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

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 1, January 2015

Fuzzy C Means Algorithm for inferring User Search Goals with Feedback Sessions

N.Vidhyapriya[1], S.Sampath[2]

Department of Computer Science, P.K.R Arts College for Women Gobichettipalayam[1]

Associate Professor, Department of Computer Science, P.K.R Arts College for Women Gobichettipalayam[2]

Abstract: For a broad-topic and ambiguous query, different users may have different search goals when they submit it to a search engine. The inference and analysis of user search goals can be very useful in improving search engine relevance and user experience. The project proposes a novel approach to infer user search goals by analyzing search engine query logs. First, it proposes a framework to discover different user search goals for a query by clustering the proposed feedback sessions. The feedback session is defined as the series of both clicked and unclicked URLs and ends with the last URL that was clicked in a session from user click-through logs. Second, the pseudo-documents are produced to better represent the feedback sessions for clustering.

The pseudo-documents are clustered using Fuzzy C Means, the fuzzy similarity based self- constructing algorithm. A novel optimization method is used to map feedback sessions to pseudo-documents which can efficiently reflect user information needs and finally, a new criterion "Classified Average Precision (CAP)" is used to evaluate the performance of inferring user search goals. Experimental results are presented using user click-through logs from a commercial search engine to validate the effectiveness.

Keywords: User Search Goal, Feedback Session, Fuzzy C Means Algorithm

1. INTRODUCTION

1.1 Web Search Engine Concepts

In web search applications, queries are submitted to search engines to represent the information needs of users. However, sometimes queries may not exactly represent user's specific information needs since many ambiguous queries may cover a broad topic and different users may want to get information on different aspects when they submit the same query. For example, when the query "the sun" is submitted to a search engine, some users want to locate the homepage of a United Kingdom newspaper, while some others want to learn the natural knowledge of the sun.

Therefore, it is necessary and potential to capture different user search goals in information retrieval. Information need is a user's particular desire to obtain information to satisfy his/her need. User search goals can be considered as the clusters of information needs for a query. The inference and analysis of user search goals can have a lot of advantages in improving search engine relevance and user experience

1.2 Classes of Analyzing User Search Goals

Due to its usefulness, many works about user search goals analysis have been investigated.
They can be summarized into three classes:

- Query Classification
- Search Result Reorganization
- Session Boundary Detection

1.2.1 Query Classification

In query classification, people attempt to infer user goals and intents by predefining some specific classes and performing query classification accordingly. They consider user goals as "Navigational" and "Informational" and categorize queries into these two classes. The query intents are defined as "Product intent" and "Job intent" and they try to classify queries according to the defined intents. Other works focus on tagging queries with some predefined concepts to improve feature representation of queries.

1.2.2 Search Result Reorganization

In the search result reorganization class, people try to reorganize search results. They learn the interesting aspects of queries by analyzing the clicked URLs directly from user click-through logs to organize search results. However, this method has limitations since the number of different clicked URLs of a query may be small. Other works analyze the search results returned by the search engine when a query is submitted. Since user feedback is not considered, many noisy search results that are not clicked by any users may be analyzed as well.

1.2.3 Session Boundary Detection

In session boundary detection, people aim at detecting session boundaries. By, predicting goal and mission boundaries to hierarchically segment query logs. However, this method only identifies whether a pair of queries belongs to the same goal or mission and does not care what the goal is in detail.

1.3 Inferring User Search Goals

Inferring user search goals aims at discovering the number of diverse user search goals for a query and depicting each goal with some keywords automatically. It first proposes a novel approach to infer the user search goals for a query

DOI 10.17148/IJARCCE.2015.4179

by clustering the proposed feedback sessions. The feedback session is defined as the series of both clicked and unclicked URLs and ends with the last URL that was clicked in a session from user click-through logs.

A novel optimization method is then used to map feedback sessions to pseudo-documents which can efficiently reflect user information needs. At last, clustering these pseudo documents to infer user search goals and depicting them with some keywords. Since the evaluation of clustering is also an important problem, it also proposes a novel evaluation criterion called "classified average precision" (CAP) to evaluate the performance of the restructured web search results. The proposed evaluation criterion can also help us to optimize the parameter in the clustering method when inferring user search goals.

1.3.1 Advantages of inferring user search goals

First, restructure the web search results according to user search goals by grouping the search results with the same search goal; thus, users with different search goals can easily find what they want. Second, user search goals represented by some keywords can be utilized in query recommendation; thus, the suggested queries can help users to form their queries more precisely. Third, the distributions of user search goals can also be useful in applications such as reranking web search results that contain different user search goals.

