Search Scripts Mining from Wisdom of the Crowds

Chieh-Jen Wang and Hsin-Hsi Chen
Department of Computer Science and Information Engineering
National Taiwan University
Taipei 106, Taiwan
{d96002, hhchen}@csie.ntu.edu.tw

***Abstract*—This paper mines sequences of actions called *search scripts* from query logs which keep large scale users' search experiences. Search scripts can be applied to predict users' search needs, improve the retrieval effectiveness, recommend advertisements, and so on. Information quality, topic diversity, query ambiguity, and URL relevancy are major challenging issues in search scripts mining. In this paper, we calculate the relevance of URLs, adopt the Open Directory Project (ODP) categories to disambiguate queries and URLs, explore various features and clustering algorithms for intent clustering, and identify critical actions from each intent cluster to form a search script. Experiments show that the model based on a complete link hierarchical clustering algorithm with the features of query terms, relevant URLs, and disambiguated ODP categories performs the best. Search scripts are generated from the best model. When only search scripts containing a single intent are considered to be correct, the accuracy of the action identification algorithm is 0.4650. If search scripts containing a major intent are also counted, the accuracy increases to 0.7315.**

Keywords-mining web logs*; *web search enhancement*; *search script

I. INTRODUCTION

A user's search intent can be as simple as getting the price of a commodity or as complicated as finding information for planning a trip, which may contain some sub-needs such as searching for location information, tourist attractions, accommodations, transportations, etc. For a complex intent, where relevant information spans many websites of different subject matters, users have to submit more than one query and browse through several relevant websites. Because it is not trivial for users with different backgrounds and search ability to express clear and complete information needs, reducing the barriers between information needs and query formations is essential for information access.

Query logs consist of a large collection of search sessions which keep searching and browsing trails of users. Thus, query logs can be considered as a type of wisdom of crowds if knowledge is mined. In a session, users represent their intents by queries and clicked URLs. Similar representations demonstrate similar intents. Critical actions embedded in similar intents represent a sequence of actions, called *search script*, to be performed for a common information need. This paper aims to mine search scripts from query logs. Search scripts, which illustrate collective intelligence of web users, will be useful for search need prediction, retrieve effectiveness improvement, advisement recommendation, and so on.

Information quality, topic diversity, query ambiguity, and URL relevancy are challenging issues in search script mining from query logs. Query logs suffer from noise because they keep information from naïve users to expert ones. Besides, the same intent may be in terms of different queries in different term orders. Queries may be ambiguous, and clicked URLs may not be always relevant. Moreover, the intent boundaries are not clear in query logs. That is, a user's session may contain more than one intent.

In this paper, we calculate the relevance of clicked URLs, disambiguate the uses of queries and clicked URLs, explore clustering algorithms on different intent representations, extract critical actions in intent clusters, and create search scripts. Figure 1 shows a search script example to draw up a plan for watching a baseball game in Philadelphia. Each node denotes an action and some sample queries. The script will be recommended when users have an information need for watching a baseball game in Philadelphia.

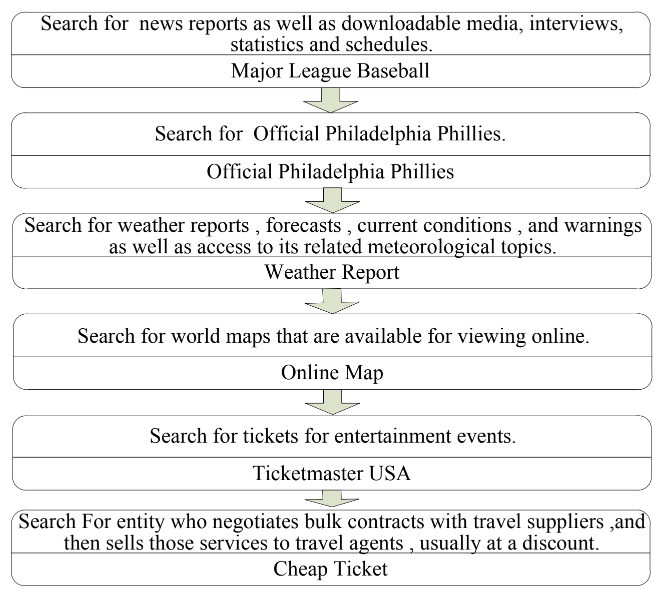

Figure 1. A script example.

