

A Scalable Framework Using Spatial Inverted Index for Context Aware Query Suggestion

¹ SARATH CHANDRA REDDY B, ² M. DHARANI KUMAR M.Tech.

¹ *M.Tech Student, Dept of CSE, P.V.K.K Institute of Technology,, Ananthapuramu, Affiliated to JNTUA,*

² *Assistant Professor, Dept of CSE, P.V.K.K Institute of Technology, Ananthapuramu, Affiliated to JNTUA*

Abstract: keyword suggestion has become one of the most fundamental features of commercial web search engines. After submitting a keyword query, the user may not be satisfied with the results, so the keyword suggestion module of the search engine recommends a set of m keyword queries that are most likely to refine the user's search in the right direction. Existing keyword suggestion techniques do not consider the locations of the users and the query results; i.e., the spatial proximity of a user to the retrieved results is not taken as a factor in the recommendation. However, the relevance of search results in many applications (e.g., location-based services) is known to be correlated with their spatial proximity to the query issuer. In this paper a weighted keyword-document graph is proposed, which captures both the semantic relevance between keyword queries and the spatial distance between the resulting documents and the user location. The graph is browsed in a random-walk-with-restart fashion to select the keyword queries with the highest scores as suggestions. To make our framework scalable, we propose a partition-based approach that outperforms the baseline algorithm by up to an order of magnitude. The appropriateness of our framework and the performance of the algorithms are evaluated using real data.

I. INTRODUCTION

However, to our knowledge, none of the existing methods provide location-aware keyword query suggestion (LKS), such that the suggested queries retrieve documents not only related to the user

information needs but also located near the user location. This requirement emerges due to the popularity of spatial keyword search.

Google processed a daily average of 4.7 billion queries in 2011,¹ a substantial fraction of which have local intent and target spatial web objects (i.e., points of interest with a web presence having locations as well as text descriptions) or geo-documents (i.e., documents associated with geo-locations). Furthermore, 53 percent of Bing's mobile searches in 2011 have a local intent.² In this paper, we propose the first Location aware Keyword query Suggestion framework. We illustrate the benefit of LKS using a toy example. Consider five geo-documents d1-d5 as listed in Fig. 1a. Each document d_i is associated with a location d_i as shown in Fig. 1b. Assume that a user issues a keyword query k_q "seafood" at location _q, shown in Fig. 1b. Note that the relevant documents d1-d3 (containing "seafood") are far from _q. A location aware suggestion is "lobster", which can retrieve nearby documents d4 and d5 that are also relevant to the user's original search intention. Previous keyword query suggestion models, ignore the user location and would suggest "fish", which again fails to retrieve nearby relevant documents. Note that LKS has a different goal and therefore differs from other location-aware recommendation methods (e.g., auto completion/instant searches, tag recommendation).

The first challenge of our LKS framework is how to effectively measure keyword query similarity while capturing the spatial distance factor. In accordance to

with the original query remains a challenging problem. In this work, we propose a novel query suggestion algorithm

previous query suggestion approaches, LKS constructs and uses a keyword document bipartite graph (KD-graph for short), which connects the keyword queries with their relevant documents as shown in Fig. 1c. Different to all previous approaches which ignore locations, LKS adjusts the weights on edges in the KD-graph to capture not only the semantic relevance between keyword queries, but also the spatial distance between the document locations and the query issuer's location q .

II. RELATED WORK

R. Baeza-Yates, C. Hurtado, and M. Mendoza.

In this paper we propose a method that, given a query submitted to a search engine, suggests a list of related queries. The related queries are based in previously issued queries, and can be issued by the user to the search engine to tune or redirect the search process. The method proposed is based on a query clustering process in which groups of semantically similar queries are identified. The clustering process uses the content of historical preferences of users registered in the query log of the search engine. The method not only discovers the related queries, but also ranks them according to a relevance criterion. Finally, we show with experiments over the query log of a search engine the effectiveness of the method.

Q. Mei, D. Zhou, and K. Church.

Generating alternative queries, also known as query suggestion, has long been proved useful to help a user explore and express his information need. In many scenarios, such suggestions can be generated from a large scale graph of queries and other accessory information, such as the clickthrough. However, how to generate suggestions while ensuring their semantic consistency

based on ranking queries with the hitting time on a large scale bipartite graph. Without involvement of twisted heuristics or heavy tuning of parameters, this method clearly captures the semantic consistency between the suggested query and the original query. Empirical experiments on a large scale query log of a commercial search engine and a scientific literature collection show that hitting time is effective to generate semantically consistent query suggestions. The proposed algorithm and its variations can successfully boost long tail queries, accommodating personalized query suggestion, as well as finding related authors in research.

