Global Journal of Computer Science and Technology: C
Software & Data Engineering
Volume 14 Issue 2 Version 1.0 Year 2014
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Information Retrieval based on Content and Location Ontology for Search Engine (CLOSE)

By Niranjan Kumar & S. G. Raghavendra Prasad

Rashtreeya Vidyalaya College of Engineering, India

Abstract- This paper mainly focuses on the personalization of the search engine based on data mining technique, such that user preferences are taken into consideration. Clickthrough data is applied on the user profile to mine the user preferences in order to extract the features to know in which users are really interested. The basic idea behind the concept is to construct the content and location ontology's, where content represent the previous search records of the user and location refer to current location of user. SpyNB is the approach used to mining the user preferences from the Clickthrough data. The ranked support vector machine (RVSM) is performed on the searched results in order to display results according to user preferences by considering Clickthrough data.

Keywords: SpyNB, personalization, ontology, RSVM, non-geographic search, geographic search, search engine optimization (SEO), personalized information retrieval (PIR).

GJCST-C Classification: H.3.3

Strictly as per the compliance and regulations of:

Information Retrieval based on Content and Location Ontology for Search Engine (CLOSE)

Niranjan Kumar [α] & S. G. Raghavendra Prasad [σ]

Abstract- This paper mainly focuses on the personalization of the search engine based on data mining technique, such that user preferences are taken into consideration. Clickthrough data is applied on the user profile to mine the user preferences in order to extract the features to know in which users are really interested. The basic idea behind the concept is to construct the content and location ontology's, where content represent the previous search records of the user and location refer to current location of user. SpyNB is the approach used to mining the user preferences from the Clickthrough data. The ranked support vector machine (RVSM) is performed on the searched results in order to display results according to user preferences by considering Clickthrough data.

Keywords: SpyNB, personalization, ontology, RSVM, non-geographic search, geographic search, search engine optimization (SEO), personalized information retrieval (PIR).

I. Introduction

In the modern information retrieval system, the results that are found should be more accurate to query submitted by the user, and also efficiency should be considered.

In order to solve the problems that are faced by the current search engine technology such as retrieving results that are irrelevant to the search query, the order in which they are displayed should be considered. According to Hele-Mai Haav [1] to solve problem of information retrieval in current information retrieval systems it should be improved by intelligence to manage the effective retrieval, filtering and presenting relevant information. So two main information retrieval models are classified as, keyword based information retrieval model and concept based information retrieval model. The indexing terms and Boolean logical queries are used in keyword based model, where indexing may be automatic or manual, when Boolean query are taken into consideration the frequency of occurrence is taken into account.

Context-aware system [2], depending on the user's relevancy the information/services is provided. For instance consider the keyword apple, it can mean as a fruit or it can mean as a mobile and laptops by Apple Company. When the query is submitted by two different users, irrespective of their interest same results are displayed for both users, if one user is interested only on apple accessories, for him both relevant and irrelevant information are displayed in random order. The information for what the user is looking may be in same document else somewhere in the overall document. The current system performs word to word matching of the search query.

Another instance in search engine is searching for places based on current location of the user. For example, if the user current location is Jaynagar and user trying to search restaurant near by current location, the search engine must show the restaurant which are near to the current location of the users and rest of the restaurant location other than jaynagar should be given next preference. The detailed discussion related to geographic and non-geographic search is given in proposed system section.

The main aspects that should be considered in information retrieval system is to reduce the complexity involved in query execution [3] such that performing lexical analysis, stemming process on the user query and construction of index terms. This paper focuses on search engine optimization (SEO) by reducing the complexity in the user query execution.

The rest of the paper is organized as: - In section II literature survey is carried out by surveying previous paper present, such that what are the technologies currently used to optimize the search engine. In section III technique to reduce the complexity for optimization of search query are studied. In section IV detailed view of implementation. In section V experimental evaluation and in IV Conclusion and future enhancements are discussed.

II. Literature Survey

M. Rami Ghoran [4] studied that for every query that is submitted by the user he will get the relevant and irrelevant information for that query. So they classify the personalized information retrieval (PIR) system into three scopes: Individualized system, community-based system and aggregate-level system.

