A Novel Probabilistic Framework to Broaden the Context in Query Recommendation Systems

Dimitris Giakoumis and Dimitrios Tzovaras

Information Technologies Institute, Centre for Research and Technology Hellas (CERTH/ITI), 6th Km Charilaou-Thermi Road, 57001 (PO Box 361), Thermi-Thessaloniki, Greece
{dgiakoum,tzovaras}@iti.gr

Abstract. This paper presents a novel probabilistic framework for broadening the notion of context in web search query recommendation systems. In the relevant literature, query suggestion is typically conducted based on past user actions of the current session, mostly related to query submission. Our proposed framework regards user context in a broader way, consisting of a series of further parameters that express it more thoroughly, such as spatial and temporal ones. Therefore, query recommendation is performed herein by considering the appropriateness of each candidate query suggestion, given this broadened context. Experimental evaluation showed that our proposed framework, utilizing spatiotemporal contextual features, is capable to increase query recommendation performance, compared to state-of-art methods such as co-occurence, adjacency and Variable-length Markov Models (VMM). Due to its generic nature, our framework can operate on the basis of further features expressing the user context than the ones studied in the present work, e.g. affect-related, toward further advancing web search query recommendation.

Keywords: context-aware, web search, query recommendation, spatiotemporal features, probabilistic framework.

1 Introduction

Automatic query recommendation is essential for modern web search systems. Typically, such systems (e.g. Google) take as input keywords expressing the user's search intent. However, user queries are often extremely concise, consisting of few (1-2) words [1], and are significantly prone to ambiguity [2]. As a result, it is common for the initial query to be refined several times by the user [3], until a "proper" query is submitted, that is, a query expressing user intent in a way enabling the search engine (SE) to provide results optimally matching that intent, satisfying the user. The capability of the SE interface to assist the user in this process, through query recommendations, is highly significant [3]; in essence, by inferring the user's search intent, the SE can provide query recommendations so as to assist the user to find easier and earlier what s/he is looking for. It has been reported that a reduction of only 1% in the time spent from a user searching in Google could lead to saving more than 187.000 person-hours (21 years) each month [4]. Automatic query recommendation has thus become one of the most fundamental features of modern web search engines [5].

A. Likas, K. Blekas, and D. Kalles (Eds.): SETN 2014, LNAI 8445, pp. 589–602, 2014.

To this end, a plethora of automatic query recommendation systems have been proposed in the past (e.g. [2],[3],[6-9]). This line of research typically focuses on providing recommendations by comparing the queries that have been issued so far in the present user session, to queries of search sessions that have been recorded in the search engine's query logs. For instance, in [6], query recommendations were provided on the basis of the user input query's *co-occurrence* with other queries within sessions of the server's log. In [7], instead of taking into account all queries that co-occurred in logged sessions with the user's query, query substitution/expansion was performed on the basis of the queries that were found in the server's log to immediately follow the user input query (*adjacency* method). Extending this approach and building upon a Markovian Prediction Suffix Tree (PST) [15], the authors of [3] presented a query recommendation method based on Variable-length Markov Models (*VMM*), where the user input query, as well as the queries preceding it within the present user session were taken into account. In [17], a similar approach, based on variable length HMMs was proposed. Further recent query suggestion studies include [18], which focused on frequency of occurrence, keyword similarity measures and modification patterns of queries, and [19], where the focus was on the identification of conceptually related queries through click-through graphs. In general, methods like the above provide suggestions by calculating the probability of each candidate suggestion to be issued next by the user, given the user context in terms of the so far issued queries, under the model defined through the server's query logs [3]. Moreover, user actions related to clicks on URLs provided as search results have also been utilized [9],[19].

As sometimes denoted within the above approaches, their aim is to infer query recommendations on the basis of the "user context" [3],[9]; however, they operate on only a limited view of context, which regards only queries and clickthrough data. These methods will be referred in the rest of the paper as SoA (State-of-Art) methods. Their limited view of context results to recommendations that are same for a given query (or query sequence, clickthrough actions), regardless for e.g. the temporal or spatial context of different user sessions. Focusing on the early provision of helpful recommendations (e.g. optimally provided right after the first query submission), the issue of same recommendations regardless the spatiotemporal user context becomes of even higher importance. The same query may be used to express different information needs in the morning than in the evening, during weekdays or weekends [10], in the summer or winter, in different locations [11]. By not taking into account such information, a query recommender will always provide the same recommendation for a given input query or query sequence. Thus, although this limited view of user context can lead to recommenders that are effective up to an extent, it may as well lead to sub-optimal results in practical application scenarios.