2. RELATED WORK

2.1 Query Recommendation with TF-IQF Model and Popularity Factor

Query recommendation is a technique that provides better queries to help users to get the needed documents when the original query submitted by the user may be insufficient or imprecise to retrieve those. In this paper a novel method for query recommendation is proposed. It is different from traditional methods in two aspects: 1) it breaks URLs into independent tokens and uses a TF-IQF model to present the queries, and calculates the query. Similarity based on that model in further steps, while traditional query log related methods take the clicked URLs recorded in query log as whole; and 2) it introduces a query popularity factor. The popularity factor adds weight to the queries that receive more user clicks, with the assumption that the quality of these popular queries is proven by previous users. In the experiments based on real commercial search engine query logs, this method outperforms others, which demonstrates the effectiveness of the proposed TF-IQF model and popularity factor.

2.2 Search Engine Query Clustering using Top-K Search Results

Clustering of search engine queries has attracted significant attention in recent years. Many search engine applications such as query recommendation require query clustering as a pre-requisite to function properly. Indeed, clustering is necessary to unlock the true value of query logs. However, clustering search queries effectively is quite challenging, due to the high diversity and arbitrary input by users. Search queries are usually short and ambiguous in terms of user requirements. Existing prevalent clustering methods, such as K-Means or DBSCAN cannot assure good results in such a diverse environment.

Agglomerative clustering gives good results but is computationally quite expensive. This paper presents a novel clustering approach based on a key insight– search engine results might themselves be used to identify query similarity. It proposes a novel similarity metric for diverse queries based on the ranked URL results returned by a search engine for queries. This is used to develop a very efficient and accurate algorithm for clustering queries.

2.3 A Feature–Free Flexible Approach to Topical Classification of Queries Web

The task of topical classification of Web queries is to classify Web queries into a set of target categories. Machine learning based conventional approaches usually rely on external sources of information to obtain additional features for Web queries and training data for target categories. Unfortunately, these approaches are known to suffer from inability to adapt to different target categories which may be caused by the dynamic changes observed in both Web topic taxonomy and Web content. This paper proposes a feature-free flexible approach to topical classification of Web queries. It analyses queries and topical categories themselves and utilizes the number of Web pages containing both a query and a category to determine their similarity. The most attractive feature of this approach is that it only utilizes the Web page counts estimated by a search engine to provide the Web query classification with respectable accuracy.

2.4 How do they compare? Automatic Identification of Comparable Entities on the Web

People love comparing things: from home mortgages and digital cameras to travel destinations and political philosophies. Today, it is mostly limited to browsing documents after issuing comparative queries to Web search engines, such as“15-year vs. 30-year mortgage”, “Nikon D90 / Canon 40D”, “Oahu or Maui”, and “communism vs. fascism”. There is an opportunity to improve the search experience by automatically offering comparisons to users. This research work proposes a first step towards this goal of comparative analysis by mining a broad class of comparable entities from search query logs and a large Web crawl. Example comparable that includes medicines, appliances, electronics, vacation destinations, and many more. It presents an extensive empirical analysis showing that this method generate comparable with high precision and recall, and showing that Web search query logs are a superior source for mining such entities as compared to Web pages, typically used for extraction tasks. Further comparing the performance of this method with “related entities” reported by Google Sets, it shows a gain of 39% in average precision and a gain of 30% in NCDG.

DOI 10.17148/IJARCCE.2015.4179

2.5 Automatic Identification of User Goals in Web Search

There have been recent interests in studying the “goal” behind a user's Web query, so that this goal can be used to improve the quality of a search engine's results. Previous studies have mainly focused on using manual query-log investigation to identify Web query goals. In this paper the study reveals whether and how goal- identification process can be automated. It presents results from a human subject study that strongly indicates the feasibility of automatic query-goal identification. It then proposes two types of features for the goal-identification task: user-click behavior and anchor-link distribution.

2.6 Learn from Web Search Logs to Organize Search Results

Elective organization of search results is critical for improving the utility of any search engine. Clustering search results is an elective way to organize search results, which allows a user to navigate into relevant documents quickly.

However, two decencies of this approach make it not always work well: (1) the clusters discovered do not necessarily correspond to the interesting aspects of a topic from the user's perspective; and (2) the cluster labels generated are not informative enough to allow a user to identify the right cluster. This paper proposes to address these two decencies by (1) learning “interesting aspects” of a topic from Web search logs and organizing search results accordingly; and (2) generating more meaningful cluster labels using past query words entered by users.

