Distributed Keyword Query Suggestion Using User Location Parameters and Document Proximity

PrathipatiRatna Kumar
Associate Professor
BattulaVaniPrasanna
M. Tech Student
C. R. Reddy College of Engineering, Eluru, West Godavari Dt, AP, India

Abstract **-**These Keyword suggestions are the most basic feature of the search engine. Naive users don't know how to express their queries correctly most of the time queries are short and unambiguous; keyword suggestion in web search assists users to access relevant information. Information will be more relevant if location of the users is considered. The location aware keyword (LKS) query suggestion method helps retrieve documents which relates to information provides by user and location where the user is located. However the search results relevance is known to be correlated with the spatial proximity of the user. In this paper we give brief study all the techniques for the keyword suggestions and discuss about location-aware keyword query suggestion framework and improved Partition Based algorithm

Keywords-Geo-Location, Keyword Search, Proximity,Query Suggestion, Spatial Databases

I. INTRODUCTION

Users often have difficulties in expressing their web search needs; they may not know the keywords that can retrieve the information they require [1]. Keyword suggestion (also known as query suggestion), which has become one of the most fundamental features of commercial Web search engines, helps in this direction. 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. Effective keyword suggestion methods are based on click information from query logs [2], [3], [4], [5], [6], [7], [8] and query session data [9], [10], [11], or query topic models [12]. New keyword suggestions can be determined according to their semantic relevance to the original keyword query. The semantic relevance between two keyword queries can be determined (i) based on the overlap of their clicked URLs in a query log [2], [3], [4], (ii) by their proximity in a bipartite graph that connects keyword queries and their clicked URLs in the query log [5], [6], [7], [8], (iii) according to their co-occurrences in query sessions [13], and (iv) based on their similarity in the topic distribution space [12]. However, none of the existing methods provide location aware keyword query suggestion, such that the suggested keyword queries can 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 [14], [15] that takes a user location and user-supplied keyword query as arguments and returns objects that are spatially close and textually\ relevant to these arguments. 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). Furthermore, 53% of Bing's mobile searches in 2011 were found to have a local intent.2 To fill this gap, we propose a Location-aware Keyword query Suggestion (LKS) framework. We illustrate the benefit of LKS using a toy example. Consider five geo-documents d1–d5 as listed in Figure 1(a). Each document di is associated with a location di:_ as shown in Figure 1(b). Assume that a user issues a keyword query kq = \seafood" at location _q, shown in Figure 1(b). 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 (e.g., [6]) ignore the user location and would_sh", 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 search 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 previous query suggestion approaches [3], [4], [5], [6], [7], [8], [10], [11], LKS constructs and uses a keyword-document bipartite graph (KD-graph for short), which connects the keyword queries with their relevant documents. 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. We apply a random walk with restart (RWR) process on the KD-graph, starting from the user supplied query kq, to find the set of m key- word queries with the highest semantic relevance to kq and spatial proximity to the user location. RWR on a KD-graph has been considered superior to alternative approaches [7] and has been a standard technique employed in various (location-independent) keyword suggestion studies [5], [6], [7], [8], [10], [11].

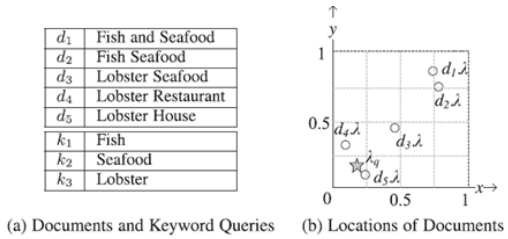

Fig:1.1 Lks example.

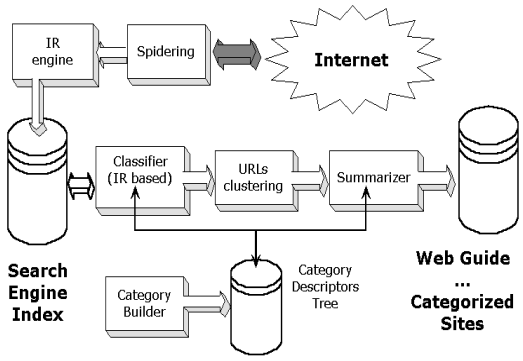

Fig: 1.2 Distributed Query Search

The second challenge is to compute the suggestions efficiently on a large dynamic graph. Performing keyword\ suggestion instantly is important for the applicability of LKS in practice. However, RWR search has a high computational cost on large graphs. Previous work on scaling up RWR search require pre-computation and/or graph segmentationpart of the required RWR scores are materialized under the assumption that the transition probabilities between nodes (i.e., the edge weights) are known beforehand. In addition, RWR search algorithmsthat do not rely on pre-computation accelerate the computation by pruning nodes based on their lower or upper bound scores and also require the full transition probabilities. However, the edge weights of our KD-graph are unknown in advance, hindering the application of all these approaches. To the best of our knowledge, no existing technique can accelerate RWR when edge weights are unknown a priori (or they are dynamic). To address this issue, we present a novel partition-based algorithm (PA) that greatly reduces the cost of RWR search on such a dynamic bipartite graph. In a nutshell, our proposal divides the keyword queries and the documents into partitions and adopts a lazy mechanism that accelerates RWR search. Pam and the lazy mechanism are generic techniques for RWR search, orthogonal to LKS, therefore they can be applied to speed up RWR search in other large graphs. In summary, the contributions of this paper are: We design a Location-aware Keyword query Suggestion (LKS) framework, which provides suggestions that are relevant to the user's information needs and can retrieve relevant documents close to the query issuer's location. _ We extend the state-of-the-art Bookmark Coloring Algorithm (BCA) for RWR search to compute the location-aware suggestions.