When individualized system is considered the system adaptive [5][6] decision are taken such that, the user interest and preferences are taken into account while

Author α σ : RVCE, Bangalore. e-mails: niranjankumar213@gmail.com, raghavendrap@rvce.edu.in

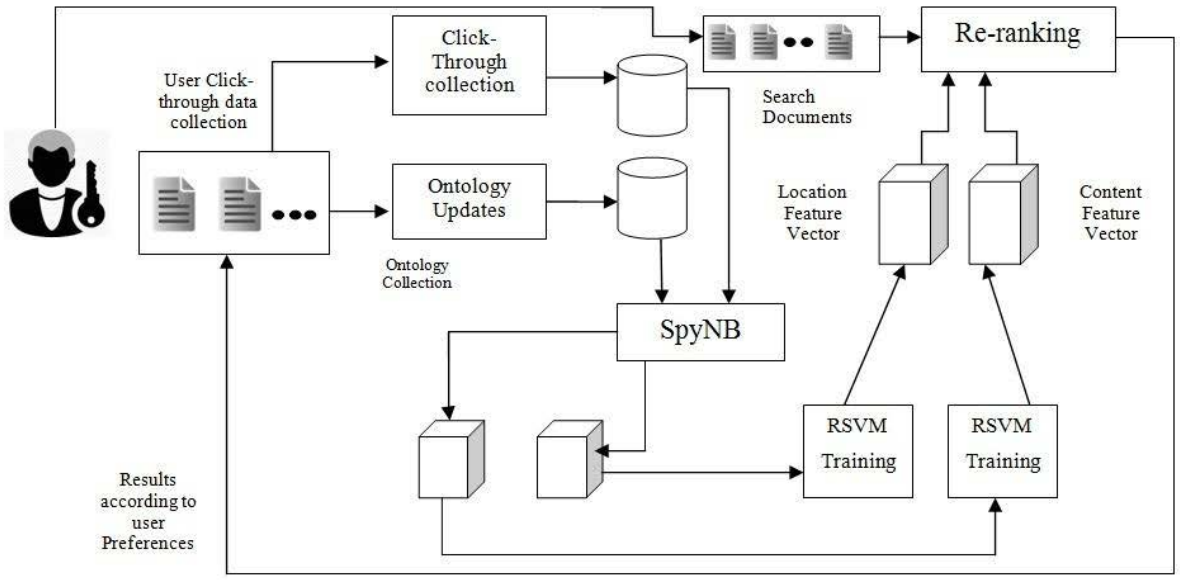

Figure 1 : Overall Architecture of the CLOSE system

Performing the search operations, while this approach leads to true to true personalization but it has some drawback such as:

Fresh start, when user is new to system his/her interest should be tracked and some time user may not compromise to share personal information with the system.

Community-based system [7] describes sharing of the information among several users/models. The data enrichment technique such as clustering technique is used in grouping of the similarity among various users. Using some similarity criteria the users among the web can be grouped into one model, so that results for this community can be personalized.

Aggregate-level system [8] where information gathered is represented in the form of summary for purpose of analysis. The common parameters such as age are considered to form clusters. For example a site selling music CD's may advertise certain CD's based on the age of the users and data aggregate for their age group. Online analytic processing (OLAP) is the simple type of data aggregation.

Browser also provides certain level of personalization by storing the cookies and recently visited web hyperlinks in the buffers. When the user is in static place browser will provide certain level of personalization, but when user place changes dynamically buffer contents are no more used.

For this purpose the new technique can be taken into consideration, such that each user's interest is maintained in the server buffer so that where ever user requests some result in form of query this can be compared with user interest buffer and relevant information can be retrieved from the system by minimizing unrelated results.

III. System Design

Fig 1 shows the complete architecture of the CLOSE system, the working procedure is as follows. When the user is new to system and enters any query for the first time the preferences for location is taken along with search keyword and search operation is performed. The keyword of the query is searched in the server and relevant results are fetched and displayed as the results. When the user clicks on some links, Click through data will be recorded. Later when the user searches for the same keyword, the previously visited pages will be displayed first with higher ranked pages and, if there is are any new links they will be ranked in lower order.

Spy NB [9] is the algorithm used to fetch the user Click through data, and these are transformed to vectors for further process. The Ranked support Vector machine (RSVM) training is performed on the vectors for Re-ranking of search results according to user preferences. The detailed description about Spy NB and RSVM is given in implementation part.