2.7 Learning to Cluster Web Search Results

Organizing Web search results into clusters facilitates users' quick browsing through search results. Traditional clustering techniques are inadequate since they don't generate clusters with highly readable names. This paper renormalizes the clustering problem as a salient phrase ranking problem. Given a query and the ranked list of documents returned by a certain Web search engine, this method first extracts and ranks salient phrases as candidate cluster names, based on a regression model learned from human labeled training data. The documents are assigned to relevant salient phrases to form candidate clusters, and the final clusters are generated by merging these candidate clusters.

3. SYSTEM METHODOLOGY

3.1 Representation of Feedback Sessions

Inferring different user search goals for a query can be easily obtained by clustering feedback sessions. Clustering feedback sessions is more efficient than clustering search results or clicked URLs directly. The distributions of different user search goals can be obtained conveniently when the feedback sessions are clustered.

3.1.1 Feedback Sessions

Generally, a session for web search is a series of successive queries to satisfy a single information need and some clicked search results. The main focus is 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 research 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.

Inside the feedback session, the clicked URLs tell what users require and the unclicked URLs reflect what users do not care about. It should be noted that the unclicked URLs after the last clicked URL should not be included into the feedback sessions since it is not certain whether they were scanned or not.Each feedback session can tell what a user requires and what he/she does not care about. Moreover, there are plenty of diverse feedback sessions in user click-through logs. Therefore, for inferring user search goals, it is more efficient to analyze the feedback sessions than to analyze the search results or clicked URLs directly.

3.2 Mapping Feedback Sessions to Pseudo-Documents

The 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. However, since different feedback sessions have different numbers of URLs, the binary vectors of different feedback sessions may have different dimensions. Moreover, binary vector representation is not informative enough to tell the contents of user search goals. Therefore, it is improper to use methods such as the binary vectors and new methods are needed to represent feedback sessions.

For a query, users will usually have some vague keywords representing their interests in their minds. They use these keywords to determine whether a document can satisfy their needs. These keywords are named as “goal texts”. However, although goal texts can reflect user information needs, they are latent and not expressed explicitly. Therefore, pseudo-documents are introduced as surrogates to approximate goal texts. Thus, pseudo-documents can be used to infer user search goals. This research work proposes a novel way to map feedback sessions to pseudo-documents and it includes two steps.

- Representing the URLs in the feedback session

In the first step, the URLs are enriched with additional textual contents by extracting the titles and snippets of the returned URLs appearing in the feedback session. Thus,

DOI 10.17148/IJARCCE.2015.4179

the each URL in a feedback session is represented by a small text paragraph that consists of its title and snippet. Then, some textual processes are implemented to those text paragraphs, such as transforming all the letters to lowercases, stemming and removing stop words. Finally, each URL's title and snippet are represented by a Term Frequency-Inverse Document Frequency (TF-IDF) vector [1], respectively, as in

$$Tui = [tw1, tw2,...., twn]T, \qquad (1)$$
$$Sui = [sw1, sw2,...,swn]T,$$

where Tui and Sui are the TF-IDF vectors of the URL's title and snippet, respectively. ui means the ith URL in the feedback session. And wj (j= 1; 2; . . . ; n) is the jth term appearing in the enriched URLs. Here, a "term" is defined as a word or a number in the dictionary of document collections. twj and swj represent the TF-IDF value of the jth term in the URL's title and snippet, respectively. Considering that URLs' titles and snippets have different significances, representing the enriched URL by the weighted sum of Tui and Sui, namely

$$Fui = wtTui + wsSui = [fw1 ,fw2, ..,fwn]T , \quad (2)$$

where Fui means the feature representation of the ith URL in the feedback session, and wt and ws are the weights of the titles and the snippets, respectively. Set ws to be 1 at first. Then, stipulate that the titles should be more significant and the weight of the titles should be higher than the snippets.

- Forming pseudo-document based on URL representations

In order to obtain the feature representation of a feedback session, the optimization method is used to combine both clicked and unclicked URLs in the feedback session. Let Ffs be the feature representation of a feedback session and ffs (w) be the value for the term w.