The rest of this paper is organized as follows. In Section II, we compare our research with others and summarize our contributions. Section III describes the primary resource used in this study. Section IV presents clicked URLs relevance estimation, queries and clicked URLs disambiguation, and intent clustering algorithms. Section V shows how to extract

978-1-4577-0653-0/11/$26.00 ©2011 IEEE

critical actions from an intent cluster. Section VI evaluates the quality of intent clusters and search scripts, and discusses the findings along with their implication. The last section concludes the remarks.

II. Related Work

Queries are usually short and even ambiguous. To realize the meanings of queries, researchers define taxonomies and classify queries into predefined categories. Broder [1] divided queries into navigational, informational and transactional types. Nguyen and Kan [2] characterized queries along four general facets of ambiguity, authority, temporal sensitivity and spatial sensitivity. Manshadi and Li [3] classify queries into finer categories. Similarly, clicked URLs can be classified into predefined categories. Shen et al. [4] adopt the Open Directory Project (ODP) taxonomy to represent clicked URLs and model the topic transition.

Clicked URLs may not be relevant to queries due to search engine performance and/or users' comprehension. Modeling users' click behaviors has been studied intensively in recent years. Joachims et al. [5] adopted an eye-tracking equipment to observe the users' behaviors when browsing web pages. The works on click models, e.g., cascade model [6], multiple-click model [7], click chain model [8], and dynamic Bayesian network click model [9], predicted click probability of a web page under different postulations.

Query recommendation is one of the most popular applications of using search query logs. Suitable queries are recommended to users when they initiate searches. That helps users satisfy their information needs quickly. Methods [10][11] utilizing click-through data and search query logs have been proposed. The major issue is how to measure the similarity of queries and recommend the similar queries. The similarity may be done by using the surface forms of queries, overlap of clicked document of queries [12][13][14], or the category or content of clicked documents [13].

Our contribution is different from the others. We integrate different cues in search sessions to group similar user intents. Besides, we propose search scripts, which are sequences of actions rather than single queries, to guide user satisfying their information needs.

III. Resource

Two main resources of our experiments are introduced in this section. Live Search Query Logs which were distributed by Microsoft Research is the primary source of data for this study. The other is Open Directory Project (ODP) [1] at dmoz.org, the largest, most comprehensive, and most widely distributed human-compiled taxonomy of websites.

A. Query Logs

We adopt MSN Search Query Log excerpt - RFP 2006 dataset [15] in this study. This data set consists of 14.9 million queries and 12.2 million clicks during a one-month period in May 2006. For each query, its query terms, query time, and the associated session is recorded in the logs. For each click, the clicked URL, the click time, and the associated query are recorded in the logs. In total, there are 7.4 million sessions.

B. Open Directory Project

ODP containing more than 4 million websites and being organized into more than 500 thousand categories is constructed and maintained by a vast, global community of volunteer editors. It contains web page annotations written by people collaboratively, provides category and category description metadata for URLs. A path [16] is an ordered hierarchical structure of hyperlink labels from the root category of a directory to a leaf. The path in the ODP is represented by $l_1/l_2/\ldots/l_n$, which is an ordered sequence of hyperlink labels l_1, l_2, …, l_n from the root (l_1) of the ODP to the leaf (l_n). The labels on the left in a path dominate the labels on the right. In other words, l_{i+1} is a subcategory of l_i.

IV. Intent Clustering

Two clustering algorithms are employed to cluster sessions of identical intent. A cluster model is a combination of features and a clustering algorithm. Clustering results will be evaluated by applying them to the application of intent boundary detection, and the best model will be selected for search script generation.

A. URL Relevance Determination

Users' clicks depend on various issues such as performance of search engines, user comprehension, positions of the results, and so on. Clicking URLs expresses users' interests after reading the corresponding snippets. Thus, a click can be regarded as some sense of relevance voting. However, a URL may be irrelevant to the submitted query after a document is browsed. Here we adopt a function to compute the relevance of a URL with considering position bias [17]. Given a query q, we collect all clicked URLs u of q in query logs at each position p, and then compute click through rate (CTR) at each position. Equation (1) defines the relevance $rel(q,u_i)$ of URL u_i for query q.

$$rel(q,u_i) = \frac{click(q,u_i)}{\sum_{p=1}^{s} q_p \times CTR_p} \tag{1}$$

The numerator is total number of clicks of URL u_j under query q in qeury logs. The denominator can be represented as expected clicks in such a way that q_p denotes the number of URL u_i results of query q at position p, CTR_p is the CTR at position p, and s denotes the maximum position of u_j in the results.

