A Quality- and Security-improved Web Search using Local Agents

Mario Kubek, Herwig Unger and Thipkunya Loauschasai

***Abstract*—**Searching the World Wide Web (WWW) is a tedious task since the search engine's queries may only contain keywords to describe the contents of interest. Any desired, higher degree of personalisation requires user accounts at the provider's side that involves a transfer of maybe confidential and private information. Moreover, the search engines do not regard the valuable and existing knowledge on the user's machines to generate search results. The herein presented locally working algorithms for search word extraction and query expansion support the development of local search agents to overcome the described drawbacks.

***Keywords*—** web information retrieval, search word extraction, source topic detection, co-occurrence analysis, local agent

I. Introduction

IN the last few years, information access in the World Wide Web (WWW) became more and more dominated by big search engines like Google, since there is no connection between any information and its place where it is stored in the WWW. Therefore, those programs are needed to find the respective correspondence between a few search words given by a user, matching contents and their location. In average, two search words are given [1] in order to determine the respective contents, what normally results in a plenty of results that are ranked by algorithms like HITS [2], PageRank [3] or PageReputation [4]. It has been figured out in several publications like [5] and [6] that query expansion may reduce the amount of search results significantly by a better description of the search subject. Therefore, most search engines already provide such functionality to their users based on the keywords entered by many users along with the initial query [7]. A personalised user account at one of the big search engine providers like Google enables a significant refinement of the search. By doing so, the search engine can store a bigger set of information of the respective user and therefore learns about his or her special behavior and interests. Of course, this approach has another aspect: the user discloses personal details related to private life that may be extracted by data-mining methods of the providers.

The aim of this contribution is to introduce algorithms that are designed to run on a local agent for an improved web search which avoids the above mentioned disadvantages (see Fig. 1) and increases the security of personal user information. Therefore, the unrivaled huge and well-indexed databases as well as access mechanisms of the big search engines are combined with a local pre-processing agent.

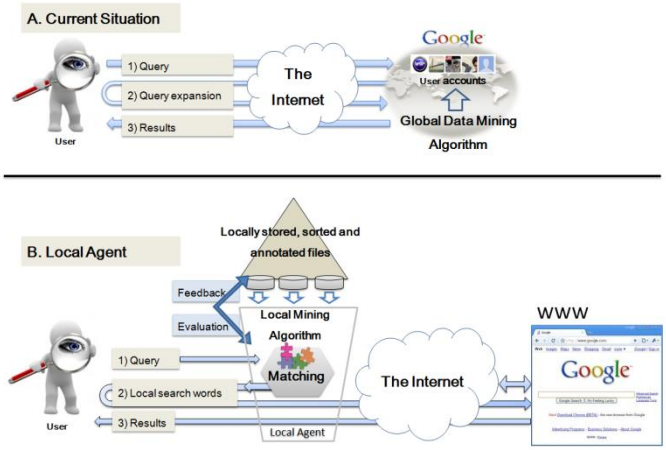

Fig. 1: Cooperation of the local agent and search engines

This agent has access to the local (and maybe confidential) files of the user and may even establish a fine-granular user profile. Since these data are kept local, there is no danger for privacy and security of them. After processing the current search words of the user and/or the knowledge the agent obtains from these local files and previous searches, a keyword suggestion is presented to the user, which can be used or not to be sent as a query to the (remote) search engine, which then returns its results in the known manner.

The main and so far not properly solved problem is how to obtain such keywords from local data, only. The classic methods employing word frequency analysis like TF-IDF [8] and difference analysis [9] do not work satisfying, since usually too less (similar) documents are available. Also, methods to generate an ontology or taxonomy need larger amounts of data to derive reliable knowledge and thus are also not applicable properly.

Mario Kubek and Herwig Unger are with the Faculty of Mathematics and Computer Science, FernUniversity in Hagen, Germany, (phone: +49 2331 987 1153; fax: +49 2331 987 375; e-mail: mario.kubek@fernuni-hagen.de and herwig.unger@fernuni-hagen.de).

Thipkunya Loauschasai is with the Faculty of Information and Communication Technology of the Mahidol University, Nakhonpathom, Thailand, e-mail: U5288103@student.mahidol.ac.th.

