Personalized Approach for Mobile Search

Feng Gui, Malek Adjouadi, Naphtali Rishe
Center for Advanced Technology and Education
College of Engineering and Computing
Florida International University
gui_feng@yahoo.com, adjouadi@fiu.edu,rishen@cs.fiu.edu

Abstract

Mobile search is becoming increasingly important for mobile users as mobile devices are more widely used. Mobile search is quite different from standard PC-based web search in a number of ways: (a) the user interfaces and I/O are limited by screen real estate, (b) key pads are tiny and inconvenient for use, (c) limited bandwidth and (d) costly connection fees. These limitations result in more navigational queries in the mobile search. Furthermore, user location, activities, preferences, and interaction history can also improve accuracy in determining relevance for mobile search. In the past, most personalized search algorithms are studied in the context of PC-based web search. Personalized mobile search should however play a bigger role at improving the user experiences. This paper focuses on the personalization strategies which explicitly and implicitly infer user search context at individual user level. We propose an architecture which collects user information (at mobile device and carrier network) and derives user intention in given situations. We show that personalized mobile search perform well for ambiguous queries and localized searches.

Keywords

Mobile Search, Personalized Search, Context Awareness, User Profile.

1. Introduction

Mobile search is the second most used application only after social networking in wireless internet [1]. Search engine like Google appear in top three of the most visited web sites in terms of wireless internet usage.

Most mobile search queries are short due to the hardware limitations such as tiny keypads and small screen. Early studies [2] attempted to provide solutions to mitigate the hardware limitations of the wireless devices. The top 100 mobile queries at AT&T [3] reveal that a great number of search queries are navigational [4] in nature. The navigational searches, for example “Google”, usually steer mobile users to specific web sites conveniently. Unlike navigational queries, words like “images” and “free” which are informational and transactional are ambiguous to search engine. A housewife and an iPhone user interpret “apple” differently in search context. A housewife is likely to know the apple variety and prices at the local grocery stores. While an iPhone user is interested in service or products related to iPhone. Researchers studied methods and models to determine the query ambiguity. Clarity score [5] was proposed to evaluate the relative entropy between the query language model and the collection language model. A large click entropy indicates that user clicks more web pages to solve the query, thus the query is ambiguous. A small click entropy means mobile users have common understanding for a search query. Song [6] developed classifier to automatically identify three types of queries, ambiguous, broad, or clear query. We believe these methods and algorithms work equally well to identify the query ambiguity in the mobile search.

Researchers have explored personalized search to improve topical relevance of result documents in PC-based web search. Shen et al. [7] studied user's immediate and short-term search context to expand the current query. Qiu and Cho [8] learned user interest from the click history and developed ranking mechanism based on the user interest. Chirita et al. [9] proposed personalized search and summarization algorithms which assist search keywords expansion based on extracted information from local desktop. Duo et al. [10] and Teevan et al. [11] investigated the personalized search strategies and stated that personalization improves the search accuracy on ambiguous queries.

So far the personalization is studied only for PC-based web search. Most of such personalization strategies are limited to the user search history,

978-0-7695-3507-4/08 $25.00 © 2008 IEEE
DOI 10.1109/CSIE.2009.718

returned search results, and documents stored in PC. We extend this line of work into mobile search and derive user context based on profiles which adapt to user location, activities, interaction history, and preferences. We also believe that the increased usage of location-based applications and services will lead mobile users to search for local services and information. Liu and Birnbaum [12] developed LoalSavvy which aggregates local views associated with news events or topics. Vadrevu et al. [13] pointed out the importance of identifying the regional sensitive queries. Hence, the mobile search must cope with local search query.

We propose a smart proxy on the wireless device to derive user context profile. A dynamic neural network based algorithm is utilized to generate context weights which adapt to the user's given situation, location, activities, and history information. In addition, the intelligent manager on the carrier server expands the search query with the context profiles and reranks the result documents based on the context profiles.

