10.31341/jios.47.1.7 UDC 025.4.03
Open Access Survey Paper

Query Refinement into Information Retrieval Systems: An Overview

Mawloud Mosbah *mos_nasa@hotmail.fr*
LRES Laboratory, Informatics Department
Faculty of Sciences
University 20 Août 1955 of Skikda, Algeria

Abstract

Query, expressing the user need and requirement, has an important role, in an information retrieval system, for reaching a high accuracy search. In this paper, we present an overview of the different refinement operations that the query may undergo, in the sake to enhance performance of an information retrieval system, such as: automatic query formulation through words prevision, query reformulation, query expansion, and query optimization.

Keywords: Information Retrieval, Query Processing, Automatic Query Formulation, Query Reformulation, Query Expansion, Query Optimization.

1. Introduction

Information retrieval consists to retrieve, from a large collection or corpus, a subset of documents relevant to the user requirement submitted as a textual query composed of some key-words. Information retrieval task is ensured commonly by automatic systems known as information retrieval systems in the case of a limited corpora and information retrieval engines, such as Google, where asking unlimited and extensible corpus in the image of the web.

An information retrieval system architecture is composed of three components: (i) indexing module, responsible to encode and represent the documents of the database to be asked in a compact and a significant form based on some discriminative features, (ii) interrogation module, designing the protocol and the language to be adopted to formulate the query reflecting the user information need and requirement, and (iii) the matching process, able to compare semantically between both pre-cited representations.

Unfortunately, the users are not well satisfied for the results returned by an information retrieval system which leads to think about improving these results and in consequence enhancing the global quality of the system. In retrospect of what happens in the past scientific researches, it is clear to observe that many efforts have been put for improving the quality of information retrieval systems through introducing new effective indexing signatures with high level semantics such as using of ontology and proposing new smart and intelligent reasoning modes for comparing between the documents and the submitted queries such as adopting of expert systems based on 'making the mind' approach and using neural networks relying on 'modelling the brain' paradigm. Moreover, the quality of information retrieval system may be improved also over the interrogation protocol either through the high interactivity with the user, relevance feedback mechanism as a post-processing step, or query refinement.

In this paper, we give an overview of query refinement, with its different aspects, in information retrieval system. Review of literature reveals that there exists some similar works, with different handling, that have addressed the chasm issue, in terms of vocabulary, between the queries submitted by the user and the existed documents. In [1], for instance, authors have presented an overview of the vocabulary gap in information retrieval between the queries and the documents through query rewriting scheme. They have categorized the existent works, in this direction, into two categories: works using shallow learning such as substitution-based and translation-based methods and works with deep learning like using of word embedding and deep reinforcement learning. In [2], authors have surveyed query expansion, query suggestion, and query refinement techniques with a detailed comparison between these three considered schemes. In terms of terminology, the authors have named query reformulation as a query refinement.

2. Query Expression and Interrogation Protocol

One of the major problems of information retrieval is the formulation of queries by the user. Indeed, two users formulate two different queries with different terms for referring to the same concepts of the same information requirement. Commonly, it is not easy, especially for a simple user, to formulate the appropriate query well reflecting the user information need and requirement. Unfortunately, in major of time, users do not know really what they are looking for. User requirement may be then tied to the information available in the asked collection, especially when the corpus is very large and rich such as in the case of the web, and the user requirement is progressively constructed over the contact with the initial results. According to [3], there are three categories of a query for web searching: (i) informational queries, covering a broad topic, (ii) navigational queries, looking for specific websites or URLs, and transactional queries, demonstrating the user's intent to execute a specific activity such as downloading papers or buying books. So, among the issues of information retrieval systems, from interrogation point of view, is the gap may exist between the user requirement and the user query. Owing to the position of query formulation, as a first step into information retrieval pipeline, it has, unfortunately, drastic effects on the quality of the information retrieval results. The richness of the interrogation language, considered frequently as a natural language with its ambiguity issue, as well as the gap between the vocabularies of the query and the collection, to be asked, may also be accused.

Unlike indexing step, performed offline, query expression and formulation, is established online, in ad hoc or in fly manner, which makes efficiency an important aspect to be taken into account although the littleness of the query compared to the documents corpus. Many works of the scientific literature has considered the interrogation process and query refinement as a module to be improved for enhancing the results of the information retrieval system. In the rest of this paper, we give an overview of the scientific works focusing on query refinement with its different aspects: formulation, auto-completion, reformulation, expansion, optimization and terms reweighting, and recommendation or suggestion.

3. Query Refinement into Information Retrieval Systems

In the hope to increase the accuracy and the effectiveness of information retrieval systems, including search engines, a lot of schemes have been adopted such as query refinement and relevance feedback. Both mechanisms, addressing respectively the system inputs and outputs, are going from outside the system in the purpose to inject obvious and more information helping the system to well understand the user information need and requirements. It is well recognized that formulating the query which represents really the user need and requirement is a difficult task especially from user view point. Accordingly, there is always a gap between the user requirement and the user query which has to be filled. Query refinement constitutes then an important scheme may help to fill this gap especially when the user is not satisfied for the returned results. This satisfaction comes from three situations: (i) the results are out of the scope of user requirement, (ii) the results are very vague where there is a lot of results, with general purpose, difficult to be explored, and (iii) the results are not enough required to be enriched and re-enforced. As shown in Figure 1, the query refinement, the scope of this paper, is implemented by three different manners regarding the involvement degree of the user: the manual way where there is a fully user intervention relying only on his/her experience and his/her own heuristics with no machine contribution, the interactive manner where the decision of terms selection is taken by the user with the assistance of the machine through suggesting some eventual terms to add, and the automatic alternative where the task is performed completely by the machine.

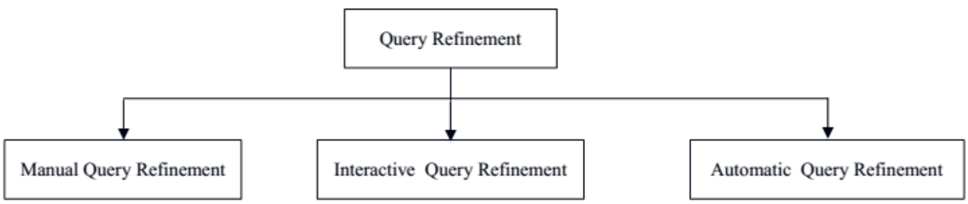

Figure 1. Query Refinement Types according the User Involvement.

