IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 11, November 2015*

# An Efficient Semantic Dynamic Query processing based on user interest for web Database using FCM Algorithm

**Prabha .P[1], Vijayakumar .P[2]**

Research Scholar, Department of Computer Science,

Sri Jayendra Saraswathy Maha Vidyalaya College of Arts and Science, Coimbatore[1]

Head, Department of Computer Applications,

Sri Jayendra Saraswathy Maha Vidyalaya College of Arts and Science,Coimbatore[2]

**Abstract:** In this journal, we aim at discovering the number of diverse user search goals for a query and depicting each goal with some keywords automatically. we propose a semantic ontology method to map feedback sessions to pseudo-documents which can efficiently reflect user information needs. At last, we cluster these pseudo documents to infer user search goals and depict them with some keywords. Since the evaluation of clustering is also an important problem, we also propose a novel evaluation criterion fuzzy score to evaluate the performance of the restructured web search results.In this paper, we propose an efficient approach to improve user search goals by analyzing search engine query logs automatically. And propose a framework to discover dissimilar user search goals for a query by clustering the proposed automatic feedback process.

**Keywords:** Clusters,C-Means,Metadata,Classification

## 1.INTRODUCTION

Measuring the semantic alike among vocabulary is a significant factor in a range of responsibilities on the internet such as similarity mining, text clustering, and atomized metadata mining. The system implements an experiential process to approximate semantic likely-hood using page calculations and text fragments retrieved from a web search engine for two words. Specifically,It define various word co-occurrence measures using page counts and integrate those with lexical patterns extracted from text snippets. To identify the numerous semantic relations that exist between two given words, we propose a novel pattern extraction algorithm and a pattern page counts-based co-occurrence measures and lexical pattern clusters is learned using support vector clustering algorithm. The optimal combination of machines.Discovered taxonomical patterns from the users' local text documents to learn ontologies for user profiles.Some groups learned personalized ontologies adaptively from user's browsing history.Alternatively, analyzed query logs to discover user background knowledge. In some works, such as, users were provided with a set of documents and asked for relevance feedback. User background knowledge was then discovered from this feedback for user profiles.However,because local analysis techniques rely on data mining or classification techniques for knowledge discovery, occasionally the discovered results contain noisy and uncertain information.As a result,local analysis suffers from ineffectiveness at capturing formal user knowledge.From this,we can hypothesize that user background knowledge can be better discovered and represented if we can integrate global and local analysis within a hybrid model.

## 2.LITERATURE REVIEW

Given taxonomy of words, a straightforward method to calculate similarity between two words is to find the length of the shortest path connecting the two words in the taxonomy. If a word is polysemous, then multiple paths might exist between the two words. In such cases, only the shortest path between any two senses of the words is considered for calculating similarity. A problem that is frequently acknowledged with this approach is that it relies on the notion that all links in the taxonomy represent a uniform distance.

The clustering problem is a difficult problem for the data stream domain. This is because the large volumes of data arriving in a stream renders most traditional algorithms too inefficient. In recent years, a few one-pass clustering algorithms have been developed for the data stream problem. Although such methods address the scalability issues of the clustering problem, they are generally blind to the evolution of the data and do not address the following issues: (1) The quality of the clusters is poor when the data evolves considerably over time. (2) A data stream clustering algorithm requires much greater functionality in discovering and exploring clusters over different portions of the stream [1].

The problem we aim to solve is the diversification of search results for ambiguous web queries. We present a model based on knowledge of the diversity of query subtopics to generate a diversified ranking for retrieved documents. We expand the original query into several related queries, assuming that query expansions expose subtopics of the original query. Moreover, each query expansion is given a weight which reflects the likelihood

of the interpretation (the fraction of users who issued this query given the general query topic). We issue all those expanded queries including the original query to a standard BM25 search engine, then re-rank the retrieved documents to generate the final ranking. Our method can detect possible subtopics of a given query and provide a reasonable ranking that satisfies both relevancy and diversity metrics. The TREC evaluations show our method is effective on the diversity task. [2]

Ranking and returning the most relevant results of a query is a popular paradigm in Information Retrieval. We discuss challenges and investigate several approaches to enable ranking in databases, including adaptations of known techniques from information retrieval. We present results of preliminary experiments. [3]

Measuring similarity or distance between two entities is a key step for several data mining and knowledge discovery tasks. The notion of similarity for continuous data is relatively well-understood, but for categorical data, the similarity computation is not straightforward.