2. Architecture Design

The traditional client/server paradigm fits well into the mobile search application. The carrier's network is the server which provides data and voice services to subscribers. User submits mobile queries to network. The mobile devices play an extensive role of collecting, analyzing, and extracting context entities. Context profiles compiled at client side adapt to the mobile user's current situation.

Figure 1 shows the paradigm of the client/server model. The user inputs (voice or data) and surrounding environment inputs (temperature, GPS reading, altitude, e.g.) are collected by the hardware logic or the applications such as the user calendar. The context-aware proxy further inspects inputs and extracts context entities from the inputs. Finally, the proxy compiles the context profile and sends it to the carrier server.

The carrier server learns the user situation from the context profile in addition to data collected at the network. For example, the network context unit collects user information, i.e., location and query history. The intelligent manager carries out a few important functions such as identifying the ambiguous queries and expanding these queries with context profiles.

Expanded query contains extra information which could facilitate search engines and improve the topic relevance of returned documents. Regional sensitive query could be solved if context profile reflects the user intention and location.

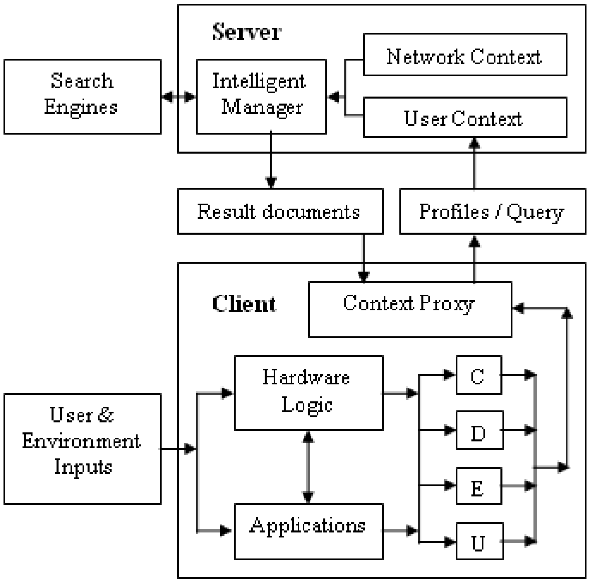


Figure 1: System architecture

E: environment profile	U: user profile
C: cache profile	D: device profile

3. Context Proxy

One of the trends in the wireless device is to integrate more sensors, e.g. tilt sensor and touch screen LCD on iPhone. The software applications on wireless devices advance in parallel. Hence, context proxy on wireless devices is able to collect user information, analyze user search context, and compile context profiles.

3.1. Context Profiles

The context profile [14] is a collection of context entities extracted from the on-board sensors, client applications, user activities, and so on. Ideally, context profiles should be a good reference of user's current situation from which the carrier's network could derive user's context or intention. Thus, the proxy on mobile devices frequently updates context profiles and uploads them to network for reference if necessary. Due to the limited bandwidth [15], the context profiles should be concise. As user changes activities/context, context entities are added to or dropped from the profiles based on the algorithms we specify in the later section. Whenever addition or deletion of context entities occurs, the proxy notifies network the changes. If the user's situation remains the same or changes a little, then only updated weights in the context profiles are sent to network. There are four context profiles managed by the client proxy;

1. User profile describes the user's characteristics, preferences, activities, emotions, and so on. For example, user's age, gender, height, weight, health conditions, past incidents, and planned events can be sorted into this category.
2. Device profile describes the configuration and functions of the hardware. In addition, this profile includes the software applications and user interfaces. Examples of context entities in this profile are processor, memory, display type, device size, and OS.
3. Environment profile describes surrounding area of the mobile user. This profile stores user location, weather, noise level, or temperature, etc.
4. Data profile caches user's data in the local memory.