As such a scenario, let us suppose the case of a tourist, using a web search system through his smartphone during midday, so as to find information regarding nearby restaurants in a town he is visiting (e.g. Thessaloniki), while being at a museum located at a suburb of that city. While wandering in that suburb, prior to visiting the museum, he has noticed that a number of fine-dining restaurants exist therein. Unfortunately, he recalls neither the names of those restaurants, nor the name of the suburb. Therefore, he provides "Thessaloniki restaurants" as a query to the search engine. The

SE web interface provides a large number of results, along with recommendations assisting the user in refining the query, so as to easier find what s/he is looking for.

However, if the above recommendations are based only on the query-oriented context of the user's web search session, they would be the same, regardless the time (e.g. midday, night) or specific location (e.g. city centre, a city suburb) of the user. Thus, since the "Thessaloniki restaurant" searches recorded in the SE log typically focus on taverns located at the city center, the recommendations will be rather irrelevant to the present user's actual needs. Suppose now that the above recommender system operated on the basis of a broadened view of context, taking into account, apart from the queries submitted in the present session, also the time of day, as well as the location of the user, boosting in the recommendations list, queries that have been provided to the SE from similar location and time as the one of the user. In this case, the name of the suburb would be highly probable to appear in the recommendations, along with terms such as "fine-dining restaurants", significantly reducing the time and effort needed from the user to find what he is looking for.

In the latter case of the above example, the recommender system, by operating on a broadened notion of the user context, would have been enabled to provide more personalized results, better suited to the present user's needs. Web search is highly important for tourists, whereas more personalized search, better adapted to their needs at the time of search engine usage are essential, given that relevant use cases typically regard searching information about places that the SE user visits for the first time. The above may as well hold in more generic use cases, where, by utilizing spatial, temporal and even further characteristics of the user context, a query recommender system could provide results better tailored to the user needs.

To this end, although context-aware recommender systems is a highly active research field [13], only few past works have studied a broadened notion of context in web search query recommendation, focusing for instance on location metadata [11] or temporal information [20]. However, the significance of broadening the context in query recommendation calls for novel methods that would allow incorporating spatial, temporal, as well as further types (e.g. affect-related) of contextual features.

1.1 Contribution

Following the above, the present work introduces a novel framework that allows spatiotemporal and potentially further (e.g. affect-related) features describing the user context to be included in the process of automatic query suggestion. To this end, it utilizes as a basis SoA query recommendation approaches that predict user search intent from past queries of the present search session, refining their results by taking into account a broader view of the user context. At its current implementation, our framework has been experimentally tested using spatial and temporal contextual features, so as to augment SoA query suggestion approaches such as co-occurrence, adjacency and VMM. Through experimental evaluation, our framework was found capable to increase the initial performance of the SoA methods. Due to its generic nature, our proposed framework can be potentially used in conjunction with practically any SoA query suggestion method that is based solely on session query terms or clickthrough data. Moreover, it can be extended toward incorporating further spatial, temporal or other contextual features, describing the user's context in a more thorough way.

1.2 Paper Outline

In the rest of the paper, Section 2 describes our probabilistic framework for broadening the context in query suggestion and Section 3 describes the process that was followed for experimental evaluation, along with the obtained results. Conclusions are drawn in Section 4.

2 The Query Recommendation Context Broadening Framework

If viewed as a black box, our proposed framework (Fig. 1) receives as input past queries of the present user session, as well as parameters that describe the session's spatiotemporal context. As shown in Fig. 1, by incorporating a SoA query recommendation method, it obtains candidate suggestions (SoA query suggestions), along with the probability that they would stand for the user's next submitted query. Thereafter, prior to providing the candidate suggestions to the user, ranking them via their probability score (as would have been typically done at this point in SoA query suggestion), our framework fuses for each candidate its current probability with the probability that the specific query would have been submitted to the search engine given the present spatiotemporal context (SoA query suggestions refinement). This final joint probability is eventually used so as to rank candidate suggestions and provide them to the search engine user (Final Query Suggestions).