B. Category Disambiguation

We postulate that the sequence of clicked URLs in a session satisfies a user's intent. Thus, the clicked URLs are coherent and co-related. The clicked URLs surrounding a specific clicked URL form its context. The contextual information is employed to disambiguate the categories of clicked URLs and thus the categories of queries. Of course, some URLs may not be covered in the ODP. We propose an approximate approach to deal with the ODP coverage problem.

[1] http://www.dmoz.org

This approach condenses an uncovered URL one level at a time and looks up the ODP to check if the condensed URL exists in the ODP. If exists, its ODP category is adopted. Otherwise, we condense one more level until either a match is found or a miss is reported. If it still misses, we use the query associated with the uncovered URL to find approximately ODP categories. The query is submitted to the ODP and the top 10 frequent paths are assigned to the URL.

For a URL, we consult the ODP to collect all possible paths. Because a clicked URL may belong to more than one path, we must find its correct meaning. The goal of category disambiguation is to find the most semantically-coherent path of each URL. Take Figure 2 as an example. There are six possible trails which consist of probable path from each clicked URL. The score of a trail is the sum of similarity between this path and the other paths in the session. The similarity of two paths is the number of common categories between these two paths. The number of common categories among paths reflects the degree of intent coherency. Therefore, a trail with the highest score will be selected, and the trail contains the disambiguated path for the clicked URLs.

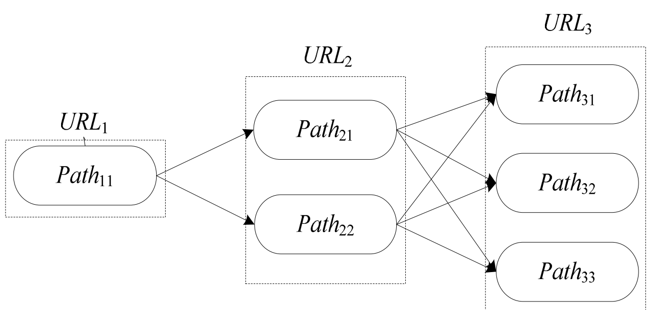

Figure 2. Category disambiguation procedure

C. Clustering Models

Two sessions are similar if the search behaviors in them are similar. Sessions of the same intent will be clustered together to reveal the common actions related to the intent. The group of sessions forms an *intent cluster*. Five sets of features, including queries *Q*, clicked URLs with/without considering relevance (abbreviated as *U* and *relU*, respectively), and ODP categories of clicked URLs with/without considering disambiguation (abbreviated as *C* and *DC*, respectively) are used to cluster sessions. Table 1 shows the details of each feature and feature combination.

In a query, terms are transformed to lower case, but not stemmed, and stop words are removed. Query terms are the first type of features. Here, a bag-of-words strategy is employed. That is, two queries consisting of the same terms in different orders are regarded as the same query. Queries may suffer from typo, but this issue is neglected in this work. A complete URL is the second type of features. The third type of features is similar to the second type except that the relevance of URL is considered. The relevance of URL determines the feature weight of *U*. A path is considered in the fourth and fifth types of features. We do not disambiguate the uses of the ODP categories of each URL in the former type. In contrast, only the best ODP categories are selected as features in the latter type. Different combinations of the above 5 types of features are shown from the 6^{th} -11^{th} rows.

TABLE I. DESCRIPTION OF FEATURE SETS

Features	Descriptions
Q	query terms as features
U	URLs as features
relU	relevant URLs as features
C	ODP categories of URLs as features
DC	disambiguated ODP categories of URLs as features
Q+U	query terms and URLs as features
Q+C	query terms and ODP categories as features
Q+ relU	query terms and relevant URLs as features
Q+DC	query terms and disambiguated ODP categories as features
Q+U+C	query terms, URLs and ODP categories as features
Q+ relU +DC	query terms, relevant URLs and disambiguated ODP categories as features

The weight of a feature (a query term, a URL, or a category) is determined by two schemes: binary or *tf-idf*. In the binary setting, the weight of a feature is set to 1 if the feature appears in the session, 0 otherwise. In the *tf-idf* setting, the weight of a feature is defined in Equation (2).

$$w_{i,s} = (0.5 + \frac{0.5 freq_{i,s}}{\max_s freq}) \times log \frac{N}{n_i} \tag{2}$$

Where $freq_{i,s}$ is the frequency of feature *i* in session *s*, $\max_s freq$ is the maximum feature frequency in session *s, N* is total number of sessions, and n_i is the number of sessions in which feature *i* appears. When the relevance of URLs is considered in a feature set, e.g., *relU* and *DC*, the weight of a feature is multiplied by *relP*(*q*,*u*), a relevance probability of URL *u* to query *q*.

