IMPROVING INFORMATION RETRIEVAL BASED ON QUERY CLASSIFICATION ALGORITHM

Myomyo Thannaing[1], Ayenandar Hlaing[2]

[1,2] University of Technology (Yadanarpon Cyber City), near Pyin Oo Lwin, Myanmar

ABSTRACT

The ongoing struggle of information retrieval systems is to present users with information that most relevant user needs. So, IR researchers have begun to expand their efforts to understand the nature of the information need that users express in their queries. If system is able to understand the intension behind user's needs and contents, it will retrieve more accurate results. This system presents algorithm and techniques for increasing a search service's understanding of user search queries. Web query classification is to classify a web search query into a set of user intended categories. Previous query classification techniques performed classification process on query logs and neighbouring queries in search session time. We propose Query Classification Algorithm (QCA) for automatic topical classification of web queries based on domain specific ontology. Ontology is a specialization of concepts in domain and relationships that holds between those concepts. Using ontology as a controlled vocabulary in the process of classification, performance accuracy is improved in the classification process. Evaluation of classification accuracy and retrieving performance are explored. The system measures the performance accuracy of retrieving documents by using the number of documents relevant with the user intended category by the total number of retrieved documents. Classification accuracy is measured with recall, precision and f-measure.

KEYWORDS

Intelligent Information retrieval, Query Classification Algorithm (QCA), Domain Term Extraction Algorithm, Domain Ontology.

1. INTRODUCTION

Search engines have become one of the most popular tools for web users to find their desired information. If user searches information, he has an idea of what he wants but user usually cannot formalize the query. As a result, understanding the nature of information need behind the queries issued by Web users have become an important research problem. Classifying web queries into predefined target categories, also known as web query classification, is important to improve search relevance and online advertising. Successfully classification of incoming general user queries to topical categories can bring improvements in both the efficiency and the effectiveness of general web search. There are several major difficulties which are needed to consider in query classification. First, many queries are short and query terms are noisy.

A second difficulty of web query classification is that a user query often has multiple meanings. Web query classification aims to classify user input queries, which are often short and ambiguous, into a set of target categories. Query Classification has many applications including page ranking in Web search, targeted advertisement in response to queries, and personalization. In this paper, we propose Query Classification Algorithm, denoted as (QCA), classifies user

queries into the intended categories for ranking purpose. After the query classification process, input query is labeled with one or more categories sorted according to their scores. Domain ontology is used as a controlled vocabulary. The creation of domain ontology is also fundamental to the definition and use of an enterprise architecture framework. The process of classification queries based on the ontology is presented to improve accuracy value for retrieving information. This intends to provide better search result pages for users with interests of intended categories in top list, for digital library system. The rest of the paper has been organized as follows. Section 2 presents the some of the existing techniques related to query classification. Overview of the proposed system is discussed in Section 3. Query classification process is presented in section 4. And domain term extraction algorithm is explained in Section 5 and Query classification algorithm is in section 6. Section 7 is about implementation of classification-based information retrieval system. Performance measure of proposed system is described in 8. Section 9 is about overall performance accuracy of proposed system. Evaluation of ontology-based classification accuracy is described in 10. This paper is concluded in Section 11. Section 12 is about future work.

2. RELATED WORKS

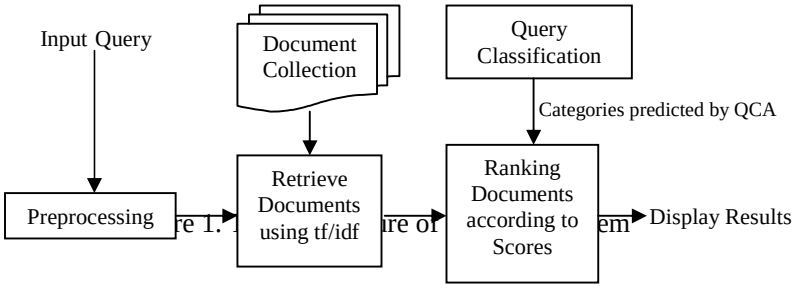
User query classification is an important step for a number of information retrieval. The task of web query classification is to classify user search query into categories. Lovelyn proposed Web Query Classification based on Normalized Web Distance in [1]. In this system, intermediate categories are mapped to the required target categories by using direct mapping and Normalized Web Distance (NWD). The feature set is the set of intermediate categories retrieved from a directory search engine for a given query. The categories are then ranked based on three parameters of the intermediate categories namely, position, frequency and a combination of frequency and position. In [2], the system proposed Taxonomy-Bridging Algorithm to map target category. The target categories typically does not have associated training data, the KDD CUP 2005 is used. The Open Directory Project (ODP) is used to build an ODP-based classifier.