Therefore, this paper describes previously developed solutions for automatic and local keyword extraction and query expansion by the authors and introduces a new method to determine characteristic terms from texts by analysing their directed co-occurrence graphs using an extended version of the HITS algorithm that can run on the local agent. It is shown, that the obtained terms are well suited to be used for the automatic retrieval of semantically similar and related documents from large corpora like the WWW through automatic and local query formulation. As local documents often represent user interests well, these search words also provide a proper basis for user specific query expansion to narrow down the large amount of search results web search engines usually return. In order to improve search term extraction, the direction of the relation of co-occurring terms is determined to model the generally asymmetric real-life relationships between the concepts they represent and to also indicate a recommendation of one term for another one by this means. Among the approaches discussed to obtain these directed term relations, one novel solution statistically determines the impact two co-occurring terms have on each other by computing the influence that the context of one term, that is the set of the most significant terms it co-occurs with, has on the context of the respective other term involved in that relation. Two additional advantages of the proposed methods and solutions are that they do not rely on preferably large document collections or on third-party datasets like reference corpora and that they can be applied on single texts. Derived application scenarios like following topics across multiple documents and structural reranking of web search results are elaborated on as well.

The remaining paper is structured as follows: the next section explains the methodology used and presents a new and locally working graph-based solution for extracting keywords and source topics based on an extended version of the HITS algorithm. In this section, it is also outlined, how to calculate directed term relations from texts by applying co-occurrence analysis in order to obtain directed co-occurrence graphs for this purpose. Section three focuses on the conducted experiments with this algorithm. It is also shown that the extracted terms can be used to find similar and related documents in the WWW. Further use cases, partly based on existing applications, for the described local search agent are provided in section four to improve the search in the web. Section five provides a look at future options to enable a fully decentralized and collaborative search using such local agents.

II. Methodology and Algorithms

The local und unsupervised extraction of keywords and keyphrases, that are to be used as search words, should return state-of-the-art quality results. In order to achieve this goal, the authors propose graph-based methods to construct semantic networks which contain the relations between the terms in documents to be analysed and which are then used to detect these keywords by various means. Despite other well-known and well-studied classic approaches for this task like TF-IDF [8] and difference analysis [9] these graph-based methods do not rely on preferably large reference corpora or document sets to determine characteristic terms. Instead, while taking into account not only the strength but also the direction the semantic relations of terms, the graph-based methods do not need such corpora. Yet, they return state-of-art results, can run locally on the user's machines and provide several solutions to determine keywords within the scope of different application scenarios, namely to determine keywords for finding similar documents, for source topic detection to realise topic tracking and for expanding user queries. These scenarios, as use cases for the local agent, will be described in detail in section four.

The mentioned term relations can be obtained using statistical co-occurrence analysis and represented as a graph of connected nodes with the terms as the vertices and their interconnecting relationships as weighted edges. Well-known measures to gain co-occurrence significance values on sentence level are for instance the mutual information measure [10], the Dice [11] and Jaccard [12] coefficients, the Poisson collocation measure [13] and the log-likelihood ratio [14]. The resulting co-occurrence graphs are generally undirected which is suitable for the flat visualisation of term relations as shown in Fig. 2 and for applications like query expansion via spreading activation techniques [6].

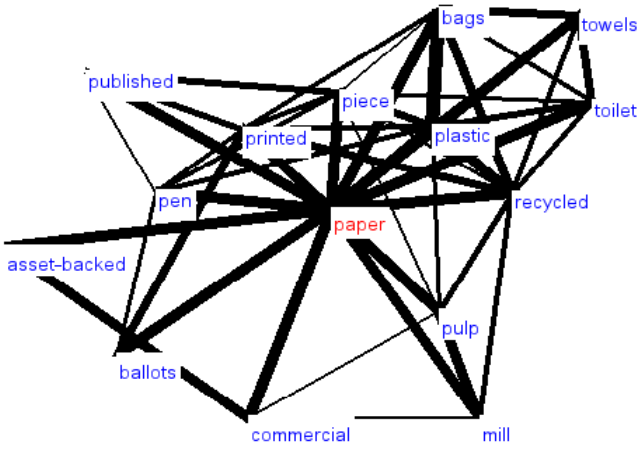

Fig. 2 A co-occurrence graph for the word ”paper” (Source: http://corpora.informatik.uni-leipzig.de/)

However, real-life associations are mostly directed, e.g. a BMW is a German car but not every German car is a BMW. The association of BMW with German car is therefore much stronger than the association of German car with BMW. Therefore, it actually makes sense to deal with directed term relations to improve keyword extraction and to realise the application scenarios mentioned above. Thus, in the next two subsections, algorithms will be presented to generate directed co-occurrence graphs and to determine keywords and keyphrases by analysing them.

A. Generating directed Co-occurrence Graphs

In this subsection, various approaches to determine directed term relations will be described, whereby two of them solely rely on statistical computations that can be carried out locally, effectively and efficiently without consulting external databases.