3.2. ANN Algorithm

Artificial Neural Networks (ANN) are widely utilized in many of today's novel applications for solving practical problems, such as pattern recognition, classification, forecast studies, and cluster analysis. The learning rule is based on the following general equation:

$$\Delta W_{ij}(t) = x_i(t)y_j(t) \qquad (1)$$

We modify $x_i(t)$ as a function of the context changes ΔC and frequency count ΔF of the context entities in the context cache managed by the client proxy between the current epoch and the previous epoch. In addition, $y_j(t)$ is defined as the weight values, $W_{reference}$, in the current user situation.

Thus, the derived equation of weight change for our approach at epoch t follows:

$$\Delta W_k(t) = \left|\Delta C(t) + \mu\Delta F(t)\right|_k W_{k,reference}(t) \qquad (2)$$

where $\Delta C(t)$ is calculated as $\Delta C(t) = \frac{|C(t) - C(t-1)|}{C(t-1)}$. $C(t)$ is the current context value, and $C(t-1)$ is the previous context value. If $\Delta C(t)$ is zero, then $\Delta C(t) = -|C(t) - Caverage|$. Caverage is the mean of C(1) through C(t-1). $\Delta F(t)$ is the difference in frequency counter for context entity between at current epoch and previous epoch, $\Delta F(t) = F(t) - F(t-1)$. μ is the coefficient that is machine specific.

Finally, the weights for context entity k at epoch t are updated as

$$W_k(t) = W_k(t-1) + \Delta W_k(t) \qquad (3)$$

If the current user location contributes new context weights that do not exist in the previous context setting, then the initial value of $W_k(t)$ is set to 1.

Environment profiles, user profiles, and data profiles are updated based on the above equations such that these profiles adapt to current user situation. Ambiguous queries submitted to the search engines are expanded with the more user information from these profiles.

4. Intelligent Manager

Intelligent manager is the most important component at the carrier's server side. It learns the mobile user's context information at the network level and analyzes the context profiles sent from the client devices. When mobile user submits query, intelligent manager identifies the ambiguous queries and further expands these queries with derived user context. The expanded queries are then submitted to search engines.

4.1. Search Query Expansion

The context profiles compiled by proxy at client devices reflect the hardware configuration, user applications, usage history, and derived user situation. Higher weights mark the importance of context entities. Of course, not all context entities in the profiles have changing weights. The hardware profile has weight-fixed context weights due to the stable hardware and software configuration at the client device.

Context profiles include the weights of the context entities calculated at proxy, p = [CWi+1, CWi+2, CWi+3, … , CWi+n], where i Є [0, 1, 2, 3…]. Let the query forms the set, q = [T1, T2, T3, … ,]. After query expansion, the augmented context profile is:

$$AP = \sum_{i=1}^{m} CW_i + \sum_{j=1}^{n} K(T_j) \qquad (4)$$

where $K(T_j)$ is the offset function. An example augmented context profile could be AP = CW_{i+1}, CW_{i+2}+ $K(T_2)$, CW_{i+3}, … , CW_{i+n}, $K(T_{i+n+1})$, …]. If the terms/words from the query set are matched in the context profile, then the offset weight of the term/word, $K(T_j)$, is added to the context weight, CW_{i+n}. Furthermore, if the terms/words from the query set are not found in the context profile, then the offset weights of these terms, $K(T_{i+n+1})$, are merged to the context profile. It is obvious that the augmented context profile is likely to have more weights on the query words.

4.2. Ranking Documents

The search engines return all the documents related to the query based on the augmented context profile. The returned documents are grouped into categories and ranked in terms of descending weights in the context profile. Categories that match more context weights in augmented context profile have higher rank. Because most mobile devices have small displays, the maximum documents in each category are limit to a number. This method relies on the user selection to further narrow down user current context.

When the user selects the data or services downloaded from the server, his/her selection would greatly help the system understand user's intention. Hence, it is important to utilize the user's selection, as feedback, to adjust context weight calculation at client and server sides. The server embeds the calculated context weights in data to the user device. The embedded context weights are transparent to users. However, the user selection on the data allows the related context weights to feed back into the proxy. This feedback mechanism significantly improves the accuracy of the server response.