The system mainly concentrates on building the method of ontology for all the possible keywords. The word can have different meaning in different context [2].

For example when the keyword "JAVA" is considered, in several perspectives it mean as the programming language, but by the name JAVA there is an island in Indonesia, and java coffee is referred to as a coffee beans.

When the two users submit the query both will get similar results either list of Java Island or list of java coffee beans is displayed or list of java programming is displayed, but one user expecting only about island and other only programming language. The system mainly

focuses on differentiating which user is really interested in what. For this purpose the ontology is constructed for each keyword with their meaning. The fig 2 shows the construction of ontology for some words.

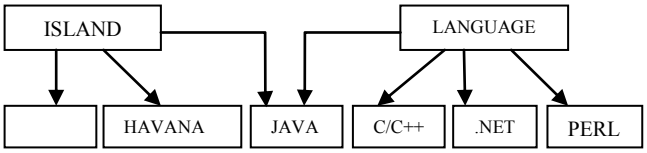

Figure 2 : Ontology for keyword JAVA

Clickthrough data: It is the process of recording the links or advertisement that is clicked by the user(s), for the purpose of determining which link is viewed how many times. The system makes use of these Clickthrough [10] data in personalizing each specific user's interest by maintaining the records for each user in the database. In formal language it can be defined as, it is triplets of (Q, R, C) where Q is the query, R is the ranking order in which it is displayed and C is the set of URLs that are clicked by the users. To achieve personalization the system is classified into two distinct levels namely, content ontology and location ontology [11] [12]. The detailed descriptions about two levels are elaborated in below section:

Content Ontology: The concept works on extracting the keywords/phrase from the web snippets by eliminating all the stems in the query Q. The content ontology is classified differently to different users based on their interest. The co-existence of the keyword in the query Q is calculated to find similarity among the user interest by using following support and confidence rule [3]:

$$Support(c_i) = \frac{sf(c_i)}{n} . |c_i|$$

Where $sf(c_i)$ is the web snippet frequency of the keyword/phrase in the query Q, n is the total number of web snippet and $|c_i|$ is the number of terms in the keyword/phrase c_i. If the support of the keyword/phase ci is higher than threshold ΔT (where threshold ΔT is set by user), than we consider c_i as the concept for query Q.

In this system the value of ΔT is set to 5 because, if ΔT value is assigned with lesser value than for each search, ranking should be updated this leads to consume more time for reordering of links. If ΔT value is assigned with larger value than perfect personalization cannot be achieved.

The following two prepositions are adopted to find relationship between concepts for ontology:

- Similarity: The two concepts which coexist more in the search results can be considered or represented as the same topic of interest. If occurrence of document c_i, c_j > ΔT (where ΔT is the threshold) then c_i and c_j can be considered as similar.
- Parent-Child Relationship: specific concepts appear with general terms, but backtracking is not true. If the preference of c_i and c_j > ΔT then we can conclude that c_i is child of c_j.

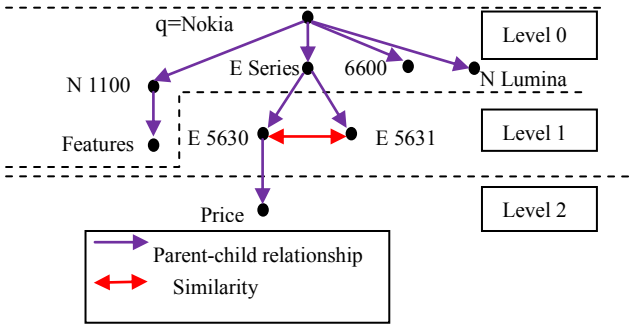

Figure 3 : Ontology's classification for q=Nokia

Fig 3 shows the content ontology for the query q=Nokia, where the concept linked with single head arrow indicates parent child relationship and double head arrow indicates similarity concepts. In the fig 2 the possible concept space determined for the keyword/phrase "Nokia" while Click through data will determine the preferences on the concept based. The concept space for the query "nokia" consists of different types of models such as E-series, N-Lumina etc. When E-series is taken into consideration both has similarity that they belong to same parent.