Let Fucm (m = 1; 2, . . ,M) and Fucl (l = 1, 2, . . . , L) be the feature representations of the clicked and unclicked URLs in this feedback session, respectively. To obtain such an Ffs that the sum of the distances between Ffs and each Fucm is minimized and the sum of the distances between Ffs and each Fucl is maximized. Based on the assumption that the terms in the vectors are independent, perform optimization on each dimension independently. Up to now, the feedback session is represented by Ffs. Each dimension of Ffs indicates the importance of a term in this feedback session. Ffs is the pseudo-document. It reflects what users desire and what they do not care about. It can be used to approximate the goal texts in user mind.

3.3 Inferring User Search Goals by Clustering Pseudo-Documents

With the proposed pseudo-documents, the user search goals can be inferred and depicted with some meaningful keywords. Each feedback session is represented by a pseudo-document and the feature representation of the pseudo-document is Ffs. The similarity between two pseudo-documents is computed as the cosine score of Ffsi and Ffsj , as follows:

Sim i,j = cos (Ffsi , Ffsj)
= Ffsi . Ffsj
|Ffsi||Ffsj|

And the distance between two feedback sessions is

Disi,j = 1- Simi,j

Then the Fuzzy C Means Algorithm is used to cluster the pseudo documents.

The Fuzzy C-Means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This algorithm works by assigning membership to each data point corresponding to each cluster centre on the basis of distance between the cluster and the data point. More the data is near to the cluster centre more is its membership towards the particular cluster centre. Clearly, summation of membership of each data point should be equal to one.

The algorithm is based on minimization of the following objective function

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m}\left\|x_i - c_j\right\|^2 , \quad 1 \le m < \infty$$

where m (the Fuzziness Exponent) is any real number greater than 1, N is the number of data, C is the number of clusters, uij is the degree of membership of xi in the cluster j, xi is the ith of d-dimensional measured data, cj is the d-dimension centre of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the centre. After clustering all the pseudo-documents, each cluster can be considered as one user search goal. The centre point of a cluster is computed as the average of the vectors of all the pseudo-documents in the cluster.

Finally, the terms with the highest values in the centre points are used as the keywords to depict user search goals. Note that an additional advantage of using this keyword based description is that the extracted keywords can also be utilized to form a more meaningful query in query recommendation and thus can represent user information needs more effectively. Moreover, the number of the feedback sessions in each cluster is the useful distributions of user search goals that can be obtained simultaneously. The ratio of the number of the feedback sessions in one cluster and the total number of all the feedback sessions is the distribution of the corresponding user search goal.

3.4 Evaluation based on Restructuring Search Results

It is necessary to develop a metric to evaluate the performance of user search goal inference objectively. Considering that if user search goals are inferred properly, the search results can also be restructured properly, since restructuring web search results is one application of inferring user search goals. Therefore, the method based

DOI 10.17148/IJARCCE.2015.4179

on restructuring web search results to evaluate whether user search goals are inferred properly or not is proposed. In this section, the novel criterion "Classified Average Precision" is used to evaluate the restructured results. Based on the proposed criterion, the method selects the best cluster number.

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. The inferred user search goals are represented by the vectors and the feature representation of each URL in the search results can be computed. Then, categorize each URL into a cluster centred by the inferred search goals. Categorization is done by choosing the smallest distance between the URL vector and user-search-goal vectors. Thus, the search results can be restructured according to the inferred user search goals.

3.5 Evaluation Criterion: Classified Average Precision

In order to apply the evaluation method to large-scale data, the single sessions in user click-through logs are used. Because from user click-through logs, implicit relevance feedbacks can be obtained, namely "clicked" means relevant and "unclicked" means irrelevant. A possible evaluation criterion is the average precision (AP) 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. However, AP is not suitable for evaluating the restructured or clustered searching results. The proposed new criterion "Voted AP (VAP)" which is the AP of the class including more clicks namely votes. Considering an extreme case, if each URL in the click session is categorized into one class, VAP will always be the highest value namely 1 no matter whether users have so many search goals or not.

Therefore VAP is still an unsatisfactory criterion, so there should be a risk to avoid classifying search results into too many classes by error. The Risk calculates the normalized number of clicked URL pairs that are not in the same class. Then the new evaluation criterion Classified Average Precision (CAP) is proposed by using Voted AP and Risk.

$$CAP = VAP \times (1 - Risk)\,\gamma$$

CAP selects the AP of the class that user is interested in (i.e., with the most clicks/votes) and takes the risk of wrong classification into account. It is used to adjust the influence of Risk on CAP, which can be learned from the training data. Finally, utilize the CAP to evaluate the performance of restructuring search results.