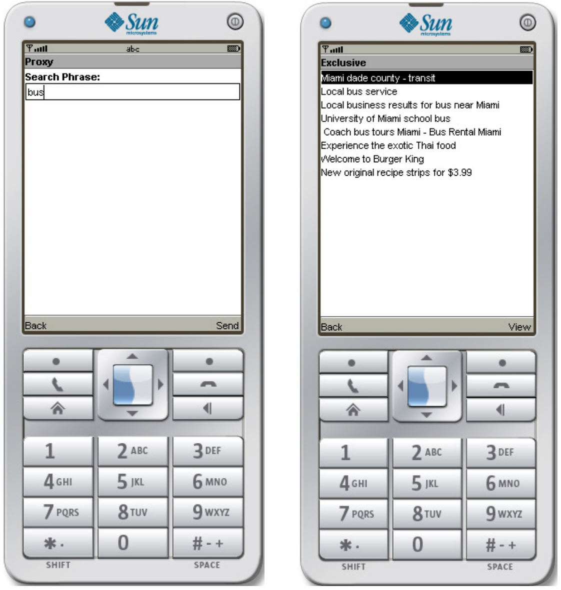
5. Experiment

In this section, we provide one user scenario that is close to the daily activities of mobile user. The value of μ is set to 0.05 in our simulation. To simplify the context computing, we quantify some context entities as real numbers. Weather: sunny = 1, cloudy = 2, raining = 3; Location is expressed by (x, y) such as (1, 1), (2, 2),…; Store Type: department store 10, Woman Cloth = 20, foot wear = 30; food store has a type in 40, specifically Chinese food counter = 41, Japanese food counter = 42, Subway = 43, Pizza Hut = 44. To simplify the simulation, we only present a few context entities in simulation.

In this scenario, the user carrying mobile device walks at varying speeding in a fixed direction. During his trip he submits search query. We describe the first scenario with the following seven steps.

1) The user is walking by location (1, 1) at the speed of 3 mile/h when the weather is sunny (1), and the temperature is 77. He goes by department store (10).
2) While the user is still walking at the same speed passing location (2, 2), the weather changes to cloudy (2) and the temperature drops to 73. He goes by woman cloth (20).
3) When it starts raining (3) and temperature continues to drop to 69, the user is running at the speed of 6 mile/h towards the same direction and passing location (4, 4). He goes by shoe store.
4) User is still running at the same speed passing location (6, 6). Temperature changes slightly to 68. It is raining (3). He goes by Chinese restaurant.
5) User keeps running at the same speed reaching location (8, 8) while it is raining (3). Temperature remains the same. He goes by Subway.
6) User stops at location (8, 8) while temperature drops slightly to 67 and it rains (3). He is at Pizza Hut. He submits search query "bus".
7) User stays at location (8, 8) while temperature rises slightly to 68 and it is raining (3).

The Sun Java Wireless Toolkit is used to implement the context proxy. The query is submitted in the search application. The query word and returned documents are shown in Figure 2.



Proxy

Search Phrase:

bus

Exclusive

Miami dade county - transit
Local bus service
Local business results for bus near Miami
University of Miami school bus
Coach bus tours Miami - Bus Rental Miami
Experience the exotic Thai food
Welcome to Burger King
New original recipe strips for $3.99

Figure 2: Search application

The returned documents are summarized in table 1.

Sites	Description
www.miamidade.gov/transit	Miami Dade Country - Transit
www.miami-airport.com/html	The local city bus service , provides scheduled services to all areas of Miami-Dade County.
Maps.google.com	Local business results for bus near Miami, FL
www.bus.miami.edu	Home: University of Miami school of business
Tai's Cuisine	Experience the Exotic Thai food.
www.burgerking.com	Welcome to burger king.
www.kfc.com	New original recipe strips for $3.99.