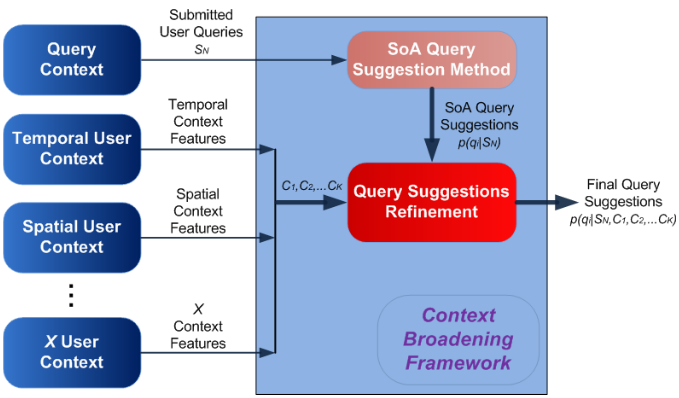

Fig. 1. Conceptual Architecture of the proposed context-broadening framework

For describing our approach in more detail, let us suppose a currently active user session in a web search system, which consists of N queries submitted so far. Let $S_N = \{q_1, q_2, \ldots, q_N\}$ be this submitted query sequence and a probabilistic query suggestion method (e.g. co-occurrence), which recommends M candidate queries q_i, $1<i<M$, on the basis of S_N. At this, typically concluding point of SoA query recommendation methods, the result is a list of the candidate queries, ordered by the probability that they will be issued next by the user given S_N, that is $P(q_i)=p(q_i|S_N)$.

Taking now a step forward from SoA, instead of providing query recommendations on the basis of the conditional probability $p(q_i|S_N)$, our framework takes into account S_N and also further features describing the user context. Therefore, if K is the number of the utilized contextual features C_k, the probability of q_i to be issued next is estimated within our framework as:

$$p(q_i \mid S_N, C_1, C_2, \ldots, C_K) \tag{1}$$

Equation 1 calculates the probability that q_i will be issued next by the user, on the basis of her/his past queries and also a series of further parameters that could potentially describe user context in a thorough way. For instance, C_k's can be parameters related to the spatial, temporal, or even the affective context of the user. They could also be used to encode user actions during search engine usage, i.e. clickthrough results selection data.

Using the Bayes theorem and assuming conditional independence between S_N, $C_1,\ldots,C_K$ for a given q_i, from Equation 1 it holds:

$$\begin{aligned} p(q_i \mid S_N, C_1, C_2, \ldots, C_K) &= \frac{p(q_i)\,p(S_N, C_1, \ldots, C_K \mid q_i)}{p(S_N, C_1, \ldots, C_K)} = \\ &\frac{p(q_i)\,p(S_N \mid q_i)\,p(C_1 \mid q_i)\ldots p(C_K \mid q_i)}{\sum_{j=1}^{M} p(q_j)\,p(S_N \mid q_j)\,p(C_1 \mid q_j)\ldots p(C_K \mid q_j)} = \\ &\frac{p(q_i)\dfrac{p(q_i \mid S_N)\,p(S_N)}{p(q_i)}\,p(C_1 \mid q_i)\ldots p(C_K \mid q_i)}{\sum_{j=1}^{M} p(q_j)\dfrac{p(q_j \mid S_N)\,p(S_N)}{p(q_j)}\,p(C_1 \mid q_j)\ldots p(C_K \mid q_j)} = \\ &\frac{p(q_i \mid S_N)\,p(S_N)\,p(C_1 \mid q_i)\ldots p(C_K \mid q_i)}{\sum_{j=1}^{M} p(q_j \mid S_N)\,p(S_N)\,p(C_1 \mid q_j)\ldots p(C_K \mid q_j)} \ldots \Rightarrow \end{aligned} \tag{2}$$

$$p(q_i \mid S_N, C_1, C_2, \ldots, C_K) = \frac{p(q_i \mid S_N)\,p(C_1 \mid q_i)\ldots p(C_K \mid q_i)}{\sum_{j=1}^{M} p(q_j \mid S_N)\,p(C_1 \mid q_j)\ldots p(C_K \mid q_j)} \tag{3}$$

In Equation 3, $p(q_i|S_N)$ describes the conditional probability that q_i will be issued next, given the so far user submitted queries S_N, which, as explained above, can be obtained through any probabilistic state-of-art query suggestion method, such as co-occurrence, adjacency etc. The terms $p(C_1|q_i)$,… $p(C_K|q_i)$ describe the conditional probabilities that the current value of each contextual parameter $C_1,\ldots C_K$ (describing the present session's context) is expected to hold, given that query q_i is submitted by the user to the search engine. For each candidate query q_i, the $p(C_k|q_i)$ terms, can derive from histograms expressing the distribution of q_i occurrences within the server's log, in respect of the possible values of each C_k parameter.