The complete link and the average link hierarchical clustering algorithms with different sets of features are explored. Euclidean distance determines the similarity between two sessions.

V. SEARCH SCRIPT GENERATION

A search script consists of a sequence of actions to complete an information need. To identify the critical actions in an intent cluster, we first transform each session in an intent cluster into a transition graph, where a node denotes a disambiguated ODP category of a clicked URL and a directed edge from node *u* to node *v* means the clicked URL of *v* immediately follows the clicked URL of *u* in a chronological order. Initially, all the edges radiating outward from a node have the same weight. Then, transition graphs of all the sessions in the intent cluster are combined into one graph by merging nodes with the same ODP category. The ODP categories denote the purpose of actions in a search script. The weights for an edge are summed during merge. A search script is extracted from the graph, which contains actions to accomplish an information need.

The basic idea of action identification is to select the more significant nodes from a graph. We adopt PageRank [18], a link analysis algorithm widely used in search engines for assessing the importance of web pages, to grade the importance of nodes in a graph. The PageRank value measures the degree of significance of an action. A node with more in-links is more significant. In other words, the nodes with low PageRank values, which are less likely to be related to the main search intent of the intent cluster, will be filtered out. Algorithm 1 is proposed to select the critical actions based on the PageRank value.

This algorithm sorts the nodes in the descending order of their PageRank values and selects those nodes of PageRank values larger than a threshold. The threshold is determined by the largest PageRank value multiplying a weight R. It means the selected nodes must be authoritative enough to have at least a fixed percentage (R) of the largest PageRank value. Recall that nodes denote actions. Critical actions are proposed to form a search script by their selection order.

Algorithm 1. An action identification algorithm

Input: A graph G consisting of a set of nodes N and a set of directed edges E, a weight R

Output: Sub-graph(s) of G

$H \leftarrow \emptyset$ and $E'' \leftarrow \emptyset$

$n^* \leftarrow \text{argmax}_{n \in N}$ PageRankVal(N, E, n)

threshold $\leftarrow$ PageRankVal(N, E, n^*) $\times R$

$H \leftarrow \forall n \in N$, PageRankVal($N$, E, n) > threshold

Sort H and result in a sequence (n_1, …, n_m) where PageRankVal(N,E, n_j) >= PageRankVal(N, E, n_{j+1})

$i \leftarrow 1$

while (i < m)

  add (n_i, n_{i+1}) to E''

  $i \leftarrow i + 1$

end while

return H and E''

An action corresponds to an ODP category. In the ODP, human write down a natural language description for each ODP category. For understanding, each action is expressed by a natural language sentence rather than a category. Figure 1 shows a script example. Each action is marked as "*Search for* …", for example, the third action is "*Search for web sites of large hotel chains that operate on multiple continents*".

We summarize ODP category descriptions to represent actions. First, we use the Stanford part-of-speech tagger to tag the first sentence of a category description. Next, the first noun following the first verb (e.g., "*is*" and "*contains*") is identified. The entire phrase starting from the noun to the end of the sentence is extracted. For readability, we add "*Search for*" to the front of the extracted phrase to form an action description. For example, the first sentence of category description of the third action is "*This* category *is for central web sites of large hotel chains that operate on multiple continents*". The first noun of the sentence following the first verb (e.g., "*is*") is "web". Thus, "*Search for*" is added to the front of the phase (e.g., "*web sites of large hotel chains that operate on multiple continents*") to form a natural language description of the action.