Y. Song, D. Zhou, and L.-w. He.

Query suggestion is an interactive approach for search engines to better understand users information need. In this paper, we propose a novel query suggestion framework which leverages user re-query feedbacks from search engine logs. Specifically, we mined user query reformulation activities where the user only modifies part of the query by (1) adding terms after the query, (2) deleting terms within the query, or (3) modifying terms to new terms. We build a termtransition graph based on the mined data. Two models are proposed which address topic-level and term-level query suggestions, respectively. In the first topic-based unsupervised Pagerank model, we perform random walk on each of the topic-based term-transition graph and calculate the Pagerank for each term within a topic. Given a new query, we suggest relevant queries based on its topic distribution and term-transition

probability within each topic. Our second model resembles the supervised learning-to-rank (LTR) framework, in which term modifications are treated as documents so that each query reformulation is treated as a training instance. A rich set of features are constructed for each (query, document) pair from Pagerank, Wikipedia, Ngram, ODP and so on. This supervised model is capable of suggesting new queries on a term level which addresses the limitation of previous methods.

S. Basu Roy and K. Chakrabarti.

Users often search spatial databases like yellow page data using keywords to find businesses near their current location. Such searches are increasingly being performed from mobile devices. Typing the entire query is cumbersome and prone to errors, especially from mobile phones. We address this problem by introducing type-ahead search functionality on spatial databases. Like keyword search on spatial data, type-ahead search needs to be location-aware, i.e., with every letter being typed, it needs to return spatial objects whose names (or descriptions) are valid completions of the query string typed so far, and which rank highest in terms of proximity to the user's location and other static scores. Existing solutions for type-ahead search cannot be used directly as they are not location-aware. We show that a straightforward combination of existing techniques for performing type-ahead search with those for performing proximity search perform poorly. We propose a formal model for query processing cost and develop novel techniques that optimize that cost. Our empirical evaluations on real and synthetic datasets demonstrate the effectiveness of our techniques. To the best of our

knowledge, this is the first work on location-aware type-ahead search.

R. Zhong, J. Fan, G. Li, K.-L. Tan.

Location-Based Services (LBS) have been widely accepted by mobile users recently. Existing LBS-based systems require users to type in complete keywords. However for mobile users it is rather difficult to type in complete keywords on mobile devices. To alleviate this problem, in this paper we study the location-aware instant search problem, which returns users location-aware answers as users type in queries letter by letter. The main challenge is to achieve high interactive speed. To address this challenge, in this paper we propose a novel index structure, prefixregion tree (called PR-Tree), to efficiently support locationaware instant search. PR-Tree is a tree-based index structure which seamlessly integrates the textual description and spatial information to index the

spatial data. Using the PRTree, we develop efficient algorithms to support single prefix queries and multi-keyword queries. Experiments show that our method achieves high performance and significantly outperforms state-of-the-art methods.

III. EXISTING SYSTEM

In Existing system after submitting a keyword query, the user may not be satisfied with the results, so the keyword suggestion module of the search engine recommends a set of m keyword queries that are most likely to refine the user's search in the right direction.

However, none of the existing methods provide location-aware keyword query suggestion (LKS), such that the suggested queries retrieve documents not only related to

the user information needs but also located near the user location.

This requirement emerges due to the popularity of spatial keyword search. Google processed a daily average of 4.7 billion queries in 2011, a substantial fraction of which have local intent and target spatial web objects (i.e., points of interest with a web presence having locations as well as text descriptions) or geo-documents (i.e., documents associated with geo-locations).

DISADVANTAGES OF EXISTING SYSTEM:

Existing keyword suggestion techniques do not consider the locations of the users and the query results; i.e., the spatial proximity of a user to the retrieved results is not taken as a factor in the recommendation.

However, the relevance of search results in many applications (e.g., location-based services) is known to be correlated with their spatial proximity to the query issuer.

IV. PROPOSED SYSTEM

We propose the first Location-aware Key word query Suggestion framework. We illustrate the benefit of LKS using a toy example. Consider five geo-documents d1-d5 as listed.

Each document is associated with a location. Assume that a user issues keyword query seafood at location q. Note that the relevant documents d1–d3 (containing “seafood”) are far from q. A location-aware suggestion is “lobster”, which can retrieve nearby documents d4 and d5 that are also relevant to the user’s original search intention.

That LKS has a different goal and therefore differs from other location-aware recommendation methods (e.g.,

auto-completion/instant search tag recommendation). Section 5 provides a detailed discussion about the differences between LKS and these models, while in Section 4 we experimentally show that an adaptation of the method is less effective than LKS.

The first challenge of our LKS framework is how to effectively measure keyword query similarity while capturing the spatial distance factor. In accordance to previous query suggestion approaches LKS constructs and uses a keyword-document bipartite graph (KD-graph for short), which connects the keyword queries with their relevant documents.

