (Zhang & Nasraoui, 2008) which show that the hit rate of the related search keywords is over 10%, this is yet a promising results. In future work we intend to extend the demography of users by offering our system to different target groups of participants from different cities. The last but not the least is to prepare the proposed system for evaluation regarding concurrent systems like Google and Bing, in a matter of data (queries or keywords) that our proposed system lack, so system could be tested for keywords that exist in all systems that our system is comparing to, in order to avoid cases where we experience low coverage as result of our dataset

Table 2: Recommended queries for a random keyword search.

Query	Search engine recommended queries		
	Qrecco	Google	Bing
de	"de anza college" "deawoo auto parts", "deeb real estate omaha" "deer leather products"	"delta" "dell" "debenhams" "deadpool"	"delta airlines" "dell" "delta" "dell support"

[4] http://www.bing.com

[5] http://www.iask.com

fla	"flanagan and hunter and admiralty" "flash games" "flashlight bulb replacements"	"flashscore" "flash player" "flappy bird" "flashlight"	"Flash player" "flash" "flash player download" "flap"
sta	"staford auto mall" "state three of prostate cancer" "stanislaus california" "star of smokey and bandit" "star wars tree","state of california" "state of ia chamber of commerece" "state of washington map"	"status" "starbucks" "star wars" "staples"	"staples" "starbucks" "state farm" "state farm insurance" "staples office supply" "starfall" "startpage","stacey"
te	"tea tree oil" "ted low the low group" "teen pool parties" "teens and skull and crossbones" "teledyne laars parts" "tennesse county map" "test for autism"	"tesco" "test" "testris" "tesla"	"teamviewer" "tesla" "ted walks" "tetris" "textnow" "teleflora" "teamviewer download" "tesla motors"
ge	"genotype female cat" "gentech cancer" "gentech inc cancer" "gerogiame the poet" "georgian terrace hotel"	"geico" "george w bush" "genvideos" "george soros"	"geico" "geek squad" "geico insuarnce" "george michael" "gearbest" "general hospital"

REFERENCES

Ahmedi, L., Rrmoku, K. & Sylejmani, K., 2012. *Tourist tour planning supported by social network analysis.* Washington D.C, International Conference on Social Informatics.

Carrer-Neto, W., Hernandez-Alcaraz, M. L., Valencia-Garcia, R. & Garcia-Sanchez, F., 2012. Social knowledge-based recommender system. Application to the movies domain. *Expert Systems with Applications*, 39(12), pp. 10990-11000.

Cheng, H. & Cantú-Paz, E., 2010. *Personalized click prediction in sponsored search.* s.l., Proceedings of the Third ACM international Conference on Web Search and Data Mining.

He, J. & Chu, W. W., 2010. *A Social Network-Based Recommender System.* s.l.:Springer.

Kang, D. et al., 2013. Content Recommendation Method Using FOAF and SNA. In: *Advanced Technologies, Embedded and Multimedia for Human-centric Computing.* s.l.:Springer, pp. 93-104.

Konstas, I., Stathopoulos, V. & Jose, J. M., 2009. *On social networks and collaborative recommendation.* New York, Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval.

Phillips, J. M., 2013. *Jaccard Similarity and Shingling*, Utah: University of Utah.

Shokouhi, M., 2013. *Learning to Personalize Query Auto-completion.* Dublin, Ireland, ACM, New York, NY, USA, pp. 103--112.

Sohn, J.-S., Bae, U.-B. & Chung, I.-J., 2013. Contents Recommendation Method Using Social Network Analysis. *Wireless Personal Communications ,* 73(4), pp. 1529-15646.

Wasserman, S. & Faust, K., 1994. *Social Network Analysis: Methods and Applications.* Cambridge: Cambridge University Press.

Wen, J.-R., Nie, J.-Y. & Zhang, H.-J., 2001. *Clustering User Queries of a Search Engine.* Hong Kong, Proceedings of the 10th International Conference on World Wide Web.

Wu, S.-H., Hsieh, Y.-H., Chen, L.-P. & Ku, T., 2015. *Integrating Social Features and Query Type Recognition in the Suggestion Track of CLEF 2015 Social Book Search Lab.* s.l., Proceedings of the 6th International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction.

Zhang, Z. & Nasraoui, O., 2008. Mining Search Engine Query Logs for Social Filtering-based Query Recommendation. *Applied Soft Computing,* 8(4), pp. 1326-1334.