This taxonomy is then mapped to the target categories using Taxonomy-Bridging Algorithm. Thus, the post-retrieval query document is first classified into the ODP taxonomy, and the classifications are then mapped into the target categories for web query. The system is considered to address the problem of query classification by using conditional random field (CRF) models in [3]. This system uses neighbouring queries and their corresponding clicked URLs (Web pages) in search sessions as the context information. The system is not able to find a search context if the query is located at the beginning of search session. Beitzel exploits both labeled and unlabeled training data for web query classification system in [4]. Diemert and Vandelle propose an unsupervised method based on automatically built concept graphs for query categorization in [5]. Ernesto William presents an approach to classify search results by mapping them to semantic classes that are defined by the senses of a query term. The criteria defining each class or ‘sense folder’ are derived from the concepts of an assigned ontology in [6]. Some work has been dedicated to using very large query logs as a source of unlabeled data to aid in automatic query classification. In our proposed approach, domain ontology is used as controlled vocabulary for query classification. This proposed system combines the query classification algorithm with the benefits of statistical approaches based on IR techniques.

3. OVERVIEW OF THE PROPOSED SYSTEM

Figure 1 is shown as step by step procedure of the model. The user input query is passed into the keyword-based search engine which uses TF/IDF approach. Meanwhile, domain terms of input query are extracted by using domain term extraction algorithm. These domain terms are inputs of

Query Classification Algorithm (QCA), which is used to label the input query into the intended categories. After query classification process, the result documents are ranked according to the scores of important categories predicted by the Query Classification Algorithm instead of term frequency. Query classification process is detail explained next section.



4. QUERY CLASSIFICATION PROCESS

Query classification process is presented as figure 2. Domain terms of input query are first extracted by domain term extraction algorithm. These domain terms are inputs of Query Classification Algorithm (QCA) and user intended categories are outputs of algorithm.

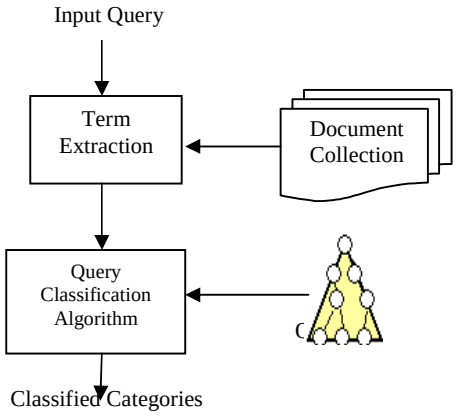


Figure 2. Query Classification Process

5. DOMAIN TERM EXTRACTION ALGORITHM

Domain terms of input query are extracted by using Domain term extraction algorithm from Domain corpus. This algorithm works incrementally by first computing the frequency of one-grams and then considering tri-grams of increasing length, each time keeping those which occur with a frequency above a threshold. Instead of using fixed size n, by varying the length of gram from of one to three, domain terms of all the possible characters appearing in that corpus are got. It is called the set of generalized character n-grams [7]. Domain term extraction algorithm is presented in below.