Determining directed term relations using conditional relative frequencies

To measure the significance of the directed relation of term A with term B, which can also be regarded as the strength of the association of term A with term B, the following formula of the conditional relative frequency can be used, whereby $|A \cap B|$ is the number of times term A and B co-occurred in the text on sentence level and $|A|$ is the number of sentences term A occurred in:

$$Assn(A \rightarrow B) = \frac{|A \cap B|}{|A|} \quad (1)$$

Often, this significance differs greatly in regards of the two directions of the relations when the difference of the involved term frequencies is high. The association of a less frequently occurring term A with a frequently occurring term B could reach a value of 1.0 when A always co-occurs with B, however B's association with A could be almost 0. This means, that B's occurrence with term A is insignificant in the analysed text. That is why it is sensible to only take into account the direction of the dominant association (the one with the higher value) to generate a directed co-occurrence graph for the further considerations. However, the dominant association should be additionally weighted. In the example above, term A's association with B is 1.0. If another term C, which more frequently appears in the text than A, also co-occurs with term B each time it appears, then its association value with B would be 1.0, too. Yet, this co-occurrence is more significant than the co-occurrence of A with B. An additional weight that influences the association value and considers this fact could be determined by

- the (normalised) number of sentences, in which both terms co-occur or
- the (normalised) frequency of the term A. The normalisation basis could be the maximum number of sentences, which any term of the text has occurred in.

The association Assn of term A with term B can then be calculated using the second approach by:

$$Assn(A \rightarrow B) = \frac{|A \cap B|}{|A|} \cdot \frac{|A|}{|n_{max}|} \quad (2)$$

Hereby, $|n_{max}|$ is the maximum number of sentences, any term has occurred it. A thus obtained relation of term A with term B with a high association strength can be interpreted as a recommendation of A for B. Relations gained by this means are more specific than undirected relations between terms because of their direction. They resemble a hyperlink on a website to another one. In this case however, it has not been manually and explicitly set and it carries an additional weight that indicates the strength of the term association. The set of all such relations obtained from a text represents a directed co-occurrence graph. The next subsection provides a more sophisticated approach to gain asymmetric relations between terms.

Determining directed term relations using term context dependencies

The context of a term is the set of terms it co-occurs with significantly. Two terms can be regarded as similar to each other when their contexts overlap to a great extent using the bag-of-words model. This significance can be determined using the cosine similarity measure, that takes the term contexts as term vectors and the angle between them is calculated. The lower the angle, the more similar the contexts and terms in question are. In this case, it is likely that these two terms also co-occurred in the analysed text.

The authors now propose a simple yet powerful language model to detect not only such similarities, but also their asymmetric dependencies. A similar model has been applied in literature [15] to automatically generate asymmetric links between documents in order to rerank documents using an initial retrieval method. For this purpose, asymmetric document-generation probabilities are determined based on the term vectors of two documents. To calculate the strength of a relation between a term A and another term B using their contexts the following steps must be carried out:

1. Determine the contexts C_a and C_b (e.g. up to 15 entries) of both terms A and B using co-occurrence analysis.
2. Calculate the overlap of both contexts $|C_a \cap C_b|$ which is the number of terms that appear in both contexts.
3. Calculate the association of term A with term B using the following formula, whereby $|C_a|$ is the number of terms in the context C_a:

$$Assn(A \rightarrow B) = \frac{|C_a \cap C_b|}{|C_a|} \quad (3)$$

Note that the steps 2 and 3 must be repeated to gain the strength of the reverse relation of term B with term A. An additional weighting of the gained relations as shown in formula 2 is not urgently needed, because the maximum number of terms in the contexts is fixed and therefore cannot vary greatly. This solution is also similar to the first approach described. However, as the steps of co-occurrence calculation and comparison of the contexts are involved and only the most important co-occurrences of both terms will appear in the contexts, the quality of the resulting relations will be higher, yet the computations will take more time. Also, it is sensible to only take into account the dominant relation between two terms to construct a directed co-occurrence graph.

Enhancing the detection of directed term relations

Detecting the relationship of term pairs by statistical means is very effective. However, there are further approaches to enhance this detection and to correct wrongly detected relations. One possible way is to consult the manually created

semantic network WordNet [16], a large lexical database that contains semantic relationships for the English language and covers relations like polysemy, synonymy, antonymy, hypernymy and hyponymy (i.e. more general and more specific concepts), and part-of-relationships. If two co-occurring terms are e.g. synonyms then it is sensible to merge their respective vertices of the co-occurrence graph, or to add this relationship information to their interconnecting edges as an additional feature or to at least draw an undirected edge with a high weight between the vertices of the terms. If one term A is a hyponym of another co-occurring term B or it is in a part-of-relationship with that term B, then a directed and accordingly annotated edge should be drawn from the vertex of term A to term B whereby the weight is depending on the distance of these terms in the WordNet graph [17]. Another way to determine term relations in texts to build directed term graphs is the usage of dependency parsers [18]. After an applied part-of-speech tagging they identify syntactic relation-ships among words in the text and generate dependency trees of them for each sentence. Detecting term relations using lexico-syntactic patterns is another well-known approach [19] for this task. Hereby, interesting patterns of parts of speech and/or wordforms in a specified order are defined and searched for in the texts. This way, part-of- and is-a-relationships can be easily detected. A pattern like "[NN] like [NN] and [NN]" can be used to uncover hyponyms and hypernyms and thus determine the direction of the relation between the terms referred to.