Table 1: Result documents

The returned documents show that the personalized search results include documents related to the mobile query. Key word “bus” is ambiguous without specifying location. Because the location is included in the user context, then the search engine understands it is a search in Miami area. In addition, the context profiles reveal another important topic which is food. Thus, the returned documents also include links related to food or restaurants in the local area. The context profiles assist the search engine to provide local information.

6. Conclusion

With the wide spread of mobile devices, mobile search becomes an important application for mobile experience. Mobility, social activities, and locality add to the sophistication of mobile search. Personalized search has been the focused approach for PC-based search lately. We extend this line of work to mobile search. We propose a client-side context proxy which collects user information, analyses context information, and compiles context profiles. These context profiles, which reflect user current context, assist query expansion to solve ambiguity. An intelligent manager is proposed to process context profiles, identify ambiguous query, expand query, and reorder the result documents based on user current topics in the context. The simulation result shows that the proposed architecture can adapt to user situation and provide local information for mobile search.

References

[1] Opera Software, “State of the Mobile Web Report: First Quarter”, May 20, 2008

[2] A. Soffer, Y. Maarek, and B.W. Chang, “WWW2002 Workshop on Mobile Search”, Mobile Search, Honolulu, Hawaii, May, 2002.

[3] B Meunier, “Characteristics of the Top 100 Mobile Search Queries at AT&T”, Jan, 2008.

[4] B.J. Jansen, D.L. Booth, and A. Spink, “Determining the User Intent of Web Search Engine Queries”, The International World Wide Web Conference, Banff, Canada, May 8-12, 2007.

[5] S. Cronen-Townsend and W.B. Croft, “Quantifying Query Ambiguity”, Proceedings of the Conference on Human Language Technology, San Diego, 2002, pp. 94–98.

[6] R Song, Z.X. Luo, J.R. Wen, and H.W. Hon, “Identifying Ambiguous Queries in Web Search”, The International World World Wide Web Conference, Banff, Canada, May 8-12, 2007.

[7] X.H. Shen, B. Tan, and C.X. Zhai, “Implicit User Modeling for Personalized Search”, Conference on Information and Knowledge Management, Bremen, Germany, 2005.

[8] F. Qiu and J.H. Cho, “Automatic Identification of User Interest for Personalized Search”, The International World Wide Web Conference, Edinburgh, UK, May 22-26, 2006.

[9] P.A. Chirita, C.S. Firan, and W.Nejdl, “Summarizing Local Context to Personalize Global Web Serach”, Information and Knowledge Management, Arlington, Virginia, November 5-11, 2006.

[10] Z.C. Dou, R.H. Song, and J.R. Wen, “A Large-scale Evaluation and Analysis of Personalized Search Strategies”, International World Wide Web Conference, Banff, Canada, May 8-12, 2007.

[11] J. Teevan, S.T. Dumais, and D.J. Liebling, “To Personalize or Not to Personalize: Modeling Queries with Variation in User Intent”, ACM SIGIR Conference, Singapore, July 20-24, 2008.

[12] J. Liu, and L. Birnbaum, “What do they think? Aggregating Local Views about News Events and Topics”, International World Wide Web Conference, Beijing, China, April 21-25, 2008.

[13] S. Vadrevu, Y. Zhang, B. Tseng, G. Sun, and X. Li, “Identifying Regional Sensitive Queries in Web Search”, The International World Wide Web Conference, Beijing, China, April 21-25, 2008.

[14] F. Gui, C.M. Liu, X. Chen, and D.D. Wang, “A Context-Awareness Algorithm Based on Data Cache for Mobile Computing”, World Wireless Congress, San Francisco, May 25-28, 2004.

[15] S Shiraishi, D Sakuma, M Haseyama, H. Kitajima, “Bandwidth-efficient Information-providing System for Mobile Phone Users”, International Conference on Consumer Electronics, Jan 2006.