Input	:	Domain Corpus, *minf*, input query
Output	:	the set S of domain terms

```
Begin
        W1=Tokenize (input query)
        W2=StopWords (W1)
        S= W2 with frequency>minf
        i=2;
        Repeat
                {
                Si=0;
                G=PairWords(S,i);
                for each occurrence of i-grams in G
                {
                        N(g)=CountofWordinCorpus(C,g);
                        if(N(g)>minf)
                        Si=Si+g;
                }
                S=S U Si;
                i=i+1;
                }
                Until i=3;
        Return S;
End
```

Let C be the domain corpus, minf be a threshold, and N(g) be the count of n-gram in C. In this algorithm, three inputs are considered to get domain terms of input query. Domain corpus of computer science area is used to extract domain terms. It is necessary to tokenize the user input query firstly. And then, stopword removal process (i.e. a, an, of, the, is, are, was, were, when, where, what, before, after, since, ago and etc) is needed to get necessary keywords. minf is a threshold number to define domain term.

For example (1),

Input query: “Decision Making Cluster Head Selection System for Wireless Sensor Networks”.

This input string is tokenized and then stopword removal is processed. After this algorithm searches each keyword in the domain corpus and continues for bi-grams and tri-grams.

Output: “decision making, cluster head selection, cluster, cluster head, sensor network, sensor, wireless, wireless sensor network, cluster”

.

6. QUERY CLASSIFICATION ALGORITHM

The aim of query classification is to classify a user query Qi into a list of n categories ci1, ci2,…cin, where cij selected from set of N categories {ci1, ci2,….cin} [8]. Among the output ci1 is ranked higher than ci2 and ci2 higher than ci3 and so on. Extracted Domain terms of user query are used as input. The matched terms of each domain terms are the set of terms defined in the domain ontology. Algorithm is presented as follows.

Input: Ontology, Domain Terms
Output: User intended categories

Begin

Step (1): Extracting Matched Terms for each domain terms
Step (2): Probability for each domain terms

Input: Domain Ontology O, Extracted matched terms T

Output: P = {p_{11}, p_{12}, …, p_{cw}}

```
Begin
    N(C, T) = 0;
    for eachword t in T
    {
        for each concept c in O
        {   If (c.contains (t))
            N(c, t) ++;
        }
    }
    P(C, T) = 1/N(C, T)
    Return
End
```

After computing the probability of terms, the value of each category which contains matched terms is calculated in equ(1).

Step (3): Compute Value (C): the value of particular category containing matched terms.

$$Value(C) = \frac{P(C,T) \times \text{no of matched terms for particular category}}{\text{Total no of matched terms}} \quad (1)$$

For more than one domain terms, the system decides important categories by summation the value of same category for all terms shown in equ (2).

Step (4): Compute Score(C): the score of each category for all domain terms

$$Score(c) = \sum_{C=1}^{n} groupbyCategory\,(Value(C)) \quad (2)$$

End

Example of query classification algorithm is shown in below.
User query: “Query Process of Natural Language statement Using Metadata”
Domain terms: “Query Process”, “Natural Language”, “Metadata”

Step (1): Query processing, Natural language processing, Natural language, Natural language interfaces, Metadata.

Step (2): For the term “Query process” relates to Intelligent_Database category and probability is 1. For “Natural Language”, it relates two categories such as Artificial_Intelligent and Information_system and the probability of each matched category is 0.5. For “Metadata”, it relates to Intelligent Database category and probability is 1.

Step (3): The value for “Query process” is 1.The values of each category for “Natural Language” is (0.5*(2/3) =0.334) and (0.5*(1/3) =0.167), respectively. The value for “Metadata” is 1.
Step (4): Finally, scores for Intelligent Database is 2, Artificial Intelligent is 0.334, and Information system is 0.167.

7. IMPLEMENTATION OF PROPOSED SYSTEM

To implement the classification-based information retrieval system, software implementation, ontology construction, basic requirements for ontology construction, categories of ontology are detail explained in below.

7.1. Software Implementation

To implement the proposed ontology-based query classification for information retrieval system, J2EE is used for implementation information retrieval system. Protégé-OWL plug-in is used to create domain ontology in the area of computer science as our case study. Terms from ontology are extracted using java embedded with SPARQL language for via Jena Ontology, based Jena Library.