Several data-driven similarity measures have been proposed in the literature to compute the similarity between two categorical data instances but their relative performance has not been evaluated. In this paper we study the performance of a variety of similarity measures in the context of a specific data mining task: outlier detection. Results on a variety of data sets show that while no one measure dominates others for all types of problems, some measures are able to have consistently high performance.[4]

Relational database systems are becoming increasingly popular in the scientific community to support the interactive exploration of large volumes of data. In this scenario, users employ a query interface (typically, a web-based client) to issue a series of SQL queries that aim to analyze the data and mine it for interesting information.

First-time users, however, may not have the necessary knowledge to know where to start their exploration. Other times, users may simply overlook queries that retrieve important information. To assist users in this context, we draw inspiration from Web recommender systems and propose the use of personalized query recommendations.[5]

We investigate the problem of ranking the answers to a database query when many tuples are returned. In particular, we present methodologies to tackle the problem for conjunctive and range queries, by adapting and applying principles of probabilistic models from information retrieval for structured data.

Our solution is domain independent and leverages data and workload statistics and correlations. We evaluate the quality of our approach with a user survey on a real database. Furthermore, we present and experimentally evaluate algorithms to efficiently retrieve the top ranked results, which demonstrate the feasibility of our ranking system.[6]

## 3.PROPOSED TECHNIQUES

In the proposed system, this thesis presents a complete framework and an experience in mining Web usage patterns with real-world challenges such as evolving access patterns, dynamic pages, and external data describing ontology of the Web content and how it relates to the business actors. The Web site in our study is managed by a nonprofit organization that does not sell anything but only provides free information that is ideally complete, accurate, and up to date. To understand the different modes of usage and to know what kind of information the visitors seek and read on the Web site and how this information evolves with time, we perform clustering of the user sessions extracted from the Web logs to partition the users into several homogeneous groups with similar activities and then extract user profiles from each cluster as a set of relevant URLs. This procedure is repeated in subsequent new periods of Web logging (such as biweekly), then the previously discovered user profiles are tracked, and their evolution pattern is categorized. The Web site hierarchy is inferred both from the URL address and from a Web site database that organizes most of the dynamic URLs. We also enrich the cluster profiles with various facets, including search queries submitted just before landing on the Web site. The automatic identification of user profiles is a knowledge discovery task consisting of periodically mining new contents of the user access log files .

### 3.1 Proposed algorithm

**FCM clustering algorithm**

As the FCM algorithm is very sensitive to the number of cluster centers, cluster centers initialization often artificially get significant errors, and even get the actual opposite results .FCM algorithm is hard on data sets too, so the data sets must be quite regular, in order to solve problems, first of all we use information entropy to initialize the cluster centers to determine the number of cluster centers. It can be reduce some errors, and also can Improve the algorithm introductions weighting parameters after that combine with the merger of ideas and divide the large chumps into small clusters. Then merge various small clusters according to the merger of the conditions, so that you can solve the irregular datasets clustering. Document similarity measures as shown in below.

The algorithm as follows

Initialize number of clusters
Intialize $C_j$ (cluster centers)
Intialize $\alpha$ (threshold value)
Repeat
For i=1 to n:update $\mu_j(X_i)$
For k=1 p;
Sum=0
Count=0
For i=1 to n;
If $\mu(X_j)$ is maximum in $C_k$ then
If $\mu(X_j)>= \alpha$
Sum=sum+$X_i$