Moreover, as presented in Figure 2, automatic query refinement has five materialization mays namely: automatic query formulation helping user to formulate his/her query, query reformulation which proceeds to reformulate the query completely, query expansion through adding some useful terms, query optimization for choosing the better terms and weights combination from the original query terms, query recommendation (or query suggestion) where information retrieval engine proposes a new query considered to be more adequate to the user intent. It is well recognized that the query formulation is coming to help user for expressing his/her information requirement especially for the users whose the intent is not yet clear while query reformulation has to be adopted in the case where the initial returned results are out of the scope of the user need. For query expansion, it is well suitable for the situation where the results are not enough to satisfy user requirement and needed to be enlarged while query optimization tries to narrow the vagueness of the returned results. Unfortunately, these query refinement types are confused in many works where the schemes are named with roughly speaking. Indeed, query reformulation, for instance, is restrained to query expansion such as in [1] where authors have combined, as a pipeline, query expansion with ranking model based on a neural network. The important contribution of the work introduced in [4] is that each module takes profit from the pseudo relevance feedback of the other. The pre-cited refinement schemes may be partially combined such as in [5] and [6] that combine expansion and terms re-weighting tied to query optimization. The five refinement mechanisms will be further detailed in the following sub-sections.

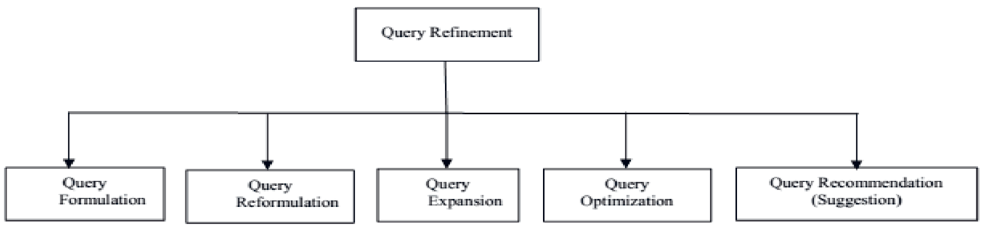

Figure 2. Query Refinement Implementations.

As depicted in Figure 3, there are two query refinement kinds: global query refinement and dedicated-query refinement. This classification is tied to the largeness and the narrowness of two criteria namely: the utilized resource, from where the knowledge is extracted, and the user for who the information retrieval system is addressed. A large resource means that the query refinement is tied globally to the adopted natural language while a resource with specific field refers to the fact that the query refinement is related to the terminology and the nomenclature of the addressed domain. On the other hand, a public user means that the query refinement is tied completely to the query without any additional information about the user while a specific user refers to the fact that the query refinement is adapted to the addressed user taking into account his/her past formulated queries as well as his/her profile.

4. Automatic Query Formulation based on Words Prediction (Query auto-completion)

Words prediction, widely used within text editors coming into digital devices with limited visualization screen and memory space such as smart-phones and tablets, means to guess the missing words that likely follow in a given segment of a text [7] in the purpose to get a desired piece of information quickly and with as little knowledge and effort as possible. Words prediction, for query formulation, depicted in Figure 4, commonly known as query auto-completion [8], refers to guess the words that the user intends to use or predict the possible completions for user queries in the aim to help the user to formulate his/her query, reduce query entry time, and potentially prepare the search results in advance of query submission [9].

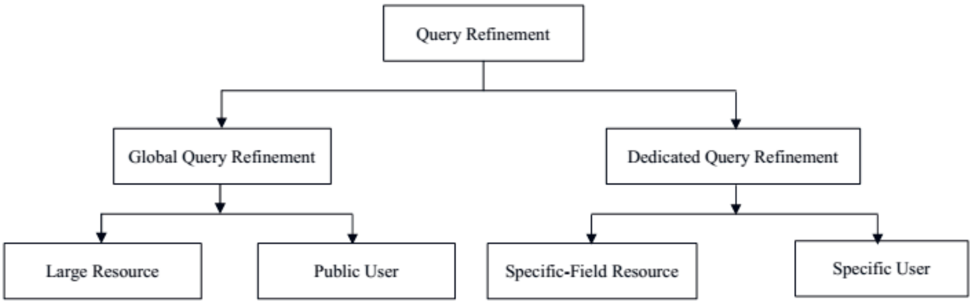

Figure 3. Query Refinement regarding the Largeness and the Narrowness of Utilized Resource and Addressed User.

Indeed, in query auto-completion, a user receives a ranked list of query candidates, by entering a particular prefix including characters or primary words [10]. In [11], authors have considered query auto completion, whose goal is to predict what the searcher is typing, and by extension, what the searcher is seeking based on the current query text string (prefix), as one form of query suggestion, while we consider it, here, as a separate task of query processing and refinement. Information retrieval engines, such as *Google* and its service dedicated for scientific searching that of *Google Scholar*, consider recently words prediction to ease query writing through guessing some words, suggested in a list, to the user, that might be used in that position. The space of possible completions narrows down and thus the prediction probability increases as the user is typing more characters. Commonly, there are three approaches for prediction [7] to help formulation of the query, namely: statistical modelling using word frequency and word sequence frequency, knowledge-based modelling adopting syntactic, semantic and pragmatic aspects, and heuristic or adaptive modelling with short and long learning. In addition, there are two alternatives for query auto-completion regarding the number of query terms implicated in auto-completion, namely: whole-query completions and term-by-term query auto-completion [12]. In [9], authors have compared 11 approaches for ranking candidate queries, in the sense of query auto-completion, namely: *Most Popular Ranker*, *Sentence Occurrence Ranker*, *Time Ranker*, *Most Popular Time Ranker*, *Term Occurrence Ranker*, *Near Words*, *String Similarity Ranker*, *WordNet Similarity Ranker*, *N-Gram Similarity Ranker*, *Kernel Similarity Ranker*, and *Clicked Documents Ranker*. Although *Clicked Documents Ranker* and *N-Gram Similarity Ranker* approaches seem to be more effective, authors concluded that the most effective approach to query auto-completion is largely dependent on the number of characters that the user has typed so far and that personalized information can be used to more effectively rank the query candidate completions. In [13], authors have addressed candidates ranking in query auto-completion, through considering homologous queries, which share the same terms but ordered differently, and the semantic relatedness of pairs of terms inside a query as well as pairs of queries inside a session. In addition, they have revealed that search engines adopt *Most Popular Ranker* approach for ranking candidate queries when they consider query auto-completion mechanism. For handling unseen prefixes, authors in [14], have proposed a recurrent neural language model that yielded good preliminary performances in a public dataset compared either to traditional methods, handling previously seen prefixes, or to the state of the art query auto completion methods for previously unseen prefixes. In [15], another alternative, for *Most Popular algorithm*, baptized as *NearestCompletion*, which is based on the user context, captured through recent queries and recently visited Web pages, has been introduced. Although the introduced algorithm outperforms clearly *Most Popular Ranker*, its performance tends, unfortunately, to zero when there is no user context. A hybrid algorithm, known as *HybridCompletion*, combining both *NearestCompletion* and *Most Popular Completion*, has been then proposed to tackle the absence of the context. Besides effectiveness character, authors, in [16], have discussed also the efficiency aspect of eBay's query auto completion system. Several optimizations, including the use of a forward index and Front Coding, are adopted.