As mentioned above, Equations 2 and 3 are formed following Equation 1, under the assumption of conditional independence between the parameters for a given q_i. In essence, this means that optimally, knowledge over one's parameter value for a given query should have absolutely no effect over the probability distribution of the other parameters. This assumption calls for careful selection of the contextual parameters to be used in our framework. However, cases of suboptimal conditional independence between the parameters can as well be tolerated as an approximation [16], keeping in mind that they might induce negative effects on our framework's performance.

In the present study, the contextual parameters shown in Table 1 were considered, expressing aspects of the spatiotemporal context of the user during the web search session. The prior probabilities of queries, regarding the specific parameters, were estimated herein as described in the following.

Table 1. Contextual parameters used in the present study

Parameter Name	Description
TD	Time of Day [0-23]
DW	Day of Week [1-7]
DM	Day of Month [1-28]
MY	Month of Year [1-12]
LOC (CI, CO)	Location (City, Country)

2.1 Contextual Parameter Priors for Candidate Query Suggestions

Focusing first on temporal features of the user context, in the present study we considered the contextual parameters "time-of-day" (TD), "day-of-week" (DW), "day-of-month" (DM) and "month-of-year" (MY). For each temporal context parameter, we first calculated for each query (q_i) existing in the database, the histogram of query occurrences in respect of the different possible parameter values. From each histogram, we then estimated the *pdf* of the specific parameter, as a mixture of Gaussians, fitted to the priors through Expectation-Maximization (EM). In the present implementation, six GMM components were used for modeling the TD parameter, three for the DW, seven for the DM and three for the MY. An example of a query's histogram, related to the TD (time of day) temporal context parameter, as well as the respective resulted *pdf* are shown in Fig. 2. Fig. 2b shows the six GMM components of the TD parameter, which have been fitted through EM to the histogram of Fig. 2a. Fig. 3c shows the cumulative *pdf* of the Fig. 2b GMM components, which is the result of the components summation.

The amount of GMM components, as well as their initial distribution, was selected so as to provide EM with initial clusters that covered the entire span of the given parameter's values. It should be noted at this point that, as shown in Table 1, the DM parameter was defined to have its allowed values in the span [1, 28], by normalizing the actual month-dependent span of this parameter (e.g. 1-31 for January, 1-30 for April) in this, common for all months span, so as to consider day-of-month values over a common, month-independent reference.

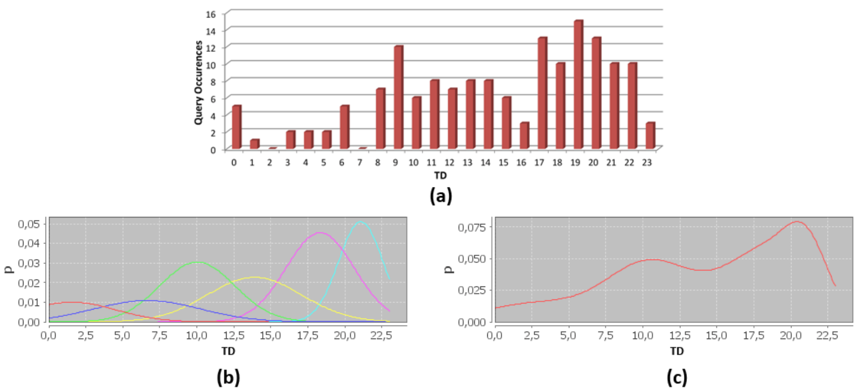

Fig. 2. (a) Example of query histogram from AOL dataset for the time-of-day (TD) parameter, (b) GMM components after EM, (c) cumulative *pdf*

In the context of query recommendation, data sparseness in the server query logs is a typical problem; the same query may have only a few (e.g. less than ten) occurrences. In our case, this issue can affect the afore-described pdf estimation process. The generation of smoother *pdfs* for queries of low occurrence, is achieved in the present study by employing Laplacian (add-one) smoothing. This leads, in cases of queries with low amount of occurrences, to *pdfs* which in fact describe a tendency of a query to appear more in specific times-of-day, days-of-month etc., without being over-fitted to the specific time-of-day etc. of the limited occurrences that have appeared in the logs. Of course, as the amount of query occurrences becomes higher, the role of this smoothing is suppressed, allowing the resulting *pdf* to depict the occurrence tendencies of queries in a more precise way. At this point, it should be noted that Laplacian smoothing can be considered as a rather simple approach to deal with data sparseness in probability density estimation. However, it was used herein so as to compare our framework to SoA methods, without boosting its performance with more sophisticated smoothing approaches. Nevertheless, more sophisticated smoothing approaches could as well be employed in the future within our proposed framework.