ADVANTAGES OF PROPOSED SYSTEM:

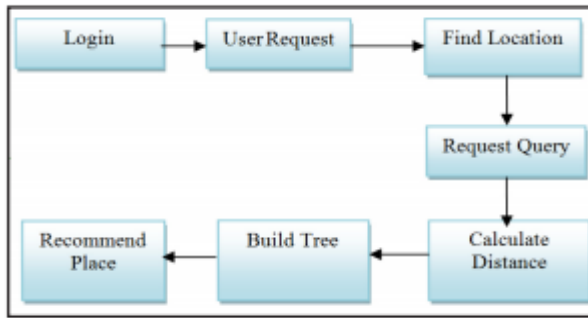
This LKS framework providing keyword suggestions that are relevant to the user information needs and at the same time can retrieve relevant documents near the user location.

A baseline algorithm extended from algorithm BCA is introduced to solve the problem. Then, we proposed a partition-based algorithm which computes the scores of the candidate keyword queries at the partition level and utilizes a lazy mechanism to greatly reduce the computational cost.

Empirical studies are conducted to study the effectiveness of our LKS framework and the performance of the proposed algorithms.

The result shows that the framework can offer useful suggestions and that PA outperforms the baseline algorithm significantly.

SYSTEM ARCHITECTURE



V. IMPLEMENTATION

• User Location Aware Module

- This is the first module the user can be authenticated whether the user is valid user or not .before that the user wants to register first.
- In registration the user have to give user name, password, mail id, location of the current place.
- For a security purpose the details will be encrypted before stored in to the data base.
- If the user is valid the user enters in to the application.

• Query Location Aware Module

- In this module the search details will be register like hotel name, location, special menu in the hotel and land mark.
- This module is used to view the details of the search query when the user searches in the search engine.
- In this module we need to to find latitude and longitude once we supply the place of the location

• user query

In consumer query module the consumer give a query to find the example the person needs to provide a current location and shopping object in a search engine, like present situation vadapalani and menu biriyani.

• Keyword Query Suggestion

- on this module the suggestion of a browsing query can be show depending upon the latitude and longitude of the person.

- We use Fast nearest Neighbor Search to find the nearest place of a user.
- The Location of the particular place will also display in a Google map.

VI. CONCLUSION

In this paper, we proposed an LKS framework providing keyword recommendations which are relevant to the user statistics needs and at the identical time can retrieve relevant files near the consumer location. A baseline set of rules extended from algorithm BCA is added to resolve the problem. Then, we proposed a partition-based totally set of rules which computes the rankings of the candidate keyword queries at the partition level and makes use of a lazy mechanism to substantially lessen the computational fee. Empirical research are carried out to look at the effectiveness of our LKS framework and the overall performance of the proposed algorithms. The end result shows that the framework can provide useful suggestions and that PA outperforms the baseline set of rules notably. In the destiny, we plan to similarly examine the effectiveness of the LKS framework by amassing more data and designing a benchmark. In addition, problem to the availability of statistics, we are able to adapt and check LKS for the case wherein the locations of the question issuers are to be had in the query log. Finally, we believe that PA also can be implemented to accelerate RWR on well known graphs with dynamic edge weights; we are able to investigate this capability in the future

References

- [1] R. Baeza-Yates, C. Hurtado, and M. Mendoza, "Query recommendation using query logs in search engines," in Proc. Int. Conf. Current Trends Database Technol., 2004, pp. 588–596.
- [2] D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log," in Proc. 6th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2000, pp. 407–416.
- [3] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li, "Context-aware query suggestion by mining click-

through and session data,” in Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2008, pp. 875–883.

- [4] N. Craswell and M. Szummer, “Random walks on the click graph,” in Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2007, pp. 239–246.
- [5] Q. Mei, D. Zhou, and K. Church, “Query suggestion using hitting time,” in Proc. 17th ACM Conf. Inf. Knowl. Manage., 2008, pp. 469–478.
- [6] Y. Song and L.-W. He, “Optimal rare query suggestion with implicit user feedback,” in Proc. 19th Int. Conf. World Wide Web, 2010, pp. 901–910.
- [7] T. Miyanishi and T. Sakai, “Time-aware structured query suggestion,” in Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2013, pp. 809–812.
- [8] A. Anagnostopoulos, L. Becchetti, C. Castillo, and A. Gionis, “An optimization framework for query recommendation,” in Proc. ACM Int. Conf. Web Search Data Mining, 2010, pp. 161–170.
- [9] P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna, “The query-flow graph: Model and applications,” in Proc. 17th ACM Conf. Inf. Knowl. Manage., 2008, pp. 609–618.
- [10] Y. Song, D. Zhou, and L.-w. He, “Query suggestion by constructing term-transition graphs,” in Proc. 5th ACM Int. Conf. Web Search Data Mining, 2012, pp. 353–362.
- [11] L. Li, G. Xu, Z. Yang, P. Dolog, Y. Zhang, and M. Kitsuregawa, “An efficient approach to suggesting topically related web queries using hidden topic model,” World Wide Web, vol. 16, pp. 273–297, 2013.
- [12] D. Wu, M. L. Yiu, and C. S. Jensen, “Moving spatial keyword queries: Formulation, methods, and analysis,” ACM Trans. Database Syst., vol. 38, no.1, pp.7:1–7:47, 2013.