VI. Experiments and Discussion

A. *Experimental Setup*

In this work, we aim to capture information need as complete as possible as. However, a smaller session may not cover the whole information, and thus has a completeness issue. In contrast, a longer session tends to introduce noise if more than one intent involves. How to trade off the completeness and the noiseless is important. We propose the following strategies to select sessions for intent clustering: (1) the duration between two continuous queries is no longer than 30 minutes; (2) at least 3 distinct queries exist; (3) there are no clicks to other search engines; (4) all clicked URLs can be found in ODP. Although only 17,277 sessions remain after our strict strategies, it is not a problem because huge collection of logs is available in the real world. More sessions of higher qualities will be selected if much more logs are available.

B. *Evaluating Quality of Intent Clusters*

Intent clustering aims to group sessions of similar intents into an intent cluster. We adopt an indirect approach to evaluate the quality of intent clusters generated by different models. Each intent cluster set is employed to identify the intent boundary of 1,000 sessions sampled from the MSN Search Query Log excerpt. The intent boundaries of these sessions are manually annotated by graduate students majoring in Computer Science. The performance of identification depends on the intent coherency of an intent cluster. The intent cluster set which gains the highest boundary identification accuracy contains the most intent-coherent clusters.

Table II shows the performance of two baseline approaches using time or query thresholds. An intent boundary is often segmented physically by users' actions such as open/close a browser, login/logout a search engine, etc., or some heuristic methods like time cutoffs [19][20] or mean session lengths [19][21]. AvgTime considers a segment spanning 20 minutes as an individual intent. Avg#Queries regards a segment consisting of 7 queries. AvgTime is the best baseline.

TABLE II. Using Time/Query Constraints

	Precision	Recall	F-Score
AvgTime	0.6368	0.6348	0.6355
Avg#Queries	0.6203	0.6195	0.6196

Table III shows the F-scores of various intent clustering models including (1) complete link+binary weighting, (2) complete link+ *tf-idf* weighting, (3) average link+binary weighting, and (4) average link+ *tf-idf* weighting. The boundary detection algorithm can refer to [22]. Using U features is better than using Q. It may be because clicked URLs express clearer users' intents, and users' queries may be ambiguous. Using C feature performs better than using U features. That meets our expectation because an ODP category is a conceptual representation of a URL. Using a Q together with U/C is better than using the U/C only. Using a Q together with U+C is better than using a Q together with U/C. Using *relU* features is better than using all features without considering URLs relevance. It indicates determining the relevance of URLs is very important. Using *DC* feature is better than using *relU*. It shows that the ODP category after disambiguation can represent the intent of clicked URL more clearly. Using Q together with *relU*/*DC* is better than using the *relU*/*DC* only as well. The intent clusters created by the clustering model integrating the features of Q, *relU* and *DC* in the complete link clustering algorithm perform the best. The clustering model achieving an F-score 0.6666 is significantly better than the two baselines in Table II (t-test, p-value<0.001). The best intent cluster will be used to generate search script and evaluated in the next paragraph.

TABLE III. PERFORMANCE OF INTENT CLUSTERING

Features	Model 1	Model 2	Model 3	Model 4
Q	0.6509	0.6540	0.6480	0.6500
U	0.6523	0.6542	0.6482	0.6502
C	0.6524	0.6550	0.6518	0.6523
Q+U	0.6553	0.6557	0.6525	0.6530
Q+C	0.6564	0.6559	0.6541	0.6543
Q+U+C	0.6567	0.6561	0.6553	0.6543
relU	0.6571	0.6573	0.6556	0.6546
DC	0.6577	0.6583	0.6558	0.6548
Q+relU	0.6597	0.6629	0.6569	0.6559
Q+DC	0.6625	0.6633	0.6585	0.6564
Q+relU+DC	0.6666	0.6643	0.6611	0.6610

C. *Evaluating Quality of Search Scripts*

The intent clusters generated automatically may contain noises. In other words, an intent cluster may contain several intents. The noise may come from sessions themselves and/or clustering errors. The intent purity of an intent cluster is evaluated as follows: (1) single intent: all actions are classified as the same intent; (2) major intent: more than half of actions correspond to the same intent; (3) multiple intents: more than one intent contains in an intent cluster, and neither of them dominates the intent cluster.

An action identification algorithm generates a search script for each intent cluster. To balance the evaluation loads, total 10 assessors are recruited in the user study. The search scripts to be examined are divided into 10 partitions. Because the evaluation is subjective, each search script will be examined by two different assessors.