7.2. Ontology Construction

Ontology renders shared vocabulary and taxonomy which models a domain with the definition of objects and/or concepts and their properties and relations [9]. Using ontology as a controlled vocabulary, accuracy value is improved in retrieving information. In here, ontology is an information model containing vocabularies and relation in the area of computer science as our case study. We assume ontology is organized as directed acyclic graphs. Each node represents a class and there is relation between them. In this system, ontology is constructed for query classification process.

7.3. Basic Requirements for Ontology Construction

In this system, ontology is supported through the protégé-OWL plug-in. Protégé is a knowledge modelling environment. Protégé core is based on Frames (object oriented) modelling. It has an open architecture and supports development of plug-in to allow backend/interface extensions. Field ontology is constructed based on ACM (Association for Computing Machinery) and Wikipedia according to the domain area (Computer Science).

7.4. Categories of Ontology

In construction of ontology model, concept and property relationship in professional field are defined and field ontology is constructed based on [10] and [11], according to the professional field (Computer Science).

There are categories in computer science domain encoded as classes. In this system, there are 12 classes such as Artificial_Intelligent, Cloud_Computing, Computer_Networking, Data_Mining, Digital_Signal_Processing, Image_Processing, Information_System, Intelligent_DataBase,

Internet_and_Distributed_Computing, Mobile_Computing, Natural_Language_Processing, Software_Engineering.

These categories consist of several subcategories or subclasses. For example, Artificial Intelligent has subcategories namely AI learning, Expert System, Natural language processing, Distributed Artificial Intelligent, Knowledge representation, Control method and Deduction and Theorem Proving.

Ontology is applied in the process of query classification to get the concepts of each term. The terms from ontology are queried to further process by using SPARQL 1.1 language. Terms from ontology are extracted using Java embedded with SPARQL language for via Jean Ontology API. Figure 3 shows the example of the class and subclass relationship of Artificial Intelligent.

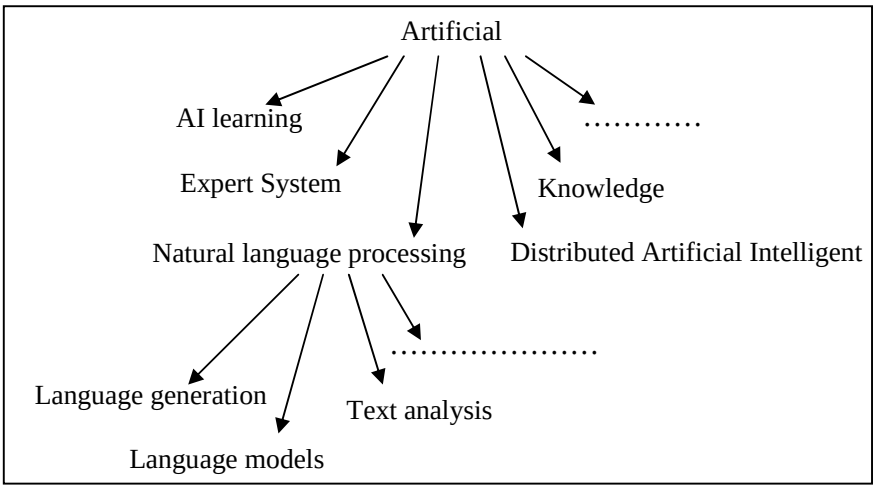


Figure 3. Class and Subclass Relationship of Artificial Intelligent

8. PERFORMANCE MEASURE

To evaluate the effectiveness of classification-based information retrieval system, a measure for both retrieval systems is used to compare them. It is important to analyze how search documents are relevant with user intended category. Therefore, result documents from proposed system and keyword-based system are measured to evaluate the relevance with user intended category by using equ (3).

$$\text{Accuracy Value} = \frac{\text{the number of relevant documents with user intended category}}{\text{the total number of retrieved documents}} \quad (3)$$

The relevance of result documents with user intended category is measured the number of relevant documents with user intended category by the total number of retrieved documents. As we assume, documents are ranked according to the scores of cosine similarity, the top result documents are more close to user requirements. So, top 100 of retrieved documents are considered as the total number of retrieved documents. For an input string A, “Query Process of Natural Language Statement Using Metadata”, comparison of accuracy values of proposed system with keyword search system is shown in table1.