B. Keyword Extraction using Extended HITS

In [20], the authors proposed an extended version of the PageRank algorithm [3] for graph-based keyword extraction applied on co-occurrence graphs that also takes into account the strength of the semantic term relations. A vertex (term) has a high relative importance (is an authority) in such a graph when it is often linked to by other important vertices and the interconnecting links have a high weight. Similar approaches to determine keywords of documents have been presented in [20] and [21] that rely on semantic graphs gained from synsets from WordNet. Furthermore, the authors have presented a solution for automatic query expansion based on a spreading activation technique applied to co-occurrence graphs in a browser-based environment [6]. It was the first approach to apply the user's local knowledge in texts to enhance the search in the web. It was shown, that the precision of the returned results increased drastically when queries containing related terms found by this technique were used. In this paper, the authors extend the approach to use graph centrality algorithms for keyword extraction by introducing an extended version of the HITS applied on directed co-occurrence graphs to not only determine keywords, but to also determine terms in texts that can be referred to as source topics. These terms strongly influence the main topics in texts, yet are not necessarily important keywords themselves. They are helpful when it comes to applications like following topics by repeatedly analysing documents that cover them primarily.

The HITS algorithm [2], which was initially designed to evaluate the relative importance of nodes in web graphs (which are directed), returns two lists of nodes: authorities and hubs. Authorities are nodes that are often linked to by many other nodes. They can be determined using the PageRank algorithm, too. Hubs are nodes that link to many other nodes and therefore topically influence the nodes they link to. Nodes are assigned both a score for their authority and their hub value. For undirected graphs the authority and the hub score of a node will be the same, which is naturally not the case for the web graph. Regarding the analysis of directed co-occurrence graphs with HITS, the authorities can be seen as the characteristic terms of the analysed text, whereas the hubs represent its source topics. Using the solutions to generate directed co-occurrence graphs it is now possible to introduce a new method to analyse them in order to find keywords and source topics in the texts they represent. For this purpose, the application of the HITS algorithm on these graphs is sensible due to its working method that has been outlined above. The list of hub nodes in these graphs returned by HITS contain the terms that can be regarded as the source topics of the analysed texts as they represent their inherent concepts. Their hub value indicates their influence on the most important topics and terms that can be found in the list of authorities. For the calculation of these lists using HITS, it is also sensible to also include the strength of the associations between the terms. These values should also influence the calculation of the authority and hub values. The idea behind this approach is that a random walker is likely to follow links in co-occurrence graphs that lead to terms that can be easily associated with the current term he is visiting. Nodes that represent terms that are linked with a low association value however should not be visited very often. This also means that nodes that reside on paths with links of high association values should be ranked highly as they can be reached easily and are visited frequently. Therefore, the formulas for the update rules of the HITS algorithm should be modified to include the association values *Assn* from the previous section. The authority value of node x can then be determined using formula 4:

$$a(x) = \sum_{v \to x} h(v) \cdot Assn(v \to x) \qquad (4)$$

The hub value of node x can be calculated using formula 5:

$$h(x) = \sum_{x \to w} a(w) \cdot Assn(x \to w) \qquad (5)$$

The following steps are necessary to obtain two lists for the authorities and hubs based on these update rules:

1. Remove stopwords and apply stemming algorithm on all terms in the text. (Optional)
2. Determine its directed co-occurrence graph G based on one of the solutions presented in the previous subsection.
3. Determine the authority value a(x) and the hub value h(x) iteratively for all nodes x in G using the formulas 4 and 5 until convergence is reached (the calculated values do not change significantly in two consecutive iterations) or a fixed number of iterations has been executed.
4. Return all nodes in descending order by their authority and hub values with their representing terms and their authority and hub values.

The effectiveness of this method will now be illustrated by experiments.

III. Experiments and Experiences

In this section, excerpts of ranked authority and hub lists of example documents will be given. It is also shown, that the proper selection of terms from these term lists can be used to find similar and related documents in the World Wide Web. Similar experiments have been conducted in [22] using an extended version of the PageRank algorithm applied on co-occurrence graphs in order to find keywords in the texts they represent. Therein, it was shown among further results for 200 documents from the English Wikipedia that the term lists with 10 terms obtained from the PageRank method and the TF-IDF measure have an average overlap of even 70%. These results clearly demonstrate that graph-based term ranking is a valid approach for high-quality keyword extraction. This finding will now be underlined using the results obtained from the extended version of the HITS algorithm introduced in the previous section.