4.1. Automatic Query Reformulation

Query reformulation, which may reflect the query refinement at all, is the process of altering an initial query through submitting another query with other terms. Although the main task of the query reformulation is to replace partially or totally the original query terms, it may include adding more words to the query (expansion), removing some words from the query (optimization), weighting the words according to its importance in the query, or all of them [17]. In addition, some structural or format aspects, such as stemming, pluralisation [18], acronym expansion, and spelling correction [19], are also addressed by query reformulation. In [20] and [21], authors have addressed manual query reformulation through trying to understand the user reasoning and reflection via examining changes in the logs of the past submitted queries of the same session. According to [22], the query reformulation is classified into two categories: *operational moves*, through reformulation using the same meaning, and *conceptual moves*, via changing the meaning of the query. In [23], authors have extended this categorization, for the case of the web, into four classes: *specification*, *generalization*, *replacement or error correction* as reported in [24], and *parallel movement* with eight types of patterns: *specified reformulation*, *generalized reformulation*, *parallel reformulation*, *building-block reformulation*, *dynamic reformulation*, *multi-tasking reformulation*, *recurrent reformulation*, and *format reformulation*.

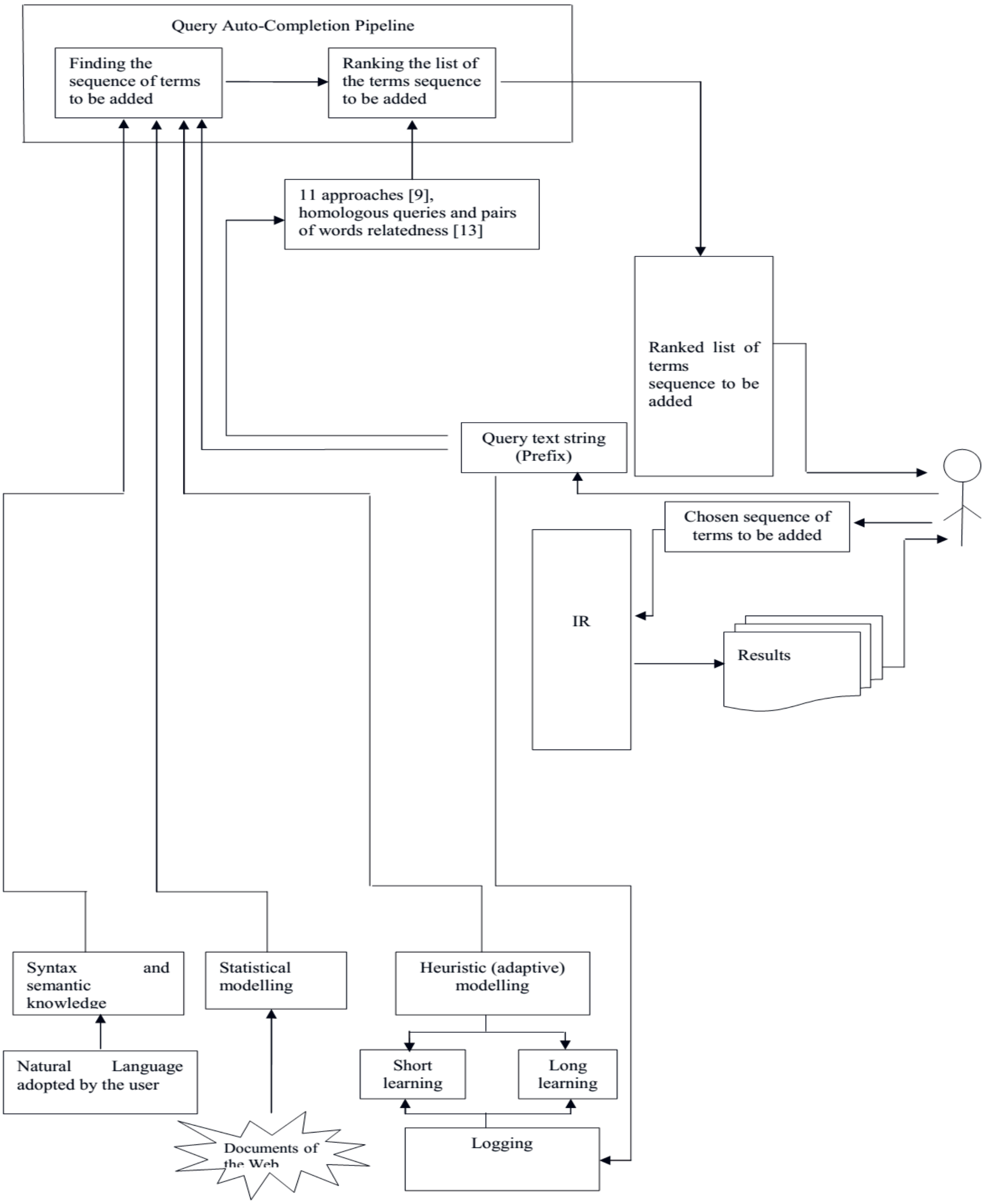

Figure 4. Query Auto-Completion Overview.