Table 1. Accuracy Evaluation for input string A

Input string	Type of retrieval	No of Relevant Documents With Categories	Total No of Retrieved Documents	Accuracy Value
A	Proposed System	97	100	0.97
A	Keyword Search System	81	100	0.81

9. THE OVERALL PERFORMANCE ACCURACY OF PROPOSED SYSTEM

The analysis has been done on the four datasets with varying number of domain terms to analyze overall accuracy value for proposed system and keyword search system using equ (3). Four datasets are queries of each category that contains one, two, three and four domain terms respectively. Each category for each dataset has 10 input queries.

According to the evaluation, if user input query has many domain terms, the proposed system results more number of relevant documents with user intended category than keyword search system. For example, three out of twelve categories are shown in table 2 to describe the accuracy differences between proposed system and keyword search system for four datasets. Let n be the number of domain terms in query. Accuracy comparisons for three categories are shown in figure 4, 5 and 6.

Table 2. Accuracy Differences between Proposed System and Keyword Search System for Three Categories

category	n=1		n=2		n=3		n 3	
	propose d system	keyword system	proposed system	keyword system	propose d system	keyword system	proposed system	keyword system
Artificial Intelligent	3.563	3.438	4.804	3.644	5.51	3.63	7.16	4.81
Data Mining	7.164	6.563	7.791	6.161	7.07	5.79	6.68	4.61
Digital Signal Processing	7.367	7.346	7.173	6.053	7.398	5.588	7.85	5