From the resulting *pdfs*, the conditional probabilities of Eq. 3 related to our temporal contextual features are estimated for any given query that is provided as suggestion from the utilized SoA method, expressing the probability of the current contextual parameter's value to hold, given the specific query. Therefore, for each candidate query q_i, the probabilities $p(C_{TD}|q_i)$, $p(C_{DW}|q_i)$, $p(C_{DM}|q_i)$, $p(C_{MY}|q_i)$ are calculated in respect of the TD, DW, DM and MY context parameters, so as to provide input to Eq. 3, toward determining the final, context-aware probability of each query suggestion.

In respect of spatial context parameters, the present study focused on the location (LOC) of the user during the web search session, in terms of her/his city (CI) and country (CO). To this end, the "MaxMind Geolite" country and city IP geolocation database[1] was utilized, enabling the detection of the user's country and city from her/his IP. Through this information, the geographical distribution of each query's occurrences was obtained, providing the prior probability that the city of the currently logged-in user could be the source of the given query; this enabled the calculation of the $p(LOC|q_i)$ term used in Eq. 3, as $p(CI|q_i)$, or as $p(CO|q_i)$ in cases where only the country of the user could be inferred from the IP. Laplacian smoothing was again applied, enabling the calculation of query probabilities for locations inexistent in the train set.

3 Experimental Evaluation

Our framework's effectiveness in enhancing automatic query suggestion was evaluated over two datasets, one of limited size, collected through a custom built web search engine interface, and the publicly available, anonymized version of the AOL query logs dataset [14]. Our aim was to compare our framework against baseline query suggestion methods, i.e. co-occurrence [1], adjacency [7] and VMM [15],[3], so as to examine whether the use of spatiotemporal contextual features would increase query suggestion performance. The datasets, experimental evaluation and the results obtained are explained in the following of the present section.

3.1 Datasets and Evaluation Process

CUBRIK Dataset. The "CUBRIK dataset" was collected through a custom built web search engine interface (Fig. 3), which had as purpose to provide users with fashion-oriented web images related to the user query. This interface allowed recording time-stamped user actions (submitted queries) along with their IPs during search engine usage. The dataset consisted of more than 1300 queries provided by 50 different users from 11 cities of 6 countries. Through this dataset, we were capable to evaluate both SoA query suggestion methods, as well as our framework, in a case of a small-size dataset. Moreover, by recording user IPs, we were capable to evaluate, apart from temporal, also spatial contextual features, over their capability to enhance automatic query suggestion through our proposed framework.

[1] http://dev.maxmind.com/geoip/legacy/geolite/

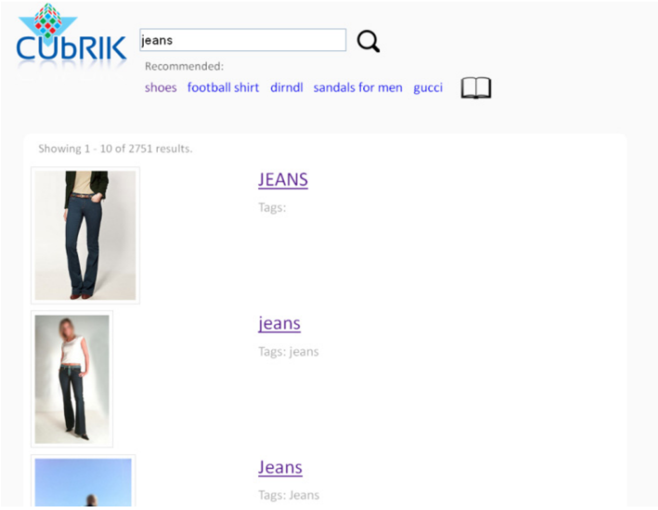

Fig. 3. The "CUBRIK fashion dataset" search engine interface

AOL Dataset. The publicly available, anonymized version of the AOL dataset [14] that was used in the present work, consists of web search query logs that were collected in 2006; it contains a sum of over 36 million queries, where the user IP has been replaced by an anonymous user ID.

Evaluation Process. In order to evaluate our method, we used it in order to re-rank, on the basis of several contextual features, the suggestion results of three state-of-art query suggestion methods, namely the co-occurrence, the adjacency and the VMM. In this context, our aim was to compare the effectiveness of our method, to the one of the SoA methods that were used to provide in each case the $p(q_i|S_N)$ term of our method's Eq. 3.

To this end, the first step was to split the queries of both our datasets into sessions. Sessions were identified based on changes on the username and by utilizing a temporal distance threshold between queries of the same user; herein, a five-minute threshold was used for splitting sessions, following [1] and [12], leading to over 2 million and 350 sessions for the AOL and CUBRIK datasets respectively.