4. ALGORITHM DETAILS

4.1 A fuzzy self-constructing algorithm (Data Mining Process)

Feature clustering is a powerful method to reduce the dimensionality of feature vectors for text classification. A fuzzy similarity-based self-constructing algorithm is used for feature clustering. The words in the feature vector of a document set are grouped into clusters, based on similarity test. Words that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation.
When all the words have been fed in, a desired number of clusters are formed automatically. Then there will be one extracted feature for each cluster. The extracted feature, corresponding to a cluster, is a weighted combination of the words contained in the cluster. By this algorithm; the derived membership functions match closely with and describe properly the real distribution of the training data. Besides, the user need not specify the number of extracted features in advance, and trial-and-error for determining the appropriate number of extracted features can then be avoided. Experimental results show that this method can run faster and obtain better extracted features than other methods.
Fuzzy clustering is a class of algorithms for cluster analysis in which the allocation of data points to clusters is not "hard" (all-or-nothing) but "fuzzy" in the same sense as fuzzy logic.

4.2 Explanation of clustering

Data clustering is the process of dividing data elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible. Depending on the nature of the data and the purpose for which clustering is being used, different measures of similarity may be used to place items into classes, where the similarity measure controls how the clusters are formed. Some examples of measures that can be used as in clustering include distance, connectivity, and intensity.

In hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster. In fuzzy clustering (also referred to as soft clustering), data elements can belong to more than one cluster, and associated with each element is a set of membership levels. These indicate the strength of the association between that data element and a particular cluster. Fuzzy clustering is a process of assigning these membership levels, and then using them to assign data elements to one or more clusters.

One of the most widely used fuzzy clustering algorithms is the Fuzzy C-Means (FCM) Algorithm. The FCM algorithm attempts to partition a finite collection of n elements into a collection of c fuzzy clusters with respect to some given criterion. Given a finite set of data, the algorithm returns a list of c cluster centres and a partition matrix, where each element wij tells the degree to which element xi belongs to cluster cj.

The standard function is:

$$w_k(x) = \frac{1}{\sum_j \left(\frac{d(\mathrm{center}_k, x)}{d(\mathrm{center}_j, x)}\right)^{2/(m-1)}}.$$

DOI 10.17148/IJARCCE.2015.4179

which differs from the k-means objective function by the addition of the membership values uij and the fuzzifier m. The fuzzifier m determines the level of cluster fuzziness. A large m results in smaller membership's wij and hence, the fuzzier clusters. In the limit m = 1, the memberships wij converge to 0 or 1, which implies a crisp partitioning. In the absence of experimentation or domain knowledge, m is commonly set to 2. The basic FCM Algorithm, given n data points (x1, . . . , xn) to be clustered, a number of c clusters with (c1, . ., cc) the centre of the clusters, and m the level of cluster fuzziness.

4.3 Fuzzy C-Means Clustering

In fuzzy clustering, every point has a degree belonging to clusters, as in fuzzy logic, rather than belonging completely too just one cluster. Thus, points on the edge of a cluster may be in the cluster to a lesser degree than points in the centre of cluster. An overview and comparison of different fuzzy clustering algorithms is available.

Any point x has a set of coefficients giving the degree of being in the kth cluster wk(x). With fuzzy c-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster. The degree of belonging, wk(x), is related inversely to the distance from x to the cluster centre as calculated on the previous pass. It also depends on a parameter m that controls how much weight is given to the closest centre.

The Fuzzy C-Means Algorithm:

i. Choose a number of clusters.
ii. Assign randomly to each point coefficients for being in the clusters.
iii. Repeat until the algorithm has converged (that is, the coefficients' change between two iterations is no more than, the given sensitivity threshold)
iv. Compute the centroid for each cluster, using the formula above.
v. For each point, compute its coefficients of being in the clusters, using the formula above.

The algorithm minimizes intra-cluster variance, and the results depend on the initial choice of weights. Using a mixture of Gaussians along with the expectation-maximization algorithm is a more statistically formalized method which includes some of these ideas: partial membership in classes.

5. CONCLUSION AND FUTURE WORK

This research work proposes a novel approach to infer user search goals for a query by clustering its feedback sessions represented by pseudo documents. First, the feedback sessions are analysed to infer user search goals rather than search results or clicked URLs. Both the clicked URLs and the unclicked ones before the last click are considered as user implicit feedbacks and taken into account to construct feedback sessions. Therefore, feedback sessions can reflect user information needs more efficiently. Second, the feedback sessions are mapped to pseudo documents to approximate goal texts in user minds. The pseudo documents can enrich the URLs with additional textual contents including the titles and snippets.