A. Detection of Authorities and Hubs

The following tables present for three documents of the English Wikipedia the lists of the 10 terms with the highest authority and hub values. To conduct these experiments, the following parameters have been used:

- removal of stopwords
- restriction to nouns and adjectives
- baseform reduction
- activated phrase detection

The generation of the directed co-occurrence graphs has been carried out using the approach described in section two.

TABLE I
TERMS AND PHRASES WITH HIGH AUTHORITY AND HUB VALUES OF THE WIKIPEDIA-ARTICLE "LOVE":

Term	***Authority value***	***Term/Phrase***	***Hub value***
love	0.54	friendship	0.21
human	0.30	intimacy	0.18
god	0.29	passion	0.16
attachment	0.26	religion	0.14
word	0.21	attraction	0.14
form	0.21	platonic love	0.13
life	0.20	interpersonal love	0.13
feel	0.18	heart	0.13
people	0.17	family	0.13
buddhism	0.14	relationship	0.12

The examples show that the extended HITS algorithm can determine the most characteristic terms (authorities) and source topics (hubs) in texts by analysing their directed co-occurrence graphs. Especially the hub list for each text provides valuable information to find suitable terms that can be used as search words in queries when background information is needed to a specific topic. Moreover, the terms found in the authority lists can also be used as suitable search words in order to find similar documents. This will be shown in the next subsection. Another finding was, that the quality of the authority and hub lists improved when analysing clusters of semantically similar documents instead of single texts. The reason for this is that smaller texts like newspaper articles often address only specific subtopics of a main topic. Therefore, document-specific terms and topics gain a high importance for such a document, but would not be of great importance when this document would be analysed together with other documents of the main topic. Moreover, when analysing clusters of documents with a given main topic, the results in the authority and hub lists will be more meaningful because they can be statistically validated. However, that does not mean, that external corpora for the ranking are necessary. A small set of topically related documents is sufficient to increase the keyword quality drastically. As an example, the hub list of the document [23] on the main topic "Paypal" contained the very general terms "plan" and "data" as well as very document-specific terms like "brick-and-mortar". In contrast, the following authority and hub lists were calculated on a basis of five freely available documents on "Paypal" in the WWW. It can be easily seen that the quality of the keywords improved notably due to the larger text basis.

TABLE II
TERMS AND PHRASES WITH HIGH AUTHORITY AND HUB VALUES OF THE WIKIPEDIA-ARTICLE "EARTHQUAKE":

Term	***Authority value***	***Term/Phrase***	***Hub value***
earthquake	0.48	movement	0.18
earth	0.30	plate	0.16
fault	0.27	boundary	0.15
area	0.23	damage	0.15
boundary	0.18	zone	0.15
plate	0.16	landslide	0.14
structure	0.16	seismic activity	0.14
rupture	0.15	wave	0.13
aftershock	0.15	ground rupture	0.13
tsunami	0.14	propagation	0.12

TABLE III
TERMS AND PHRASES WITH HIGH AUTHORITY AND HUB VALUES OF THE WIKIPEDIA-ARTICLE "ANDROID" (MOBILE OPERATING SYSTEM):

Term/Phrase	***Authority value***	***Term/Phrase***	***Hub value***
Android	0.32	source code	0.19
Google	0.31	development	0.18
application	0.27	October	0.16
version	0.24	project	0.15
open source	0.23	platform	0.14
Linux	0.22	handset	0.14
system	0.22	Alliance	0.13
December	0.21	Android Inc.	0.13
software	0.20	Java	0.13
Play Store	0.19	mobile	0.12

Beside the quality improvements, the results also show that an additional clustering of the terms in the lists based on their topical relations is a sensible option to enhance the suggestion of search words. This option will be addressed in future contributions.

TABLE IV
TERMS AND PHRASES WITH HIGH AUTHORITY AND HUB VALUES ON THE TOPIC "PAYPAL"

Important Authorities	Important Hubs
Paypal	fee
payment	account
credit card	bank
Ebay	Paypal account
money	transaction
service	customer
seller	service
fund	business
account	company
transaction	user

B. Finding Similar and Related Documents in the WWW

The suitability for the authorities and hubs as search words will now be shown. For this purpose and as an example, the five most important authorities and the five most important hubs of the Wikipedia article "Love" have been combined as search queries and sent to Google. Empiric experiments have shown that at most five terms and phrases should be used for this purpose. A larger number would limit the search results too much, while too few terms would return too many and possibly irrelevant results. The results of this test can be seen in Fig. 3 and Fig. 4.