There are many resources used to extract terms for performing query reformulation. Using the top ranked documents returned by the system as an answer for the original query, such as what has been in [25], is one of them. Authors [25] have baptized the reformulation in this case as pseudo-query reformulation owing to its pseudo or blind relevance feedback base. However, the set of documents used as a source for extracting terms to reformulate the query may be designed interactively by the user as relevance feedback information such as in [26] where authors have revised the relevance feedback characterization ideas firstly introduced by Rocchio in [27]. Both pre-cited approaches belong to the local analysis using only a few of documents rather than the whole corpus used in the case of global analysis such as in [28], [29]. Note that the global analysis refers also to use the global context of a concept through considering semantic dictionaries such as WordNet [30]. In [28], authors have employed primitive concepts representing the variety of concepts whose the encoding vectors are orthogonal and that reflect the set of topics either locally into a set of relevant documents or globally into the entire corpus. The terms with which the query is8eformlated are extracted then from these primitive concepts on the basis of their own closeness and their combination closeness with the original query. In [31], authors have introduced a reformulation method based only on the original query. This reformulation method, relying on part of speech (POS), may be considered as an optimization for the simple reason that it searches the different terms combination.

4.2. Automatic Query Expansion

Query expansion [32], [33] consists of paraphrasing the concise original query, via adding additional concepts and terms, in order to reach a high level of recall through upgrading the possibility of the search system to get access to more relevant documents. Indeed, the term mismatch problem or vocabulary gap, produced when indexers and users utilize different vocabularies and terminologies because of the natural language richness, in addition to the weakness of the submitted query, composed of few words, for expressing the user need and requirement, as well as the ambiguity, owing to the posed polysemy issue, make query expansion an intuitive and logic operation to be established. Query expansion is performed automatically without any user involvement [32] or interactively through selecting a sub-set of terms to be added from the set of terms suggested by the machine [34]. Results given in [35], where authors have considered automatic and interactive query expansion, reveals the superiority of interactivity, helping to increase recall and especially limiting precision downgrading. Consulting the literature reveals that there are two large approaches for implementing automatic query expansion: (1) statistical approach [36] in the image of adding co-occurred and collocated terms, into different granularity level of windows, using information theoretic measures like mutual information [37], [38], Z score [39], hyperspace analogue to language (HAL) [40] and its quality property notion reflecting relatively the word context, or information flow [41] viewing the entire query as one concept which suggests other terms to be added for expansion based on HAL measure, and (2) semantic approach [42], whose the main issue is word disambiguation, like adopting association rules [34], [35], [43], word embedding [44], based commonly in advanced machine learning techniques, and related linguistic words with closed meaning based on morphological expansion [45] using stemming, part of speech, and synonyms [35], [46] using ontology [47] with its both types: narrow or dependent-domain, and general or independent-domain. Query expansion may be combined with other schemes such as in [48] where the author has combined query expansion with pseudo re-ranking based on voting algorithm.

Unfortunately, improving the recall metric value may be achieved in the count of precision aspect which imposes us to be careful when thinking about adopting query expansion scheme through checking its usefulness regarding the submitted query. Moreover, some difficulties and obstacles tied to the vagueness and the multi-contextual character of natural language, such as polysemy, should be also overcome and tackled.

For pinpointing the query expansion mechanism, we should see it as a pipeline of steps (as depicted in Figure 5) ranges from data acquisition and pre-processing to candidate generation, ranking, and selection.

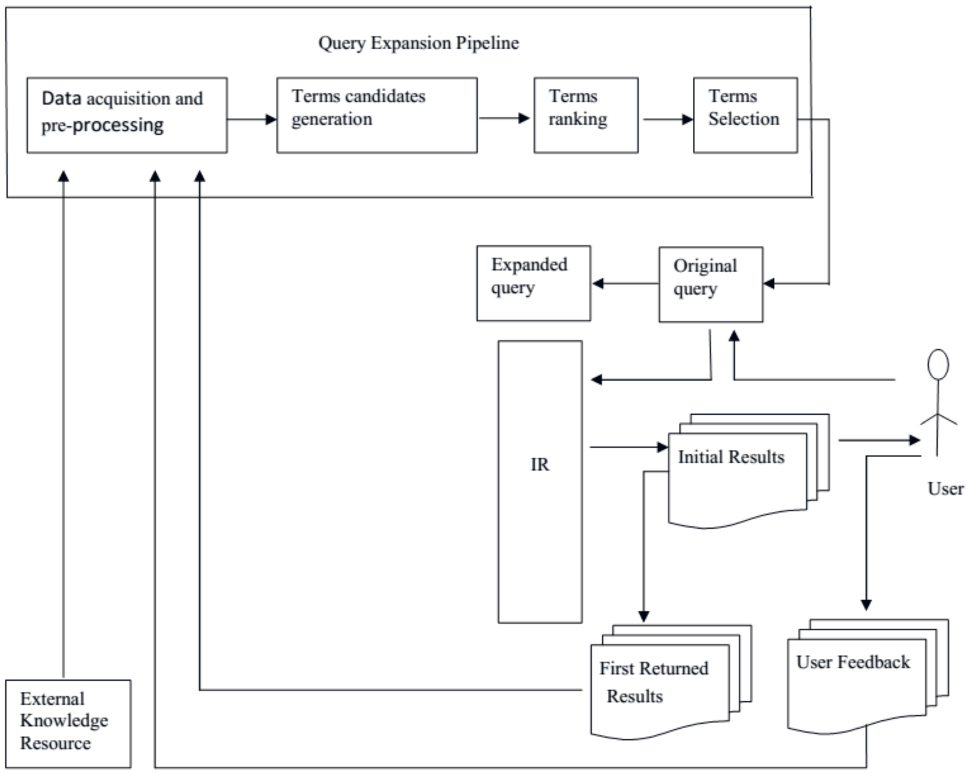

Figure 5. Query Expansion Overview.