Content space for the query "Nokia" consists of "N1100", "E-series", "6600", and so on. If the user is interested in E-series and clicks on the page containing price, the Click through of the links are captured. These Click through data is considered as the positive preferences and vector is constructed.

When the same query is issued by the same user later the vector is transferred to server by transforming this content vector into content weight vector to rank the search result according to user preferences.

Location Ontology: The approach of the location ontology [13] [14] [15] is quite different from the construction of content ontology. Following assumptions are made i.e., the parent-child relationship cannot be accurately derived for the location ontology. To construct the vector [15] for location concept following Bangalore, "Jaynagar/Bangalore/Karnataka/India", is associated with the document d.

The construction of the vector for the location ontology is similar to that of the content ontology. The Clickthrough data is transferred to the server and transformed as the location vector and this vector is used to rank the user preferences.

IV. Implementation

In this section technique that are used to personalize the search engine are discussed in detail. First, when the query q is entered by the user, look for previous records if previous search results are found then apply Content ontology concept else if the user is new then accept the query q and apply Location ontology concept.

Algorithm 1: CLOSE (Ui, q, L)

// Input: User identity Ui, Query q and Current location of User L.
// Output: Results for query with user preferences.
1. Accept the Query q from user where q ϵ {A-Z, a-z, 0-9}
2. Filter the post (documents) using the keyword q
If (∀ Post (di) == compare (q))
3. **If** (check user profile Ui for previous records)
4. Result_set ⟵ Content-Ontology (Ui, q) + Location-Ontology (Ui, q, L)
5. Update Ui ⟵ Result_set
Display "Results"
6. **Else**
7. Result_set ⟵ Location-Ontology (Ui, q, L)
8. Update Ui ⟵ Result_set
Display "Results"
9. **End if**
10. **Else**
11. Display "Query Not Matched".

Next algorithm will be related to searching keyword based on Content ontology.

Algorithm 2: Content-Ontology (Ui, q)

// Input: User Identity, and corresponding Query q.
// Output: Return Results to CLOSE
1. Let S ⟵ post (di) matched for q.
2. Retrieved ⟵ SpyNB(S).
3. Let Ps denotes Positive set and Ns denotes Negative set from SpyNB(S) where:
Ps ϵ {Links that are clicked by the users}
Ns ϵ {Links not clicked by the users}
Select Positive Set from Retrieved documents.
4. Count ⟵ Count+ Number_of_clicks.
5. Results ⟵ RSVM (Count, post_code).
6. **Return** Results.

Next algorithm will be related to searching keyword based on Location ontology.

Algorithm 3: Location-Ontology (Ui, q, L)

// Input: User-Identity U_i, Query q and Location L.
// Output: Return Results to CLOSE.
1. Let L ⟵ Current Location of User.
L1 Post-Location.
2. Let S ⟵ Post (di) matched with q && L
3. Calculate distance between current location and Post Location
Difference ⟵ L-L1
4. Result ⟵ Sort post with shortest distance to Higher Distance.
5. **Return** Result

Spy Naive Bayes (SpyNB) algorithm is used to collect the Clickthrough data. This algorithm will maintain two sets called positive set Ps and negative set Ns. Where
P_s ϵ {Links that are clicked by the users}
N_s ϵ {Links not clicked by the users}

Algorithm 4: SpyNB(s)

// Input: Post matched for Query q.
// Output: Feature vector for Post
1. Compare S with the user record.
2. **If** (S ϵ Ui)
3. Select post from the records.
Relevant_Post ⟵ Post (d i).
4. Construct the Positive set and Negative set
5. Update Positive set in corresponding User Buffer.
6. Repeat for all Query q
7. **End if**
8. **Return** Post

Ranking algorithm will rank the results according to the user preferences by calculating the weight of both content and location concepts, for keyword/ key phrase. The content weight of all posts for particular keyword is considered in calculating the ranking order.

The vector support machine is constructed for training the user preferences, loop is entered when the ranking operation is started, and the number of count is recorded for the link whenever the user clicks on it. When the post reaches the minimum threshold value then it will gain a higher order value as compared from rest of the post. The formal representation for performing these is depicted below:

Algorithm 5: RSVM (count, post_code)

// Input: count for each click is taken as the input.
// Output: Ranking order of the posts.