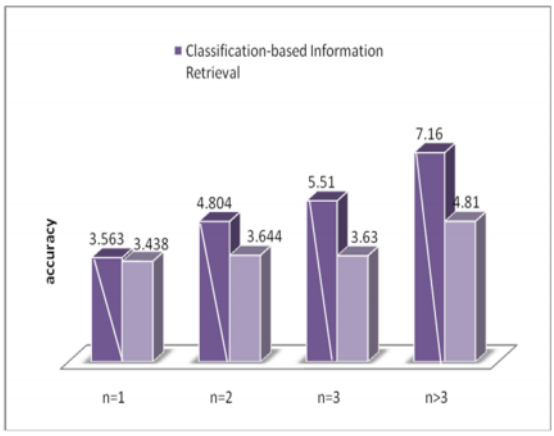


Figure 4. Accuracy Comparison for Artificial Intelligent

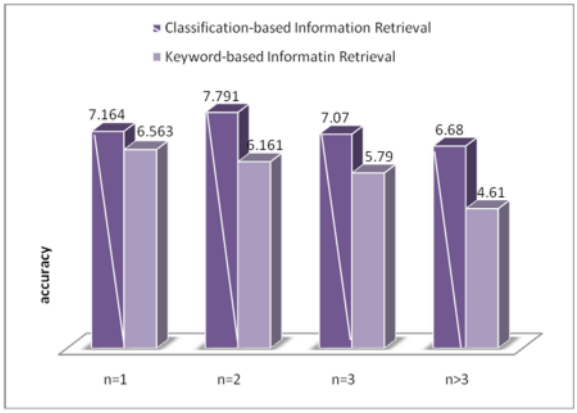


Figure 5. Accuracy Comparison for Data Mining

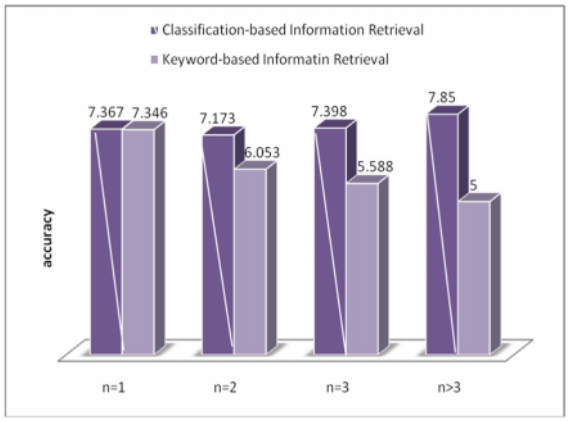


Figure 6. Accuracy Comparison for Digital Signal Processing

10. EVALUATION OF ONTOLOGY-BASED CLASSIFICATION ACCURACY

To classify the user input query into the intended category, classification system is based on domain specific ontology. No training data set is not used to classify input query. To analyze the accuracy of classification process, we measure precision, recall and F-measure of the classification system in equation (4), (5), (6).

Table 3. Measurement of accuracy for Classification

		Categorization by Domain Expert	
		Correct	Incorrect
Categorization by the System	Correct	a	b
	Incorrect	c	d

$$\text{Recall} = \frac{a}{(a+c)} \quad (4)$$

$$\text{Precision} = \frac{a}{(a+b)} \quad (5)$$

$$\text{F measure} = (2 * (\text{Recall} * \text{Precision}))/(\text{Recall} + \text{Precision}) \quad (6)$$

600 queries test data set is used to evaluate classification accuracy. As the experiment, correctly classified queries are 520 out of 600 and incorrectly classified queries are 80 in 600. Recall is 0.833 Precision is 0.890 and f-measure is 88.47%.

11. CONCLUSIONS

The idea of using the concepts of ontology to classify the input search query is explored for improving search results in informational retrieval system. Since classification of web queries into predefined target categories is important in the effectiveness of web search, ontology-based query classification algorithm (QCA) is proposed. This can be provided relevant results with user intended category. According to the experimental results, the proposed system can outperform than traditional keyword search system.

12. FUTURE WORK

This system can be used in interesting application area by using specific domain. Further research can be done to implement other domain areas of information retrieval system by adopting respective domain specific ontology. This classification algorithm can be applied by using other hierarchical-based taxonomy instead of ontology. As the future work, the scope of ontology can be extended to be more effective.

REFERENCES

[1] S. Lovely Rose, K.R. Chandran ," Normalized Web Distance Based Web Query Classification", Journal of Computer Science 8 (5): 804-808, 2012

[2] D. Shen, J. Sun, Q. Yang, Z. Chen, "Building bridges for Web query classification". In: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, Seattle, WA, USA (2006)131-138.

[3] H. Cao, D. Hao Hu, D.Shen., D. Jiang, , J.T.Sun, E.Chen,Q.Yang, "Context-Aware Query Classification",(July 19–23, 2009).

[4] S. Beitzel, E. Jensen, O. Frieder, D. Grossman, D. Lewis,A. Chowdhury, and A. Kolcz,"Automatic web query classification using labeled and unlabeled training data". InProceedings of SIGIR'05, 2005. In: Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval, Salvador, Brazil (2005) 581-582.

[5] E. Diemert, G. Vandelle,: "Unsupervised query categorization using automatically-built concept graphs". In: Proceedings of the 18th international conference on World Wide Web, Madrid, Spain (2009).

[6] E.W. De Luca, A. Nürnberger, "Ontology-Based Semantic Online Classification of Documents: Supporting Users in Searching the Web".

[7] C. Marques and Anges Braud.: "Mining Generalized Chapter n-Grams in Large Corpora".

[8] J.-R. Wen, J.-Y. Nie, and H.-J. Zhang, " Query clustering using content words and user feedback". In SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pages 442–443,2001.

[9] S. Lovely Rose, K.R.. Chandran ," Normalized Web Distance Based Web Query Classification", Journal of Computer Science 8 (5): 804-808, 2012.

[10] http://www.acm.org/class.

[11] http://en.wikipedia.org/wiki/Category:Computer_science.