The terms to be added for the initial query are, commonly, the discriminative terms generated and selected from the collection itself via global [38], [44] or local dependent-corpus knowledge approach [49] either with the making use of local context analysis (LCA) [50] which uses the highly ranked documents qualified as relevant by the search system with adopting of pseudo relevance feedback [51] or interactively employing user judgment through relevance feedback information [52] injected by the user himself. For avoiding performance degradation especially when relying on pseudo-relevance feedback, automatic documents clustering may be extensively useful such as in [53]. Employing independent-corpus knowledge approach or external knowledge resources, such as other documents set [54], thesaurus [55], DBpedia [56], Freebase [57], ConceptNet [58], Wikipedia [59], or WordNet [35], [59], [60], is another adopted alternative for generating query expansion terms. The past submitted user queries [54] for the system are also utilized with long-term query logs way either to directly extract terms adopting queries-based analysis approach or indirectly through extracting terms to be added from the highly associated documents adopting document-based analysis approach. These both lastly pre-cited alternatives can be used together to compute the correlation between query terms and their associated returned document terms, for the documents consulted by the user, adopting probabilistic model [61]. This probabilistic model has been implemented differently in [62] with the assumption that starting from the query, the next term to add is tied only to the previous added term, we speak here about Markov chain model.

Unfortunately, the expanded query may degrade information retrieval system performance in some cases especially when relying on non-relevant documents leading to add some terms not semantically associated to the entire sense or the holistic context of the query. In order to get around susceptible effectiveness degradation, some works go in the context of selective query expansion [63] while others adopt collection enrichment [64] through enrich the documents from where the new terms to add are extracted especially in the case of pseudo relevance feedback where the top-ranked documents are poor in information .

4.3. Automatic Query Optimization

As depicted in Figure 6, query optimization, or query reduction, consists of choosing the better sub-set of terms combination from those belonging to the long submitted query as well as their associated weights [65]. In other words, some superfluous terms, from the original query, have to be down-weighted or completely deleted and removed in the hope to obtain better results [66]. In [66], authors have presented query reduction techniques for long web queries that leverage effective and efficient query performance predictors. In [67], authors have provided an algorithm for predicting which word should be deleted from the query. According to [68], the first stage of terms combination is known as quantity control while the second stage, for designating the optimal weights, is referred to as quality optimization. This optimization of the original query is performed based on the distribution of terms into returned relevant documents (pseudo relevance feedback) as well as the distribution in the whole asked collection. As the query optimization issue is qualified as an optimization problem, tools such as genetic algorithm [65], [68], and particle swarm optimization (PSO) [69] have been utilized.

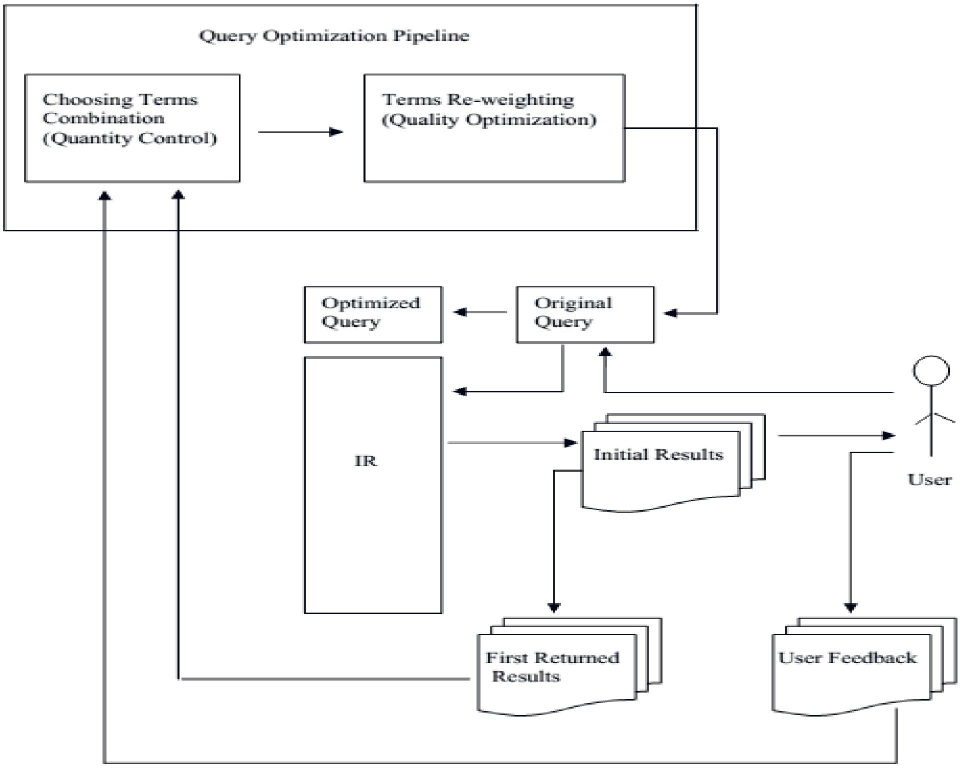

Figure 6. Query Optimization Overview.

4.4. Query Recommendation (Query Suggestion)

As depicted in Figure 7, query recommendation, or query suggestion, commonly provided by information retrieval engines such as Google and its search scientific service Google Scholar, refers to the process of revising the original query through generating, proposing and suggesting a list of related and alternative queries to those submitted by the users. Query recommendation is recently considered as an effective, efficient, and practical way to help and assist users to formulate their queries and consequently aid information retrieval systems to well understand the users' information need let alone to guess the next information requirement. Query suggestion considers, more or less, the slogan "recognition rather than recall" adopted previously by the graphical interfaces in systems interaction field. Indeed, recall is difficult and error-prone while recognition, with navigational character, is so simple and practical.

There are two major issues for query recommendation: (i) how to find the queries to be recommended and (ii) how to select, rank and order them. Some works of the literature address then this first issue, such as in [70], while the rest deal with the second issue of ranking the list of the proposed queries. Commonly, there exist two broad approaches for designating the set of queries to be suggested and recommended:

(i) techniques using session logs, exploiting the user past submitted queries and his/her historical behavioral interactivity let alone what we call 'wisdom of crowds', the knowledge mined from search engine query logs which store all the past interactions of users with the search system, and (ii) knowledge-based query recommendation using some resources such as YAGOO [71] and Freebase [71].