For maximizing fie availability in the mobile ad hoc network the replication can be used. If there is small number of replicas are used, file sharing can't be efficient. There is different file replication protocols used but they suffer from the problems like allocating limited resources to different files and second is storage as a resource for replicas. The solution provided for this is globally optimal file replication. Two models such as Random way point model and Community based models are used by Kang Chen [2]. In RWP, nodes are moving repeatedly at a selected point. So probability of meeting each node is similar. The randomly obtained speed is considered here. In case of community based mobility model the test area is taken which is split into different subareas called as caves. Each cave has one community. One node belongs to one or more communities. When node moves into its home community it has a probability Pin and when a node visits foreign community it has a probability 1- Pin .In case of optimal file replication, the meeting ability of a node as the average number of nodes it meets in a unit time and use it to investigate the optimal file replication. The probability of being encountered by other node is proportional to the meeting ability of the node. It indicates that files residing in nodes with higher meeting ability have higher availability than files in node with lower meeting ability. While creating the replica the memory is occupied. The probability of being met by others is decided by the nodes meeting ability so replica consumes both storage resource and meeting ability of the node. According to Yu-Chee-Tseng [3] the properties of MANETS can be dynamic changing topology, no base-station support, and multi hop communication capability. For communication they use the hopping concept. When two nodes are within the radio range, they communicate with each other using single hop function. The problem discussed here is about the flooding of broadcasting .The problem with broadcasting is storm problem. For this rebroadcasting can be done which is done on timely basis. The problem with broadcasting was that lower reach ability, redundancy,

contention and collision. These problems are considered in this paper which relives the broadcast problem and improves the reach ability and lowers the latency as compared to the flooding. The Probabilistic routing and file discovery protocols [4]–[6] are used to avoid broadcasting. They forward a query to a node with higher probability ofmeeting the destination. The other point of consideration will be the threshold. Threshold is the constant defined which gives the fixed host density. In this paper dynamic solutions to those problems are given which includes adaptive counter-based, adaptive location based, and neighbor coverage schemes. In adaptive counter based scheme each individual has capability to change or adjust its threshold based on neighborhood status. In adaptive location based scheme a host choose its threshold based on its current value of neighbor for determining whether to broadcast or not. Neighbor coverage scheme uses the accurate neighborhood information. Liangshan Yin [7] used concept collaborative caching in ad-hoc networks. Different collaborative techniques are used for accessing the data efficiently. The problem with MANET is infrastructure. So the data is transferred from node to node like routers. When mobile nodes works as request forwarding routers, bandwidth and power can be saved and delay can be reduced. In co-operative caching the sharing and co-ordination of the cached data is done among multiple nodes. So by using co-operative caching web performance is increased. The schemes such as Cache Path, Cache Data and Hybrid Cache are used in this paper. In Cache Data, popular items are cached locally. Intermediate node cache data and then serves this data for future requests. For caching the data space is required. The problem with Cache Data is that same data item can be cached at two or more nodes. Because of which there is wastage of large amount of cache. To avoid this problem, the rule used is that, a node does not cache data if all requests for the data are from same node. In Cache Path intermediate node knows that which node has requested which data because the path of the requesting node and destination is saved in the cache. So when other node request for particular data item, the intermediate node calculate number of hops (distance) and then data item present on nearest node is served. Which means that it cache the data path. Because of which bandwidth and query delay can be reduced. For saving the path, there is no need to save all node information as the path from current router to the destination can be found by underlying routing algorithm. In Hybrid cache path and cache data schemes are combined means that when a data tem needs to be cached it uses Cache Data and path for that data item can also be cache. According to Huang et al. [9], Wi-Fi-based wireless networks based on node mobility pattern, AP topology and file popularity, caching files in servers is done for realizing the optimal file availability to mobile users. However, the file servers considered are fixed nodes connecting to APs.we propose a distributed file replication protocol that can approximately realize the optimal file replication rule with the two mobility models in a distributed manner.

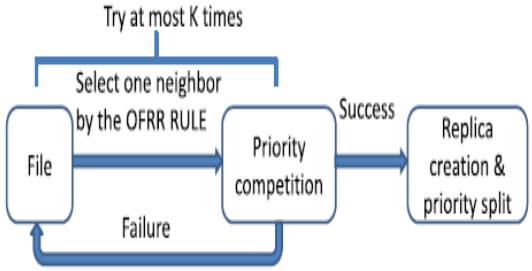

Fig: 1.3 Optimal File Replication Rule

As a result, the number of replicas of each file is proportional to the sum of the meeting abilities of its replica nodes, thereby realizing Formula (22), i.e., OFRR.