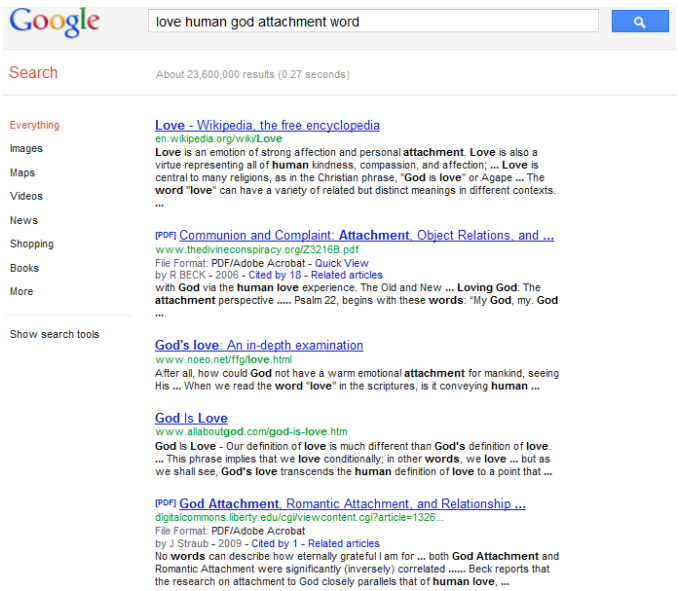

Fig. 3 Search results for the auth. of the Wikipedia article "Love"

The search query containing the hubs of this article will lead to these results:

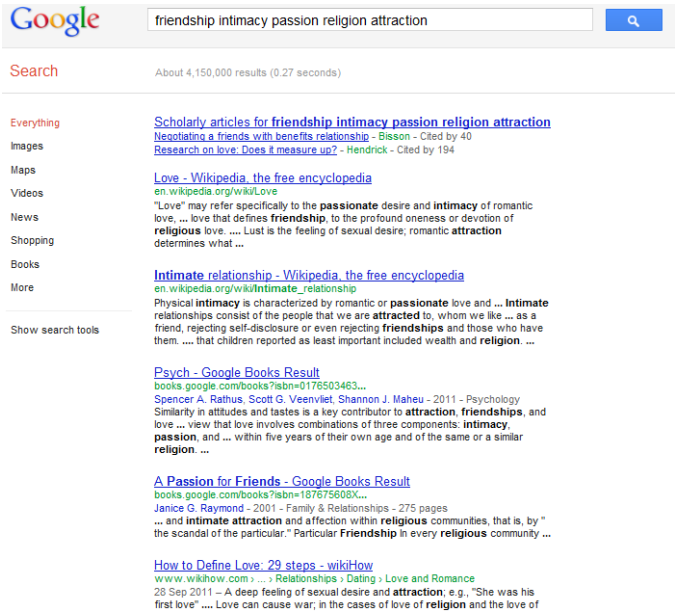

Fig. 4 Search results for the hubs of the Wikipedia article "Love"

The search results clearly show that they primarily deal with either the authorities or the hubs. More experiments confirm this correlation. Using the authorities as queries to Google it is possible to find similar documents to the analysed one in the Web. Usually, the analysed document itself is found among the first search results, which is not surprising though. However, it shows that this approach could be a new way to detect plagiarised documents. It is also interesting to emphasise the topic drift in the results when the hubs have been used as queries. This observation indicates that the hubs of documents can be used as a means to follow topics across several related documents with the help of Google. Hereby, it is desirable that the hubs of the analysed documents are the authorities of the found documents to obtain a chain of documents that are indeed topically depending. This possibility will be elaborated on in more detail in the next section of this paper that discusses several use cases and application scenarios for the introduced local agent.

IV. TOOL SUPPORT

In this section, use cases for the local agent based on the herein discussed and previously published algorithms and applications by the authors will be given in order to obtain quality-improved and personalised web search results while providing security of personal user information at the same time. This local agent then acts as a bridge between the local machine and the web search engines which enables users to e.g. search for local documents with specific search words and to use the local knowledge to search the web.

A. Local Query Expansion

In [6], the authors have introduced the Firefox browser extension "FXResearcher", to support the user in searching for documents on the local computer and on web presences. Its new approach is the usage of client-sided query expansion based on the analysis of co-occurrence graphs of local text

documents using a spreading activation algorithm in order to find more proper documents in the web. To find these matching documents "FXResearcher" integrates a full text indexer that incrementally indexes folders on the local computer that have been explicitly specified by the user.