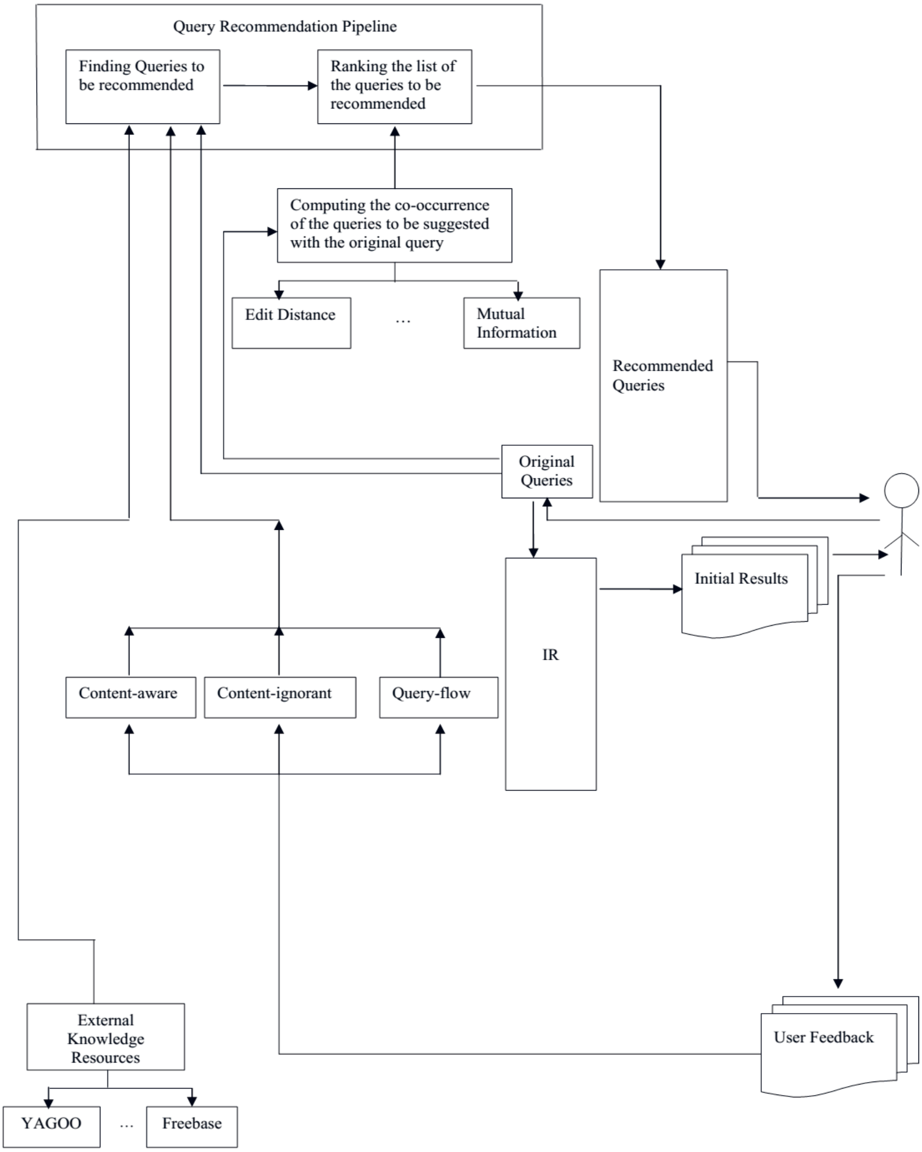

Figure 7. Query Recommendation Overview.

According to [72], there are three approaches for generating query recommendations based on mining query logs: (i) content-aware approaches, relying on search results or target pages, (ii) content-ignorant approaches that use just common clicked URLs, and (iii) query-flow approaches that consider the users' sequential search behaviour to better understand query intent. For ranking the suggested queries, researchers, such as in [73], consider commonly scores based on the co-occurrence of the queries to be suggested with the original submitted query. this co-occurrences is generally implemented using edit distance and mutual information.

As a personal critic, we do not agree, with scientific literature, that query recommendation and query suggestion are synonyms. We consider that query suggestion should have the outside character and then be based on the user interactivity relying on his/her previous submitted queries as well as words prediction tied to the considered natural language while query recommendation must has the inside aspect and to be based on the available documents to be asked and their vocabulary.

5. Summary and Conclusion

In this paper, we give an overview of some mechanisms handling a user query in the sake to well understand user requirement, namely: query auto-completion, query expansion, query optimization, query reformulation, and query suggestion or recommendation. From its graphical user interface, it is clear that Google and its service namely Google Scholar adopt, at least, query auto-completion and query suggestion. Many works of scientific literature, adopting query processing through one of the considered schemes, for information retrieval system improvement and enhancement, have been addressed here. Unfortunately, some addressed works confuse between the different query processing schemes where they are, roughly speaking, adopted. This work may be then a reference for everyone who deals with the interrogation protocol in information retrieval systems including information retrieval engines addressing the web as an unlimited and an extensible documentary collection.

References

[1] Liu, H., Yin, D., & Tang, J. (2020). Query rewriting. In Query Understanding for Search Engines (pp. 129-144). Springer, Cham.

[2] Ooi, J., Ma, X., Qin, H., & Liew, S. C. (2015, August). A survey of query expansion, query suggestion and query refinement techniques. In 2015 4th International Conference on Software Engineering and Computer Systems (ICSECS) (pp. 112-117). IEEE.

[3] Kang, I. H., & Kim, G. (2003, July). Query type classification for web document retrieval. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 64-71).

[4] Li, X., Mao, J., Ma, W., Wu, Z., Liu, Y., Zhang, M., Ma, S., Wang, Z., & He, X. (2022, February). A Cooperative neural information retrieval pipeline with knowledge enhanced automatic query reformulation. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (pp. 553-561).

[5] Diao, L., Yan, H., Li, F., Song, S., Lei, G., & Wang, F. (2018). The research of query expansion based on medical terms reweighting in medical information retrieval. EURASIP Journal on Wireless Communications and Networking, 2018(1), 1-7.

[6] Karisani, P., Rahgozar, M., & Oroumchian, F. (2016). A query term re-weighting approach using document similarity. Information Processing & Management, 52(3), 478-489.

[7] Ghayoomi, M., & Momtazi, S. (2009, October). An overview on the existing language models for prediction systems as writing assistant tools. In 2009 IEEE International Conference on Systems, Man and Cybernetics (pp. 5083-5087). IEEE.

[8] Cai, F. & De Rijke, M. (2016). A survey of query auto completion in information retrieval. Foundations and Trends in Information Retrieval. 10(4), (pp. 273-363).