DOI 10.17148/IJARCCE.2015.41164

Count-count+1
$C_k$ =sum/count
Until $C_j$ estimate stabilize

The clustering framing as follows
A set clusters C={$C_1$, $C_2$, $C_3$,.....$C_k$}
Maximum precision values:

$$\text{Purity}= \sum_{i=1}^{k}\left(\frac{(|C_i|)}{n}\right) Max_{j=1}^{n} \Pr ecision(C_i, L_j)$$

$$\text{Precision}(C_i,L_j)= \left(\frac{(|C_i \cap L_j|)}{|C_i|}\right)$$

$$\text{Inverse Purity}= \sum_{i=1}^{m}\left(\frac{(|L_j|)}{n}\right) Max_{j=1}^{k} \text{Re}call(C_i, L_j)$$

Recall(Cj,Li)=Precision(Li ,Cj)
To calculate the harmonic mean, the F-means

$$\text{Purity-F}= \sum_{i=1}^{m}\left(\frac{(|L_j|)}{n}\right) Max_{j=1}^{k}\{F(C_j, L_i)\}$$

Where the maximum is taken over all cluster F(Cj,Li) is defined as

$$\text{F(Cj,Li)}= \frac{2\times \text{Re}call(C_j, L_i)\times \Pr ecision(C_j, L_i)}{\text{Re}call(C_j, L_i)+\Pr ecision(C_j, L_i)}$$

Web usage data set

Fuzzy C-means (FCM) is a method of clustering which allows one pixel to belong to two or more clusters. The FCM algorithm attempts to partition a finite collection of pixels into a collection of "C" fuzzy clusters with respect to some given criterion. Depending on the data and the application, different types of similarity measures may be used to identify classes. Some examples of val ues that can be used as similarity measures include distance, connectivity, and intensity. In this work, the images are segmented into four clusters namely white matter, particular cluster which can be easily extracted. But grey matter, CSF and the abnormal tumor region based on the feature values.

### 3.2 Processing Strategies

### Log Creation

All the feedback sessions of a query are first extracted from user click-through logs and mapped to pseudo-documents. Then, user search goals are inferred by clustering these pseudo-documents and depicted with some keywords. Since we do not know the exact number of user search goals in advance, several different values are tried and the optimal value will be determined by the feedback from the bottom part the original search results are restructured based on the user search goals inferred from the upper part. Then, we evaluate the performance of restructuring search results by our proposed evaluation criterion CAP. And the evaluation result will be used as the feedback to select the optimal number of user search goals in the upper part.

### Auto feedback updations

Generally, a session for web search is a series of successive queries to satisfy a single information need and some clicked search results . In this paper, we focus on inferring user search goals for a particular query. Therefore, the single session containing only one query is introduced, which distinguishes from the conventional session. Meanwhile, the feedback session in this paper is based on a single session, although it can be extended to the whole session. The proposed feedback session consists of both clicked and unclicked URLs and ends with the last URL that was clicked in a single session. It is motivated that before the last click, all the URLs have been scanned and evaluated by users. Therefore, besides the clicked URLs, the unclicked ones before the last click should be a part of the user feedbacks. of a feedback session and a single session., the left part lists 10 search results of the query “the sun” and the right part is a user’s click sequence where “0” means “unclicked.”

Since feedback sessions vary a lot for different click-throughs and queries, it is unsuitable to directly use feedback sessions for inferring user search goals. Some representation method is needed to describe feedback sessions in a more efficient and coherent way. There can be many kinds of feature representations of feedback sessions. a popular binary vector method to represent a feedback session. Same as Fig. 3, search results are the URLs returned by the search engine when the query “the sun” is submitted, and “0” represents “unclicked” in the click sequence.

### Semantic Learning User Pattern

Search log sessions contain a large number of paraphrases contributed by users during query rewriting. However, it is a big challenge to distinguish paraphrases from the simply related queries in the sessions. This paper addresses this problem by making innovative use of user behavior information embodied in query sessions. Specifically, we learn paraphrase patterns from the search log sessions with a classification framework, in which three types of user behavior features are exploited besides the conventional features. We evaluate the method using a query log of a commercial search engine. Experimental results demonstrate the effectiveness of our method, especially the significant contribution of the user behavior features.