In our experiments, following past works such as [11], we used the Mean Reciprocal Rank (MRR) as evaluation metric of query suggestion performance. The MRR expresses the inverse of the rank of the relevant query suggestion provided to the user, calculated as $MRR=1/rank(q_i)$, where, supposing that q_{i-1} is the last query submitted from the user in the present session (after which, a suggestion will be provided), q_i is the next query that is known (through the database log) to have been submitted from the user, serving as the ground truth of the test case and $rank(q_i)$ is the rank of q_i within the suggestions list provided from the query suggestion algorithm.

Since our framework operates on the basis of a SoA method, in essence re-ranking its results through contextual features, we omitted from the evaluation of both the

SoA methods and our proposed one, the cases for which the SoA method failed to provide a result relevant to the ground truth (q_{i+1}) within its suggestions list, as in [11]. The AOL dataset results presented below were obtained after randomly splitting the whole database into two equally sized parts, one used as the training set and the other for evaluation. Due to the limited size of the CUBRIK dataset, the respective results presented below were obtained through leave-one-out cross validation over the dataset's sessions. In order to reduce the noise that is introduced in the datasets from significantly rare queries and sessions, we followed the rationale of [3] and kept in the train and test sets respectively only those sessions that had more than one occurrence.

3.2 Experimental Results

CUBRIK Dataset. Table 2 presents the results that were obtained for the CUBRIK dataset through the co-occurrence (coOcc) and adjacency (Adj) methods, along with the results from using coOcc or Adj as the SoA method of our framework, for different combinations of utilized features of the temporal context. In particular, for a more thorough overview of the contextual features effect, Table 4 (as well as the rest of this section's tables) shows, apart from the results obtained by using all the spatiotemporal features (TD, DW, DM, MY and LOC) together and each one alone, also the best results taken by trying all different combinations of these features in pairs, triples or quadruples.

Table 2. Results (MRR) of the **co-occurrence** (coOcc) and **adjacency** (Adj) methods on the CUBRIK dataset, used alone and in conjunction with spatiotemporal features through our proposed framework

Method	MRR	Method	MRR
CoOcc	*16,56%*	*Adj*	*20,24%*
CoOcc {TD, DW, DM, MY, LOC}	16,16%	Adj {TD, DW, DM, MY, LOC}	16,24%
CoOcc {TD, DM, MY, LOC}	16,54%	Adj {TD, DM, MY, LOC}	18,49%
CoOcc {TD, MY, LOC}	16,96%	Adj {TD, MY, LOC}	21,31%
CoOcc {TD, LOC}	17,08%	Adj {MY, LOC}	21,34%
CoOcc {TD}	***17,96%***	**Adj {TD}**	***23,42%***
CoOcc {DW}	17,34%	Adj {DW}	15,81%
CoOcc {DM}	15,50%	Adj {DM}	19,54%
CoOcc {MY}	16,48%	Adj {MY}	21,13%
CoOcc {LOC}	16,50%	Adj {LOC}	17,05%

As shown inTable 2, by using a triplet of spatiotemporal contextual features, consisting of TD, MY and LOC, our framework provided an increase in MRR performance, for both the CoOcc and Adj methods. The best performance in both the coOcc and Adj cases was obtained by using our framework, so as to enhance query suggestion with information over the time-of-day context of the user (TD); the initial performance of coOcc and Adj was increased through our framework by 1.4% and 3.18% respectively.

Table 3 shows the respective results obtained over the CUBRIK dataset for the VMM method. By comparing the first row of Table 4 and Table 3, it is clear that in accordance to the results of [3], the VMM method outperformed both the Adj and coOcc in terms of MRR over our limited-sized CUBRIK dataset. Moreover, by augmenting through our framework the VMM-based query suggestion with the TD, DM, MY and LOC contextual features, an increase in MRR of 1.46% was obtained. The use of less contextual features further increased performance, whereas, the best result was again obtained by using our framework so as to augment VMM with information regarding the session's time-of-day.

Table 3. Results (MRR) of the **VMM** method on the CUBRIK dataset, used alone and in conjunction with temporal features through our proposed framework

Method	MRR
VMM	*22,76%*
VMM {TD, DW, DM, MY, LOC}	22,69%
VMM {TD, DM, MY, LOC}	24,22%
VMM {TD, DM, MY}	24,49%
VMM {TD, MY}	24,50%
VMM {TD}	**27,13%**
VMM {DW}	20,55%
VMM {DM}	21,81%
VMM {MY}	24,78%
VMM {LOC}	22,14%

As shown from the above results, due to the limited size of this dataset, all MRR results obtained were relatively low. However, in all the coOcc, Adj and VMM cases, significant increase in query suggestion performance was introduced by augmenting through our framework, their query suggestion process with spatiotemporal contextual features.