A parameter R in Algorithm 1 affects how many actions will be selected from each intent cluster. In the extreme cases, all actions will be selected if R is 0, and only 1 action, which has the highest PageRank value, will be selected if R is 1. To determine suitable R, we randomly sample 20 intent clusters from a set of intent clusters containing more than 2 sessions, and evaluate the search scripts created by Algorithm 1 with different settings. Figure 3 demonstrates the number of scripts containing a single intent in relation to R. Algorithm 1 creates the largest number of search scripts containing a single intent when R is set to 0.7. We adopt this setting in the latter experiments.

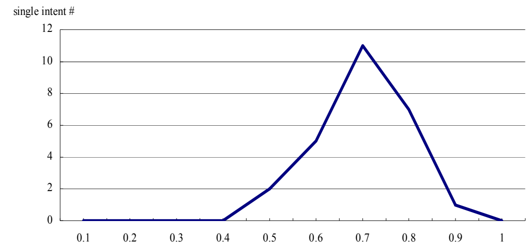

Figure 3. The number of scripts containing a single intent with different settings of R

The cost of human assessors to evaluate all the search scripts is very high. We propose two strategies to sample intent clusters for evaluation. An intent cluster which contains only 1 or 2 sessions cannot provide enough common user behavior to demonstrate the effects of our action identification algorithm. Besides, a search script consisting of 1 or 2 actions is more probable to contain a single intent. Thus, we select 1,017 intent clusters which contain more than 2 sessions and the search scripts generated from them consist of at least 3 actions.

Because search scripts created from the 1,017 intent clusters are examined by two different assessors, so that total 2,034 checks are conducted. Table IV shows the intent purity of search scripts. Over half of search scripts contain a single intent. Total 321 search scripts contain multiple intents, which occupy 15.78% of overall search scripts.

TABLE IV. INTENT PURITY OF SEARCH SCRIPTS

Purity	Single	Major	Multiple	Total
# of scripts	1,037 (50.98%)	676 (33.24%)	321 (15.78%)	2,034

We check inter-assessor agreement further. Table V shows agreements in purity assignments. Both assessors agree that 473, 271, and 139 search scripts contain a single intent, a major intent and multiple intents, respectively. Kappa value is 0.782, which is classified as substantial level.

There are two metrics to evaluate the performance. Let $I(x)$ denote the total number of search scripts annotated by two assessors with the same intent purity x, where $x \in$ {single, major, multiple}. Equations (3) and (4) define a strict and a lenient accuracy to evaluate the intent purity of search scripts, respectively. The strict accuracy, which counts single intents only as correct, is 0.4650. If we relax the measurement to include search scripts containing either the single or major intent, the accuracy increases to 0.7315 in the lenient metric.

$$strict\ accuracy_{intent} = \frac{I(single)}{Total\ scripts} \tag{3}$$

$$lenient\ accuracy_{intent} = \frac{I(single) + I(major)}{Total\ scripts} \tag{4}$$

TABLE V. INTER-ASSESSOR AGREEMENT

		Assessor 2		
		Single	Major	Multiple
Assessor 1	Single	473	43	0
	Major	48	271	26
	Multiple	0	17	139

We further analyze the topic distribution of search scripts (exclusive of pornography topic). The top 10 topics are: (1) shopping/recreation, (2) shopping/general merchandise, (3) computers/internet, (4) travel, (5) education, (6) financial services, (7) art/television, (8) news/newspapers, (9) relationships, and (10) health. The topic distributions generated from the intent clusters are consistent with those based on queries in the prior work [23].

VII. CONCLUDING REMARKS

In this paper, we learn users' searching and browsing experiences from the MSN Search Query Log excerpt, and generate search scripts of actions to guide users with the similar intents. Several models are proposed for intent clustering. Experiments show that the model integrating the features of query, relevance of clicked URLs and disambiguated categories in the complete link clustering algorithm is very useful to group similar intents. An action identification algorithm is proposed to find the critical actions in an intent cluster. We evaluate the search scripts from the aspects of intent consistency and purity. The user study shows that the proposed action identification algorithm is promising and the generated search scripts cover the most interesting topics compared with the topic distribution in other query logs.

Predicting users' intents as quickly as possible with the search script database, recommending suitable search scripts to shorten information seeking time and selecting proper websites in the search scripts for advertisements are plausible future works. In addition, users from different areas/countries may use different languages to express their intents. The search scripts may be various due to the culture difference. How to localize the search scripts to meet different life styles, cultures, and so on, has to be dealt with in the future.