### Clustering Average Precision

Since search engines always return millions of search results, it is necessary to organize them to make it easier for users to find out what they want. Restructuring web search results is an application of inferring user search goals. We will introduce how to restructure web search results by inferred user search goals at first. Then, the evaluation based on restructuring web search results will be described. The inferred user search goals are represented by the vectors in and the feature representation of each URL in the search results can be computed. Then, we can categorize each URL into a cluster centered by the inferred search goals. In this paper,


DOI 10.17148/IJARCCE.2015.41164

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

*International Journal of Advanced Research in Computer and Communication Engineering*
*Vol. 4, Issue 11, November 2015*

we perform categorization by choosing the smallest distance between the URL vector and user-search-goal vectors. By this way, the search results can be restructured according to the inferred user search goals In order to apply the evaluation method to large-scale data, the single sessions in user click-through logs are used to minimize manual work. Because from user click-through logs, we can get implicit relevance feedbacks, namely "clicked" means relevant and "unclicked" means irrelevant. A possible evaluation criterion is the average precision which evaluates according to user implicit feedbacks. AP is the average of precisions computed at the point of each relevant document in the ranked sequence,

## 4.EXPERIMENTAL RESULTS AND DISCUSSION

### 4.1 Dataset Description

The results of documents clustering for both algorithms (FCM and K Means ). As a set of documents we used 1000 random texts from the Yahoo Finance dataset of the companies' descriptions. We partitioned the set into 5 clusters using the same initial distributions and the same shared parameters. For each cluster we provide the mean inner similarity value, the number of documents and the three most characteristic keywords. The clusters are aligned therefore the results can be directly compared. It is evident that both algorithms found similar clusters. The average mean similarity is lower for c-means which might be the result of better centre localization of c-means.

As ontologies prevail in powering web applications, the task of identifying the right ontologies from apparently similar ones becomes increasingly important. This paper investigated a method that combines logic formalisms together with the emerging new web resources. More specifically, we dissolve the formal representations of ontologies into signatures. Keywords are then extracted from the signatures. We employ semantic preserving weight schema to evaluate how significant these keywords contribute to constructing the ontology. In order to minimise the intra- and inter-individual modelling variance, all the weighted keywords are projected onto a carefully crafted text corpus composed by selected While, obviously, there are many important issues to address, the crux of our immediate future work lies in the lack of automated evaluation mechanism. Inputs from human experts will be gathered to compare and contrast results produced by our approach. In the meantime, we will investigate the possibility of embedding this indexing algorithm into existing ontology editors allowing ontology engineers to generate, scrutinise, and fine-tuning the ontology index at design time.The contribution of page counts-based similarity measures, and lexical patterns extracted from snippets, on the overall performance of the proposed method. To evaluate the effect of page counts-based cooccurrence measures on the proposed method, we generate feature vectors only using the four page counts-based cooccurrence measures, to train an SVM. Similarly, to evaluate the effect of snippets, we generate feature vectors only using lexical pattern clusters. From Table 8, we see that on all three data sets, snippets have a greater impact on the performance of the proposed method than page counts.

| K-Means | FCM | SVM | Agglomerative |
|---|---|---|---|
| 75.2 | 99.73 | 80.34 | 73.5 |
| 73.2 | 99.31 | 81.02 | 74.1 |
| 76.3 | 99.82 | 79.89 | 72.9 |

**Table .1 Accuracy Table**

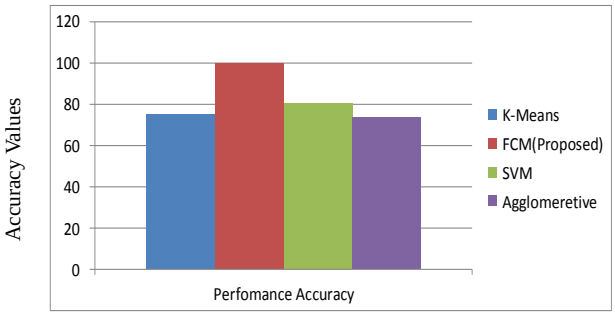
By considering both page counts as well as snippets, we can further improve the performance reported by individual methods. The improvement in performance when we use snippets only is statistically significant over that when we use page counts only in RG and WS data sets. However, the performance gain in the combination is not statistically significant. We believe that this is because most of the words in the benchmark data sets are common nouns that co-occur a lot in web snippets. On the other hand, having page counts in the model is particularly useful when two words do not appear in numerous lexical patterns.