[9] Di, Santo, G., McCreadie, R., Macdonald, C., & Ounis, I. (2015, August). Comparing approaches for query autocompletion. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 775-778).

[10] Tahery, S., & Farzi, S. (2020). Customized query auto-completion and suggestion-A review. Information Systems, 87, 101415.

[11] Smith, C. L., Gwizdka, J., & Feild, H. (2017). The use of query auto-completion over the course of search sessions with multifaceted information needs. Information Processing & Management, 53(5), 1139-1155.

[12] Vargas, S., Blanco, R., & Mika, P. (2016, February). Term-by-term query auto-completion for mobile search. In Proceedings of the Ninth ACM International Conference on Web Search and Data Mining (pp. 143-152).

[13] Cai, F., & de Rijke, M. (2016). Learning from homologous queries and semantically related terms for query auto completion. Information Processing & Management, 52(4), 628-643.

[14] Park, D. H., & Chiba, R. (2017, August). A neural language model for query auto-completion. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 1189-1192).

[15] Bar-Yossef, Z., & Kraus, N. (2011, March). Context-sensitive query auto-completion. In Proceedings of the 20th International Conference on World Wide Web (pp. 107-116).

[16] Gog, S., Pibiri, G. E., & Venturini, R. (2020, July). Efficient and effective query auto-completion. In Proceedings of the 43th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 2271-2280).

[17] Sankhavara, J., Dave, R., Dave, B., & Majumder, P. (2020). Query specific graph-based query reformulation using UMLS for clinical information access. Journal of Biomedical Informatics, 108, 103493.

[18] Teevan, J., Adar, E., Jones, R., & Potts, M. A. (2007, July). Information re-retrieval: Repeat queries in Yahoo's logs. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 151-158).

[19] Huang, J., & Efthimiadis, E. N. (2009, November). Analyzing and evaluating query reformulation strategies in web search logs. In Proceedings of the 18th ACM Conference on Information and Knowledge Management (pp. 77-86).

[20] Sloan, M., Yang, H., & Wang, J. (2015). A term-based methodology for query reformulation understanding. Information Retrieval Journal. 18(2), 145-165.

[21] Rha, E., Y., Shi, W., & Belkin, N. J. (2017). An exploration of reasons for query reformulations. Proceedings of the Association for Information Science and Technology, 54(1), 337-346.

[22] Fidel, R. (1985). Moves in online searching. Online Review, 9(1), pp. 61-74.

[23] Rieh, S. Y. (2006). Analysis of multiple query reformulations on the web: the interactive information retrieval context. Information Processing & Management, 42(3), 751-768.

[24] Boldi, P., Bonchi, F., Castillo, C., & Vigna, S. (2009, September). From "dango" to "Japanese cakes": query reformulation models and patterns. In 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (Vol. 1, pp. 183-190). IEEE.

[25] Diaz, F. (2016, March). Pseudo-query reformulation. In European Conference on Information Retrieval (pp. 521-532). Springer, Cham.

[26] Belkin, N. J., Cool, C., Kelly, D., Lin, S. J., Park, S. Y., Perez-Carballo, J., & Sikora, C. (2001). Iterative exploration, design and evaluation of support for query reformulation in interactive information retrieval. Information Processing & Management, 37(3), 403-434.

[27] Rocchio, J. (1971). Relevance feedback in information retrieval. The Smart Retrieval System Experiments in Automatic Document Processing, 313-323.

[28] Chang, Y., Ounis, I., & Kim, M. (2006). Query reformulation using automatically generated query concepts from a document space. Information Processing & Management, 42(2), 453-468.

[29] Zazo, A. F., Figuerola, C. G., Berrocal, J. L. A., & Rodriguez, E. (2005). Reformulation of queries using similarity thesauri. Information Processing & Management, 41(5), 1163-1173.

[30] Al-Khateeb, B., Al-Kubaisi, A. J., & Al-Janabi, S. T. (2017, March). Query reformulation using WordNet and genetic algorithm. In 2017 Annual Conference on New Trends in Information & Communications Technology Applications (NTICT) (pp. 91-96). IEEE.

[31] Lioma, C., & Ounis, I. (2008). A syntactically-based query reformulation technique for information retrieval. Information Processing & Management, 44(1), 143-162.

[32] Carpineto, C., & Romano, G. (2012). A survey of automatic query expansion in information retrieval. ACM Computing Surveys (CSUR), 44(1), 1-50.

[33] Azad, H. K., & Deepak, A. (2019). Query expansion techniques for information retrieval: a survey. Information Processing & management, 56(5), 1698-1735.

[34] Fonseca, B. M., Golgher, P., Pôssas, B., Ribeiro-Neto, B., & Ziviani, N. (2005, October). Concept-based interactive query expansion. In Proceedings of the 14th International Conference on Information and Knowledge Management (pp. 696-703).

[35] Abbache, A., Meziane, F., Belalem, G., & Belkredim, F. Z. (2018). Arabic query expansion using wordnet and association rules. In Information Retrieval and Management: Concepts, Methodologies, Tools, and Applications (pp. 1239-1254). IGI Global.

[36] Raza, M. A., Mokhtar, R., & Ahmad, N. (2019). A survey of statistical approaches for query expansion. Knowledge and Information Systems, 61(1), 1-25.

[37] Komiya, K., Abe, Y., & Kotani, Y. (2013, June). Query expansion using mutual information between words in question and its answer. In Proceedings of the ITI 2013 35th International Conference on Information Technology Interfaces (pp. 289-294). IEEE.

[38] Mazari, A. C., & Koudja, M. A. Concept based query expansion to improve Arabic information retrieval. Discovery and Data Analysis (KDDA'2015) 15-17 November 2015, ESI, Algiers, Algeria.

[39] Vechtomova, O., Robertson, S., & Jones, S. (2003). Query expansion with long-span collocates. Information Retrieval, 6(2), 251-273.

[40] Biancalana, C., Lapolla, A., & Micarelli, A. (2008, May). Personalized web search using correlation matrix for query expansion. In International Conference on Web Information Systems and Technologies (pp. 186-198). Springer, Berlin, Heidelberg.

[41] Bruza, P. D., & Song, D. (2002, November). Inferring query models by computing information flow. In Proceedings of the Eleventh International Conference on Information and Knowledge Management (pp. 260-269).