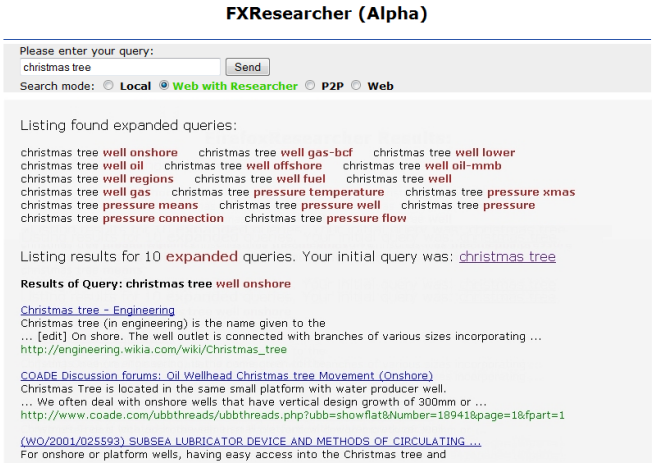

Fig. 5 Screenshot of "FXResearcher" with found expanded queries and returned web results

Documents in this index can then be interactively selected for query expansion as a relevance feedback when they contain terms of a search query. This approach is promising, because a query will likely not be expanded with improper terms as it still occurs when query expansion will be first performed by the requested web search engine. The latter solution might completely fail when for instance an expert searches for documents with a quite specific term that does not occur in many documents. The usage of suggested improper expansion terms could return only a few and inadequate results. The consideration of the local knowledge for query expansion on the other hand can provide the expert with proper expansion terms on his area of expertise and therefore return more expected search results. With this approach, even queries consisting of homonyms can be properly expanded. Furthermore, it was shown that the quality of web search results increased drastically when this approach of local query expansion is used that regards the local knowledge for the web search. Therefore, it is a suitable use case for the local agent, too.

B. Searching for Similar and Related Contents in the Web

Based on the herein presented algorithms for determining search words in local text documents, an interesting use case for the local agent is to find topically similar and related documents in the WWW. A first implementation of this idea is the interactive Firefox extension "FireMatcher" [24]. Its aim is to locally analyse text documents the user provides and to send their characteristic terms as queries to web search engines in order to find similar documents. The returned search words and web search results are generally of good quality. The implementation, however, does not use graph-based algorithms that provide state-of-art quality results in order to fulfill these tasks. Also, source topic detection, as

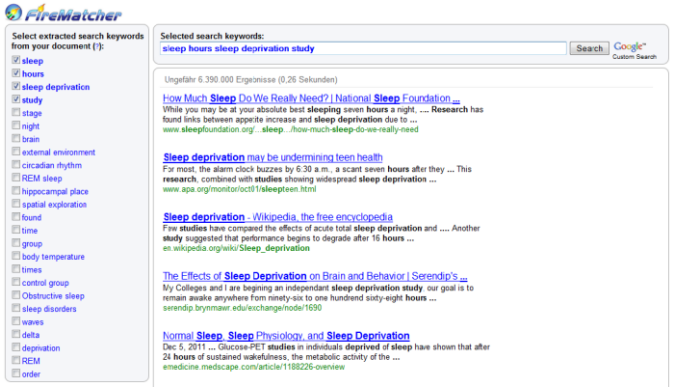

Fig. 6 Screenshot of "FireMatcher" with determined search words and web results returned by Google

described in section two, is not possible in "FireMatcher". However, as shown in section three, it is sensible to determine source topics in texts (additionally to the keywords) to find related, but not necessarily similar, contents in the WWW with their help. Moreover, the detected source topics provide the basis for the following two use cases to be integrated into the local agent.

C. Topic Tracking Based on Document Dependencies

Another interesting application using detected source topics can be seen in the automatic linking of related documents found in large corpora like the WWW. If a document A primarily deals with the source topics of another document B, then a link from A to B can be set. This way, the herein described approach to obtain directed term associations is modified to gain the same effect on document level, namely to calculate recommendations for specific documents. The local agent could download and analyse web search results automatically in the background and build up an internal index of these document relations. Thereby, the local agent incrementally learns new document relationships. New search results can then be provided with links to similar but also to related documents that primarily deal with their source topics in order to give users access to background information on a topic of interest and to also follow topics across multiple documents. The idea behind this approach is that especially the determined source topics can lead users to documents that cover important aspects of their analysed and presented search results. This goes beyond a simple search for similar documents as it offers a new way to search for related documents. This functionality can be seen as a useful addition to Google Scholar (http://scholar.google.com/), which offers users the possibility to search for similar scientific articles.

D. Local Reranking of Web Results

These automatically determined links between web search results can also be very useful in terms of positively influencing the ranking of search results, because these links represent semantic relations between documents that have been verified in contrast to manually set links e.g. on websites, which can be automatically evaluated regarding their validity using the approach for source topic detection, too. To realise this function, the web search results must be downloaded by

the local agent and analysed regarding their semantic dependencies as outlined in this section. Based on these relationships, the web search results could be reordered in such a manner, that topical clusters become visible. Also, by comparing the newly found documents with the lists of keyword and source topics of locally existing documents, it is possible to rerank them based on their similarity with the local knowledge. As this function will take a possibly large amount of time, its use is not appropriate when a timely response is needed. However, it is sensible, when an in depth analysis of a topic required and real-time demands play a secondary role.