AOL Dataset. Table 4 presents the results that were obtained for the AOL dataset through the co-occurrence an the adjacency methods, along with the results obtained by using coOcc or the Adj as the SoA method of our framework, for different combinations of utilized features of the temporal context.

As shown in Table 4, by using all features of the temporal context within our framework, the average MRR for CoOcc was increased from 47.94% to 48.23%. By trying different combinations of features, the best result was obtained from using the time-of-day (TD) alone, which was 48.52%. In the adjacency method case, by using all features of the temporal context, a slight decrease in performance was obtained, at the level of 0.02%, however by omitting the DW parameter an increase of 0.09% was noticed and omitting also DM lead to further increase in performance. Again, the TD parameter alone led to the best MRR, which was 51.55%.

Table 4. Results (MRR) of the **co-occurrence** and **adjacency** methods on the AOL dataset, used alone and in conjunction with temporal features through our proposed framework

Method	MRR	Method	MRR
CoOcc	*47,94%*	*Adj*	*51,32%*
CoOcc {TD, DW, DM, MY}	48,23%	Adj {TD, DW, DM, MY}	51,30%
CoOcc {TD, DM, MY}	48,37%	Adj {TD, DM, MY}	51,41%
CoOcc {DM, MY}	48,42%	Adj {DM, MY}	51,49%
CoOcc {TD}	***48,52%***	**Adj {TD}**	***51,55%***
CoOcc {DW}	48,36%	Adj {DW}	51,40%
CoOcc {DM}	48,04%	Adj {DM}	51,27%
CoOcc {MY}	48,10%	Adj {MY}	51,23%

Finally, Table 5 shows the respective results obtained by using the VMM method as basis. Similarly to the above, by using all features of the temporal context, a slight decrease in performance was obtained, at the level of 0.01%, however by omitting the DW parameter an increase of 0.11% was noticed and omitting also DM lead to further increase (of 0.18%) in performance. Again, the TD parameter alone led to the best result for the VMM case, which was 51.72%. This was the best result that was obtained for this study's AOL dataset –based evaluation of automatic query suggestion.

Table 5. Results (MRR) of the **VMM** method on the AOL dataset, used alone and in conjunction with temporal features through our proposed framework

Method	MRR
VMM	*51,50%*
VMM {TD, DW, DM, MY}	51,49%
VMM {TD, DM, MY}	51,61%
VMM {DM, MY}	51,68%
VMM {TD}	***51,72%***
VMM {DW}	51,57%
VMM {DM}	51,44%
VMM {MY}	51,41%

Discussion. The above results, first of all confirm, in line with [3], the increased performance that can be obtained in query suggestion through the VMM method, compared to adjacency and co-occurrence. In both our datasets, the limited-sized one and the AOL, VMM outperformed the other SoA methods examined. Thereafter, through our proposed context-broadening framework, query suggestion performance, in terms of MRR, was found to further improve.

In the limited-size (CUBRIK) dataset, an increase of 4.37% in MRR was found by incorporating information regarding the session's time-of-day in query suggestion, compared to the VMM. In relative measures, the initial MRR of VMM (22,76%) was increased by a maximum of 19,20%. In the large dataset (AOL), an increase of 0.22% was obtained by incorporating this time only temporal contextual features in the VMM-based query suggestion process.

From the above analysis, the most important of our examined contextual features was found to be the TD, since its integration in the query suggestion process constantly led to the best MRR performance. This indicates that since the interests of web search engine users vary among different hours of the day [10], it is essential for query recommendation systems to augment their operating context with such information. Moreover, as shown from the CUBRIK dataset results, in accordance to [11], the user location parameter (LOC) was found also capable to enhance query suggestion performance in a number of cases, being used along with features of the temporal context. Unfortunately, the anonymized nature of the AOL dataset did not allow for features of the spatial context to be evaluated over it, however, the CUBRIK dataset results demonstrate a strong query suggestion enhancement potential for such features. Finally, focusing again on the AOL dataset results and the individual performance of temporal contextual features, the potential of the DW (day-of-week) parameter to enhance query suggestion performance is also evident, in accordance to differences that have been reported in search engine user interests between weekdays and weekends [10].