[42] Raza, M. A., Mokhtar, R., Ahmad, N., Pasha, M., & Pasha, U. (2019). A taxonomy and survey of semantic approaches for query expansion. IEEE Access, 7, 17823-17833.

[43] Latiri, C., Haddad, H., & Hamrouni, T. (2012). Towards an effective automatic query expansion process using an association rule mining approach. Journal of Intelligent Information Systems, 39(1), 209-247.

[44] Fernàndez-Reyes, F. C., Hermosillo-Valadez, J., & Montes-y-Gomez, M. (2018). A prospect-guided global query expansion strategy using word embeddings. Information Processing & Management, 54(1), 1-13.

[45] Moreau, F., Claveau, V., & Sébillot, P. (2007, April). Automatic morphological query expansion using analogy-based machine learning. In European Conference on Information Retrieval (pp. 222-233). Springer, Berlin, Heidelberg.

[46] Lu, Y., Fang, H., & Zhai, C. (2009). An empirical study of gene synonym query expansion in biomedical information retrieval. Information Retrieval, 12(1), 51-68.

[47] Bhogal, J., MacFarlane, A., & Smith, P. (2007). A review of ontology based query expansion. Information Processing & Management, 43(4), 866-886.

[48] Mosbah, M. (2018). Improving the results of Google Scholar through Automatic Query Expansion Mechanism and Pseudo Re-ranking using MVRA. Journal of Information and Organizational Sciences, 42(2), 219-229.

[49] Vechtomova, O., Robertson, S., & Jones, S. (2003). Query expansion with long-span collocates. Information Retrieval, 6(2), 251-273.

[50] Xu, J., & Croft, W. B. (2000). Improving the effectiveness of information retrieval with local context analysis. ACM Transactions on Information Systems (TOIS), 18(1), 79-112.

[51] Keikha, A., Ensan, F., & Bagheri, E. (2018). Query expansion using pseudo relevance feedback on Wikipedia. Journal of Intelligent Information Systems, 50(3), 455-478.

[52] Lin, H. C., Wang, L. H., & Chen, S. M. (2006). Query expansion for document retrieval based on fuzzy rules and user relevance feedback techniques. Expert Systems with Applications, 31(2), 397-405.

[53] Khennak, I., Drias, H., Kechid, A., & Moulai, H. (2019, September). Clustering algorithms for query expansion based information retrieval. In International Conference on Computational Collective Intelligence (pp. 261-272). Springer, Cham.

[54] Billerbeck, B., Scholer, F., Williams, H. E., & Zobel, J. (2003, November). Query expansion using associated queries. In Proceedings of the twelfth International Conference on Information and Knowledge Management (pp. 2-9).

[55] Mandala, R., Tokunaga, T., & Tanaka, H. (2000). Query expansion using heterogeneous thesauri. Information Processing & Management, 36(3), 361-378.

[56] Dahir, S., Khalifi, H., & El Qadi, A. (2019, Marh). Query expansion using DBpedia and WordNet. In Proceedings of the ArabWIC 6th Annual International Conference Research Track (pp. 1-6).

[57] Xiong, C., & Callan, J. (2015, September). Query expansion with freebase. In Proceedings of the 2015 International Conference on the Theory of Information Retrieval (pp. 111-120).

[58] Bouchoucha, A., He, J., & Nie, J. Y. (2013, October). Diversified query expansion using conceptnet. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management (pp. 1861-1864).

[59] Azad, H. K., & Deepak, A. (2019). A new approach for query expansion using Wikipedia and WordNet. Information sciences, 492, 147-163.

[60] Zhang, J., Deng, B., & Li, X. (2009, March). Concept based query expansion using wordnet. In 2009 International e-Conference on Advanced Science and Technology (pp. 52-55). IEEE.

[61] Cui, H., Wen, J. R., Nie, J. Y., & Ma, W. Y. (2002, May). Probabilistic query expansion using query logs. In Proceedings of the 11th International Conference on World Wide Web (pp. 325-332).

[62] Collins-Thompson, K., & Callan, J. (2005, October). Query expansion using random walk models. In Proceedings of the 14th ACM International Conference on Information and Knowledge Management (pp. 704-711).

[63] Cronen-Townsend, S., Zhou, Y., & Croft, W. B. (2004, November). A framework for selective query expansion. In Proceedings of the thirteenth

ACM International Conference on Information and Knowledge Management (pp. 236-237).

[64] He, B., & Ounis, I. (2007). Combining fields for query expansion and adaptive query expansion. Information Processing & Management, 43(5), 1294-1307.

[65] Cong, H., Chen, W. N., & Yu, W. J. (2021). A two-stage information system retrieval system based on interactive multimodal genetic algorithm for query weight optimization. Complex & Intelligent Systems, 7(5), 2765-2781.

[66] Balasubramanian, N., Kumaran, G., & Carvalho, V. R. (2010, July). Exploring reductions for long web queries. In Proceedings of the 33th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. (571-578).

[67] Jones, R., & Fain, D. C. (2003, July). Query word deletion prediction. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 435-436).

[68] Chang, Y. C., & Chen, S. M. (2006). A new query reweighting method for document retrieval based on genetic algorithms. IEEE Transactions on Evolutionary Computation, 10(5), 617-622.

[69] Bindal, A. K., & Sanyal, S. (2012). Query optimization in context of pseudo relevant documents. In 3rd Italian Information Retrieval (IIR) Workshop (No. CONF).

[70] Wang, J., Huang, J. Z., Guo, J., & Lan, Y. (2017). Query ranking model for search engine query recommendation. International Journal of Machine Learning and Cybernetics, 8(3), 1019-1038.

[71] Huang, Z., Cautis, B., Cheng, R., & Zheng, Y. (2016, October). Kb-enabled query recommendation for long-tail queries. In Proceedings of the 25th ACM International Conference on Information and Knowledge Management (pp. 2107-2112).

[72] Baeza-Yates, R. (2010, September). Query intent prediction and recommendation. In Proceedings of the fourth ACM Conference on Recommender Systems (pp. 5-6).

[73] Ozertem, U., Chapelle, O., Donmez, P., & Velipasaoglu, E. (2012, August). Learning to suggest : a machine learning framework for ranking query suggestions. In Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 25-34).