II. PROPOSED MODULES

Admin has to login with valid username and password. After login successful he can do some operations such as view all user, their details and authorize them, Add hotels(Hotel name,Location,Area name, Item name, item price, item description, item image, no. Of rooms available, Room Charge Distance from Location), Add malls(Mall name,location,area name, mall description, mall specialization,mall image, Distance from Location) , View all hotel details with rank, Comments , view all mall details with rank, comments, View all hotel booking details and payment details, view hotels and mall rank result chart, view top k searched keywords in chart . In this module, there are n numbers of users are present. User should register before doing some operations and also add your location while registration. After registration successful he can login by using valid user name and password and location. After Login successful he will do some operations like view profile details, Create and manage account, search nearest neighbor hotels and malls from your location and view details, GMap, give comment, Book hotels, show top K searched keywords.

III. RELATED WORK

Related work on query suggestion is discussed in Section 3.1. Techniques for RWR computation are reviewed in Section 3.2.

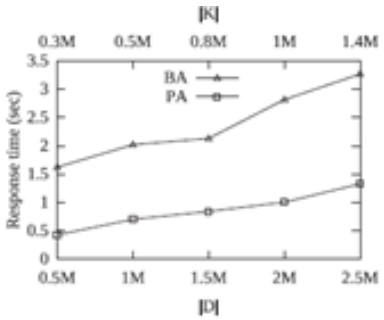

Fig 3.1: Varying number of documents and keywords

3.1 Keyword Query Suggestion:

Keyword query suggestion approaches can be classified into three main categories: random walk based approaches, learning to rank approaches, and clustering based approaches. We also briefly review alternative methods that do not belong to any of these categories. To the best of our knowledge, no previous work considers user location in query suggestion.

3.2 Random Walk Computation:

Random walk with restart, also known as Personalized Page Rank, has been widely used for node similarity measures in graph data, especially since its successful application by the Google search engine

IV.CONCLUSIONS

In this paper, we proposed an 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 (PA) 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. In the future, we plan to further study the effectiveness of the LKS framework by collecting more data and designing a benchmark. In addition, subject to the availability of data, we will adapt and test LKS for the case where the locations of the query issuers are available in the query log. In addition, we believe that PA can also be applied to accelerate RWR on general graphs with dynamic edge weights and we will investigate its general applicability in the future. Moreover, the current version of PA seems to be independent of the partitioning method. It would be interesting to investigate whether alternative partitioning heuristics can further reduce the cost of the algorithm.

V. REFRENCES

[1]. M. P. Kato, T. Sakai, and K. Tanaka, "When do people use query suggestion?A query suggestion log analysis," Inf. Retr., vol. 16, no. 6, pp. 725–746, 2013.
[2]. R. Baeza-Yates, C. Hurtado, and M. Mendoza, "Query recommendation using query logs in search engines," in EDBT, 2004, pp. 588–596.
[3]. D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log," in KDD, 2000, pp. 407–416.
[4]. 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 KDD, 2008, pp. 875–883.
[5]. N. Craswell and M. Szummer, "Random walks on the click graph," in SIGIR, 2007, pp. 239–246.
[6]. Q. Mei, D. Zhou, and K. Church, "Query suggestion using hitting time," in CIKM, 2008, pp. 469–478.
[7]. Y. Song and L.-w.He, "Optimal rare query suggestion with implicit user feedback," in WWW, 2010, pp. 901–910.
[8]. T. Miyanishi and T. Sakai, "Time-aware structured query suggestion," in SIGIR, 2013, pp. 809–812.
[9]. A. Anagnostopoulos, L. Becchetti, C. Castillo, and A. Gionis, "An optimization framework for query recommendation," in WSDM, 2010, pp. 161–170.
[10]. P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna, "The query-flow graph: Model and applications," in CIKM, 2008, pp. 609–618.
[11]. Y. Song, D. Zhou, and L.-w.He, "Query suggestion by constructing term-transition graphs," in WSDM, 2012, pp. 353–362.
[12]. 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," WWW, pp. 273–297, 2013.
[13]. U. Ozertem, O. Chapelle, P. Donmez, and E. Velipasaoglu, "Learning to suggest: A machine learning framework for ranking query suggestions," in SIGIR, 2012, pp. 25–34.
[14]. 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, 2013.
[15]. D. Wu, G. Cong, and C. S. Jensen, "A framework for efficient spatial web object retrieval," VLDB J., vol. 21, no. 6, pp. 797–822, 2012.

Author's Profile:

P.Ratnakumar he pursuing P.H.D from Acharayanagarjuna university. He has completed M.tech from REC Warangal. Hisarea of interest in Software Testing.His 20 years of teaching industrial experirnce .he is currently associated with Sir C R Reddy College of Engineering, Eluru, West Godavari District A.P. India. Affiliated to Andhra University.

B.vaniprasanna student of M.Tech. in Computer Science and Technology from C.R. Reddy College of Engineering, Eluru, West Godavari Dt, Andhra Pradesh. She is currently working on his project.