1. For i ⟵ 0 to total_post-1 do
2. Content_weight_count ⟵ count.
3. Calculate the Content weight for particular keyword.
   P_code ⟵ Post_code
4. Content_weight (%) ⟵ $\frac{Pcode\_content\ weight\ Count}{\sum_{i=1}^{n} Conent\ weight\ count}$
5. Final_content_weight ⟵ $\frac{Content\_Weight}{2}$
6. P1 ⟵ $(location)/\ \sum_{i=1}^{n} total\ distance$
7. P2 ⟵ P1-100
8. location_weight_parameter ⟵ $\frac{P1+P2}{2}$
9. Final_rank ⟵ Final_content_weight + location_weight_parameter

V. Experimental Evaluation

The Table 1 gives the dataset of the content ontology construction for some of the keywords. The table mainly consists of unique code for particular root keyword, name of keyword and parent of the corresponding keyword [17].

Table 1 : Statistic of Content Ontology

Unique Code	Keywords	Parent
101	Hotel	0
102	Reservation	101
103	Facilities	101
104	Meeting Room	103
105	Party Hall	103
106	Animal	0
107	Jaguar	106
108	Lion	106
109	Car	0
110	Jaguar	109
111	BMW	109
112	Black Jaguar	107
113	Elephant	106

Unique Code Keywords Parent 101 Hotel 0 102 Reservation 101 103 Facilities 101 104 Meeting Room 103 105 Party Hall 103 106 Animal 0 107 Jaguar 106 108 Lion 106 109 Car 0 110 Jaguar 109 111 BMW 109 112 Black Jaguar 107 113 Elephant 106

In the experimental evaluation "Hotel" is the root word and it has four children such as "Reservation", "Facilities", "Meeting Room", and "Party hall", similarly for others also constructed.

Similarly Table 2 gives the dataset of the location ontology construction for some of the locations. The table mainly consists of location code, Location name, latitude, longitude and parent of location. When location is considered, boundary value of 11 values is taken into consideration.

Table 2 : Statistic of Location Ontology

Location Code	Location Name	Parent	Latitude	Longitude
1	India	0	21.0	78.0
12	Karnataka	200	12.97	77.56
123	Bangalore	201	12.97	77.57
124	Mysore	201	12.303106	76.640228
1231	Jaynagar	202	12.93	77.6
1232	Koramangala	202	12.933881	77.622343
13	Tamil Nadu	200	13.08	80.27
2	London	0	51.51	-0.12
21	Barking and Dagenham	207	51.545268	0.147575
22	Barnet	207	51.650194	-0.200897
23	Bexley	207	51.441811	0.154297

In posting of documents the related information are stored by entering the root and location for which it belongs. In this case Hotel "comfort" comes under Bangalore city for which India will be root, and so on others are posted.

When user enters the query q, the searching process will be carried out as mentioned in the implementation section by invoking several techniques. When the corresponding documents are found, and previous records of users are analyzed, the ranking support vector machine is performed on the posts that are matched by the keyword or query q.

Table 3 gives the RSVM calculation for the Keyword "jaguar for two different users, it can be observe from the table that two user have their own preferences in choosing the link.

Later, when two users search for same keyword then threshold value changes and ranking of their search results will be altered.

Table 3 : RSVM training of the Data sets

Keyword	Posting number	Count	content Weight	Final Content Weight	Location	Distance	P_1	P_2	Final Location Weight	Final Value	Rank
Jaguar User1	1001	0	0	0	Jaynagar	0	0	100	50	50	3
	1002	10	58.82	29.41	Mysore	160	18.47	81.52	40.76	70.17	1
	1003	5	29.41	14.70	Koramangala	6	0.69	99.30	49.65	64.35	2
	1004	2	11.76	5.88	Delhi	700	80.83	19.16	9.58	15.46	4
		Total=17				Total=866					
Jaguar User2	1001	0	0	0	Mysore	0	0	100	50	50	3
	1002	5	29.41	14.70	Jaynagar	160	18.47	81.52	40.76	55.46	2
	1003	5	29.41	17.70	Koramangala	6	0.69	99.30	49.65	64.35	1
	1004	1	5.88	2.94	Delhi	700	80.83	19.16	9.58	12.52	4
		Total=11				Total=866					

VI. Conclusion and Future Enhancement

We can conclude that the CLOSE system will provide better search results as compared to rest of the search engines by considering the users Content and location concepts. CLOSE system will take user preferences in minimizing the possible time for retrieving search results. RSVM training will be performed for each individual user profile, so that system will come to know in what the user is really interested.