V. Conclusion and Outlook

A local agent and three new algorithms for local query expansion and keyword extraction have been introduced to improve the search in the Internet. This approach is just a first step towards a new content management in the Internet, avoiding an exhausting use of central resources. In the next steps, it is intended to allow a communication of those agents to support a real collaborative search. Finally, a broader exchange of information among a group of agents may build a new information space allowing a fully decentralised coordination of the data exchange among providers and consumers.

References

[1] R. Agrawal et al., "Enrichment and Reductionism: Two Approaches for Web Query Classification", In Bao-Liang Lu, Liqing Zhang, and James T. Kwok, editors, ICONIP (3), volume 7064 of Lecture Notes in Computer Science, pp. 148–157, Springer, 2011.

[2] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment", In Proc. of ACM-SIAM Symp. Discrete Algorithms, San Francisco, California, pp. 668–677, January 1998.

[3] L. Page, S. Brin, R. Motwani, and T. Winograd. "The PageRank Citation Ranking: Bringing Order to the Web", Technical Report, Stanford Digital Library Technologies Project, 1998.

[4] D. Rafiei and A. O. Mendelzon, "What is this page known for? computing web page reputations", In Proc. of the Ninth International World Wide Web Conference, Amsterdam, Netherland, pp. 823–835, May 2000.

[5] J. Xu and W. B. Croft, "Query expansion using local and global document analysis", In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '96, pp. 4–11, New York, USA, 1996.

[6] M. Kubek and H.F. Witschel, "Searching the Web by Using the Knowledge in Local Text Documents", In Proceedings of Mallorca Workshop 2010 Autonomous Systems, Shaker Verlag, Aachen, 2010.

[7] Website of Google Autocomplete, Web Search Help, http://support.google.com/websearch/bin/answer.py?hl=en&answer=106230

[8] G. Salton, A. Wong, and C.S. Yang, "A vector space model for automatic indexing", Commun. ACM, 18:613–620, November 1975.

[9] G. Heyer, U. Quasthoff, and T. Wittig, *Text Mining: Wissensrohstoff Text: Konzepte, Algorithmen, Ergebnisse*, W3L-Verlag, 2006.

[10] M. Büchler, "Flexibles Berechnen von Kookkurrenzen auf strukturierten und unstrukturierten Daten", Master's thesis, University of Leipzig, July 2006.

[11] L. R. Dice, "Measures of the Amount of Ecologic Association Between Species", Ecology, 26(3):297–302, July 1945.

[12] P. Jaccard, "Étude Comparative de la Distribution Florale dans une Portion des Alpes et des Jura", Bulletin de la Société Vaudoise des Sciences Naturelles, 37:547–579, 1901.

[13] U. Quasthoff and C. Wolff, "The Poisson Collocation Measure and its Applications", In Second International Workshop on Computational Approaches to Collocations, IEEE, 2002.

[14] T. Dunning, "Accurate methods for the statistics of surprise and coincidence", Comput.Linguist., 19:61–74, March 1993.

[15] O. Kurland and L. Lee, "PageRank without hyperlinks: Structural re-ranking using links induced by language models", ACM Transactions on Information Systems (TOIS), 28(4), 2010.

[16] C. Fellbaum, "WordNet and wordnets", In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670, 2005.

[17] T. Hughes and D. Ramage, "Lexical semantic relatedness with random graph walks", In EMNLP-CoNLL, pp. 581–589, 2007.

[18] R. McDonald et al., "Non-projective dependency parsing using spanning tree algorithms", In Proc. of the Joint Conf. on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP), 2005.

[19] E. Riloff and R. Jones, "Learning dictionaries for information extraction by multi-level bootstrapping", In Proceedings of the Sixteenth National Conference on Artificial Intelligence, pp. 474-479, Orlando, FL, 1999.

[20] J. Wang, J. Liu, and C. Wang, "Keyword Extraction Based on PageRank", In: PAKDD, pp. 857-864, 2007.

[21] R. Mihalcea, P. Tarau, and E. Figa, "PageRank on Semantic Networks, with application to Word Sense Disambiguation" In Proceedings of The 20st International Conference on Computational Linguistics, 2004.

[22] M. Kubek and H. Unger, "Search Word Extraction Using Extended PageRank Calculations", In Autonomous Systems: Developments and Trends, volume 391 of Studies in Computational Intelligence, pp. 325–337, Springer Berlin / Heidelberg, 2011.

[23] M. White, "Watch Out Visa and MasterCard: PayPal Is Coming to a Brick-and-Mortar Store Near You", Time Moneyland, http://moneyland.time.com/2012/02/15/paypal-wants-to-be-a-credit-card-alternative-in-stores/, 2012.

[24] Website of "FireMatcher", http://www.firematcher.com