### 4.2 Comparison Chart

Measuring the semantic similarity between named entities is vital in many applications such as query expansion, entity disambiguation (e.g., namesake disambiguation), and community mining . Because most named entities are not covered by WordNet, similarity measures that are based on WordNet cannot be used directly in these tasks. Unlike common English words, named entities are being created constantly. Manually maintaining an up-to-date taxonomy of named entities is costly, if not impossible. The proposed semantic similarity measure is appealing for these applications because it does not require precompiled taxonomies. In order to evaluate the performance of the proposed measure in capturing the semantic similarity between named entities, we set up a community mining task. We select 50 personal names from five communities: tennis

$$\text{Precision}(p) = \frac{\text{No. of people in C(p) with affiliation A(p)}}{\text{No. of people in C(p)}},$$

$$\text{Recall}(p) = \frac{\text{No. of people in C(p) with affiliation A(p)}}{\text{Total No. of people with affiliation A(p)}}.$$




Techniques


DOI 10.17148/IJARCCE.2015.41164

## 5.CONCLUSION

This paper presents an overview of fuzzy clustering algorithms that could be potentially suitable for document clustering based on user query, a new fuzzy c-means clustering algorithm implemented in the web documents environment, and an empirical comparison of hard c-means and fuzzy c-means as an application on web documents and 2D points.

Further work will consider: database queries connecting fuzzy c-means with web document and designing and implementing some adaptive threshold approach for converting fuzzy cluster to its crisp equivalent. This should be done in such a way that one document could be assigned to none, one or more clusters according to its membership degrees and similarities to the clusters. Furthermore we will perform statistical evaluation of hard c-means and fuzzy c-means in terms of document classification using other quality measures (besides average similarity) for generated clusters

## REFERENCES

[1] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu. A framework for clustering evolving data streams. In Proceedings of VLDB, pages 81–92, Berlin, Germany, September 2003.

[2] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In Proceedings of WSDM, pages 5–14, Barcelona, Spain, February 2009.

[3] S. Agrawal, S. Chaudhuri, G. Das, and A. Gionis. Automated ranking of database query results. In CIDR, 2003.

[4] S. Boriah, V. Chandola, and V. Kumar. Similarity measures for categorical data: A comparative evaluation. In Proceedings of SIAM International Conference on Data Mining (SDM 2008), pages 243–254, Atlanta, Georgia, USA, April 2008.

[5] G. Chatzopoulou, M. Eirinaki, and N. Polyzotis. Query recommendations for interactive database exploration. In Proceedings of SSDBM, pages 3–18, New Orleans, LA, USA, June 2009.

[6] S. Chaudhuri, G. Das, V. Hristidis, and G. Weikum. Probabilistic information retrieval approach for ranking of database query results. ACM Trans. Database Syst. (TODS), 31(3):1134– 1168, 2006.

[7] K. Chen, H. Chen, N. Conway, J. M. Hellerstein, and T. S. Parikh. Usher: Improving data quality with dynamic forms. In Proceedings of ICDE conference, pages 321–332, Long Beach, California, USA, March 2010.

[8] E. Chu, A. Baid, X. Chai, A. Doan, and J. F. Naughton. Combining keyword search and forms for ad hoc querying of databases. In Proceedings of ACM SIGMOD Conference, pages 349–360, Providence, Rhode Island, USA, June 2009.

[9] S. Cohen-Boulakia, O. Biton, S. Davidson, and C. Froidevaux.Bioguidesrs: querying multiple sources with a user-centric perspective. Bioinformatics, 23(10):1301–1303, 2007.


DOI 10.17148/IJARCCE.2015.41164