As a future enhancement it can be extended by considering time as one of the parameter to even more optimize the search results. The sessions can also be considered as one of the parameter, so that when user stop work at particular instance, later when user get into system, at moment where user stopped working or viewing content of some documents, from that session it should be started (with respect to two or more different systems).

VII. Acknowledgement

Foremost, I would like to express my sincere gratitude to my guide Mr. S G Raghavendra Prasad Assistant Professor, ISE Dept, RVCE, for the continuous support of my M. Tech study, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of writing this technical paper.

Besides my guide, I would like to thank the rest of my M.Tech committee: Dr. Jitendranath Mungara PG Dean, ISE Dept, RVCE, and Dr. Cauvery N K. HOD ISE Dept, RVCE.

References Références Referencias

1. Hele-Mai Haav, Tanel-Lauri Lubi "A Survey of Concept based Information Retrieval Tools on the Web" White paper.
2. Deepika Bhatia et al "Context-aware Personalized Mobile Web Search Techniques-A Review" vol. 2(5), (IJCSIT) International Journal of Computer Science and Information Technologies, 2011, pp. 2440-2443.
3. Baeza-Yates R., RIBEIRO-NETO, B. 2013 Modern Information Retrieval: The Concepts and Technology behind Search, Pearson Edition.
4. M. Rami Ghorab et al "Personalised information retrieval: survey and classification", Centre for Next Generation Localisation Knowledge & Data Engineering Group. Pp. 1-40.
5. Kanika Arora, Kamal kant "Techniques for Adaptive websites and Web Personalization without any user effort" IEEE Students conference on Electrical, Electronics and Computer Science, 2012.
6. Athanasios Papagelis, Christos Zaroliagis "A Collaborative Decentralized Approach to Web Search" vol. 42 No. 5, IEEE Transaction on Systems, Man, and Cybernetics , 2012, pp. 1271-1290.
7. Dou Shen et al "Query Enrichment for Web-query Classification" vol. 24 No. 3, ACM Transactions on Information Systems, 2006, pp. 320-352.
8. Bamshad Mobasher et al "Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization" Vol. 6 No. 1, ACM Transaction on Data Mining and Knowledge Discovery, 2002, pp. 61-82.
9. Wilfred Ng et al "Mining User Preference Using Spy Voting for Search Engine Personalization" Vol. 7 No. 3, ACM Transaction on Internet Technologies,2007, pp. 1-28.
10. Veningston .K, R. Shanmugalakshmi "Enhancing personalized web search re-ranking algorithm by incorporating user profile" IEEE (ICCCNT), 2012, pp. 1-6.
11. LI Qing-shan et al "Ontology based User Personalization Mechanism in Meta Search Engine" IEEE (URKE), 2012, pp. 230-234.

12. Abdelkrim Bouramoul et al "An ontology-based approach for semantics ranking of the web search engines results" IEEE (ICMCS), 2012, pp. 797-802.
13. Varun Mishra et al "Improving Mobile Search through Location Based Context and Personalization" IEEE (ICCSNT), 2012, pp.392-396.
14. Sandeep Jain, Aakanksha Mahajan "Data Mining Based on Semantic Similarity to Mine New Association Rules" Vol. 12 Issue 12 Version 1.2 Global Journal of Computer Science and Technology Software & Data Engineering, 2012.
15. Mingyang Sun et al "FoSSicker: A Personalized Search Engine by Location-Awareness" IEEE (ICNC), 2012, pp. 456-460.
16. Al Sharji Safiya et al "Enhancing the Degree of Personalization through Vector Space Model and Profile Ontology" IEEE Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2013 IEEE RIVF, 2013, pp. 248-252.
    Shikha Goel et al "Search Engine Evaluation Based on Page Level Keywords" IEEE (IACC), 2013, pp. 870-876.
17. Vishwas Raval, Padam Kumar "SEReleC (Search Engine Result Refinement and Classification) - A Meta Search Engine based on Combinatorial Search and Search Keyword based Link Classification" IEEE (ICAESM), 2012, pp. 671-631.

This page is intentionally left blank