VIII. ACKNOWLEDGMENTS

This research was partially supported by National Science Council, Taiwan under grant NSC99-2221-E-002-167-MY3. We are also grateful to Microsoft Research Asia for the support of MSN Search Query Log excerpt.

IX. REFERENCES

[1] A. Broder, "A taxonomy of web search," *SIGIR Forum*, vol. 36, no. 2, pp. 3-10, 2002.

[2] V. Nguyen and M. Kan, "Functional Faceted Web Query Analysis," in *Query Log Analysis: Social And Technological Challenges. A workshop at the 16th International World Wide Web Conference (WWW 2007)*, 2007.

[3] M. Manshadi and X. Li, "Semantic tagging of web search queries," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2*, pp. 861-869, 2009.

[4] X. Shen, S. Dumais, and E. Horvitz, "Analysis of topic dynamics in web search," in *Special interest tracks and posters of the 14th international conference on World Wide Web*, pp. 1102-1103, 2005.

[5] T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay, "Evaluating the accuracy of implicit feedback from clicks and query reformulations in Web search," *ACM Transactions on Information Systems*, vol. 25, no. 2, p. 7, 2007.

[6] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey, "An experimental comparison of click position-bias models," in *Proceedings of the international conference on Web search and web data mining*, pp. 87-94, 2008.

[7] F. Guo, C. Liu, and Y. M. Wang, "Efficient multiple-click models in web search," in *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pp. 124-131, 2009.

[8] F. Guo et al., "Click chain model in web search," in *Proceedings of the 18th international conference on World wide web*, pp. 11-20, 2009.

[9] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan, "Expected reciprocal rank for graded relevance," in *Proceeding of the 18th ACM Conference on Information and Knowledge Management*, pp. 621-630, 2009.

[10] R. Baeza-Yates, C. Hurtado, and M. Mendoza, "Query Recommendation Using Query Logs in Search Engines," in *Current Trends in Database Technology - EDBT 2004 Workshops*, 2005, pp. 588-596.

[11] Z. Zhang and O. Nasraoui, "Mining search engine query logs for query recommendation," in *Proceedings of the 15th international conference on World Wide Web*, pp. 1039-1040, 2006.

[12] D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log," in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 407-416, 2000.

[13] J. Wen, J. Nie, and H. Zhang, "Clustering user queries of a search engine," in *Proceedings of the 10th international conference on World Wide Web*, pp. 162-168, 2001.

[14] H. Cao et al., "Context-aware query suggestion by mining click-through and session data," in *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 875-883, 2008.

[15] N. Craswell, R. Jones, G. Dupret, and E. Viegas, "Proceedings of the 2009 workshop on Web Search Click Data," Barcelona, Spain, p. 95, 2009.

[16] S. Perugini, "Symbolic links in the Open Directory Project," *Information Processing and Management: an International Journal*, vol. 44, no. 2, pp. 910-930, 2008.

[17] W. Zhang and R. Jones, "Comparing Click Logs and Editorial Labels for Training Query Rewriting," in *Query Log Analysis: Social And Technological Challenges. A workshop at the 16th International World Wide Web Conference (WWW 2007)*, 2007.

[18] A. Altman and M. Tennenholtz, "Ranking systems: the PageRank axioms," in *Proceedings of the 6th ACM Conference on Electronic Commerce*, pp. 1-8, 2005.

[19] C. Silverstein, M. Henzinger, H. Marais, and M. Moricz, "Analysis of a Very Large AltaVista Query Log," *Technical Note: Digital Equipment Corporation*, 1998.

[20] A. L. Montgomery and C. Faloutsos, "Identifying Web Browsing Trends and Patterns," *Computer*, vol. 34, no. 7, pp. 94-95, 2001.

[21] B. J. Jansen, A. Spink, C. Blakely, and S. Koshman, "Defining a session on Web search engines: Research Articles," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 6, pp. 862-871, 2007.

[22] C. J. Wang, K. H. Y. Lin, and H. H. Chen, "Intent boundary detection in search query logs," in *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pp. 749-750, 2010.

[23] S. M. Beitzel, E. C. Jensen, A. Chowdhury, O. Frieder, and D. Grossman, "Temporal analysis of a very large topically categorized Web query log," *J. Am. Soc. Inf. Sci. Technol.*, vol. 58, no. 2, pp. 166-178, 2007.