4 Conclusions

The present study introduced a novel probabilistic framework for broadening the context in automatic query suggestion. Our proposed framework takes a step forward from SoA methods that regard user context only in terms of queries that are submitted to the search engine, providing recommendations dependant also on spatiotemporal user context parameters. By evaluating our approach against the co-occurrence, the adjacency and VMM methods, we found increase in query recommendation performance, in terms of MRR. These results first of all demonstrate that our proposed framework, by broadening the context of query recommendation systems, has potential to increase their performance in future practical applications. In line with past works [10][11], the results of the present study further highlight the importance of incorporating spatiotemporal features in the query suggestion process. Moreover, due to its generic nature, our framework can incorporate in the future further contextual parameters than only spatiotemporal ones, e.g. affect-related, toward query recommendation systems that will operate in an even more broadened notion of the user context.

Acknowledgments. The present study was supported by the EU-funded research project CUBRIK.

References

1. Wen, J.-R., Nie, J.-Y., Zhang, H.-J.: Clustering user queries of a search engine. In: Proceedings of the 10th International Conference on World Wide Web (2001)
2. Cui, H., Wen, J.-R., Nie, J.-Y., Ma, W.-Y.: Probabilistic query expansion using query logs. In: Proceedings of the 11th International Conference on World Wide Web (2002)

3. He, Q., Jiang, D., Liao, Z., Hoi, S.C.H., Kuiyu, C., EePeng, L., et al.: Web Query Recommendation via Sequential Query Prediction. Paper presented at the IEEE 25th International Conference on Data Engineering, ICDE, March 29-April 2 (2009)
4. Qiu, F., Cho, J.: Automatic identification of user interest for personalized search. In: Proc. Fifteenth Int'l Conf. on World Wide Web (WWW 2006), pp. 727–736. ACM, New York (2006)
5. Kato, M., Sakai, T., Tanaka, K.: When do people use query suggestion? A query suggestion log analysis. Information Retrieval 16(6), 725–746
6. Huang, C.-K., Chien, L.-F., Oyang, Y.-J.: Relevant term suggestion in interactive web search based on contextual information in query session logs. Journal of the American Society for Information Science and Technology 54(7), 638–649 (2003)
7. Jones, R., Rey, B., Madani, O., Greiner, W.: Generating query substitutions. In: ACM WWW, pp. 387–396 (2006)
8. Meij, E., Bron, M., Hollink, L., Huurnink, B., de Rijke, M.: Learning Semantic Query Suggestions. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 424–440. Springer, Heidelberg (2009)
9. Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E., et al.: Context-aware query suggestion by mining click-through and session data. Paper Presented at the Proceedings of the 14th ACM SIGKDD Int'l. Conf. on Knowledge Discovery and Data Mining (2008)
10. Mei, Q., Church, K.: Entropy of search logs: how hard is search? with personalization? with backoff? Paper Presented at the Proceedings of the 2008 International Conference on Web Search and Data Mining (2008)
11. Bennett, P.N., Radlinski, F., White, R.W., Yilmaz, E.: Inferring and using location metadata to personalize web search. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (2011)
12. Silverstein, C., Marais, H., Henzinger, M., Moricz, M.: Analysis of a very large web search engine query log. SIGIR Forum 33(1), 6–12 (1999)
13. Adomavicius, G., Tuzhilin, A.: Context-Aware Recommender Systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 217–253. Springer, US (2011)
14. Pass, G., Chowdhury, A., Torgeson, C.: A picture of search. In: InfoScale 2006, p. 1 (2006)
15. Ron, D., Singer, Y., Tishby, N.: Learning probabilistic automata with variable memory length. In: COLT, pp. 35–46 (1994)
16. Zhang, H.: The optimality of naive Bayes. In: Proceedings of the 17th International FLAIRS Conference (FLAIRS2004). AAAI Press (2004)
17. Liao, Z., Jiang, D., Pei, J., Huang, Y., Chen, E., Cao, H., Li, H.: A vlHMM approach to context-aware search. ACM Trans. Web 7(4), Article 22 (2013)
18. Qumsiyeh, R., Ng, Y.K.: Assisting web search using query suggestion based on word similarity measure and query modification patterns. In: World Wide Web, pp. 1–20 (2014)
19. Goyal, P., Mehala, N., Bansal, A.: A robust approach for finding conceptually related queries using feature selection and tripartite graph structure. Journal of Information Science 39(5), 575–592 (2013)
20. Miyanishi, T., Sakai, T.: Time-aware structured query suggestion. In: Proc. of the 36th Int'l ACM SIGIR Conf. (SIGIR 2013) (2013)