Based on these pseudo documents, user search goals can then be discovered and depicted with some keywords. Finally, a new criterion CAP is formulated to evaluate the performance of user search goal inference. Experimental results on user click through logs from a commercial search engine demonstrate the effectiveness of the proposed methods.

The complexity of the approach is low and can be used in reality easily. For each query, the running time depends on the number of feedback sessions. However, the dimension of *Ffs* is not very high. Therefore, the running time is usually short. In reality, the approach can discover user search goals for some popular queries offline at first. Then, when users submit one of the queries, the search engine can return the results that are categorized into different groups according to user search goals online. Thus, users can find what they want conveniently. The work mainly focused in comparative performance using different combinations of variables (e.g., click through logs vs. unclicked URL's) than in the comparative performance of different techniques.

However, better understanding the most useful models is an important direction both practically and theoretically. The system can be enhanced with new approach to produce feedback sessions from user click through logs. The performance can be improved by using another optimization method instead of binary vector representation and pseudo documents. For clustering the algorithm such as Soft K means or any other Fuzzy based algorithm can be used. Another direction would be to explore different evaluation methods to restructure web results.

6. RESULTS AND ANALYSIS

The work infers user search goals for a query by clustering its feedback sessions. User search goals are represented by the centre points of different clusters. Since each dimension of the feature vector of a centre point indicates the importance of the corresponding term, the keywords with the highest values in the feature vector is used to depict the content of one user search goal. This method demonstrates that when inferring user search goals, clustering the proposed feedback sessions are more efficient than clustering search results and clicked URLs directly. The performance evaluation and comparison are based on the restructuring web search results. If the search results of a query are restructured properly, Risk should be small and VAP should be high and the point should tend to be at the top left corner.

In this section, some intuitive explanation showing why clustering feedback sessions namely pseudo documents is better than the other methods when inferring user search goals is analysed. With the introduction of feedback

DOI 10.17148/IJARCCE.2015.4179

sessions, there will be a lot of advantages. Some advantages are summarized as follows:

1) Feedback sessions can be considered as a process of resampling. If the original URLs are viewed in the search results as original samples, then feedback sessions can be viewed as the "processed" or "resampled" samples which differ from the original samples and reflect user information needs. Without resampling, there could be many noisy URLs in the search results, which are seldom clicked by users. If the clustering is applied to the search results with these noisy ones, the performance of clustering will degrade greatly. However, feedback sessions actually "resample" the URLs and exclude those noisy ones.

2) Feedback session is also a meaningful combination of several URLs. Therefore, it can reflect user information need more precisely and there are plenty of feedback sessions to be analysed.

For example, in Fig. 6.1, the solid points represent the clicked URLs mapped into a 2D space and suppose that the users have two search goals: the star points belong to one goal and the circle points belong to the other goal. The large ellipse in Fig. 6.2 represents a feedback session which is the combination of several clicked URLs.

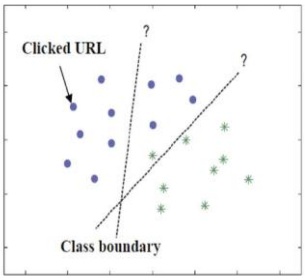

Figure *6.1*

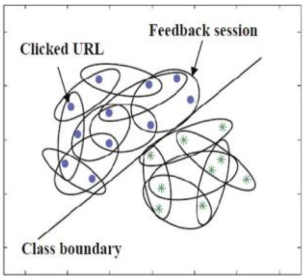

Figure *6.2*

The main result of our experiments is that the FCM algorithm has the advantage of robustness for ambiguity and maintains much more information than the traditional k-means algorithm. For a better understanding, consider the simple mono-dimensional example. Given a certain data set, suppose to represent it as distributed on an axis. The figure below shows this:

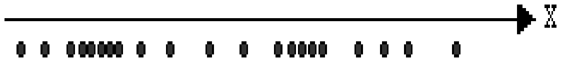

Figure 6.3 Dataset on an Axis

Looking at the picture, identify two clusters in proximity of the two data concentrations. Refer them using 'A' and 'B'. In the first approach shown is the k-means algorithm - associated each datum to a specific centroid; therefore, this membership function looked like this:

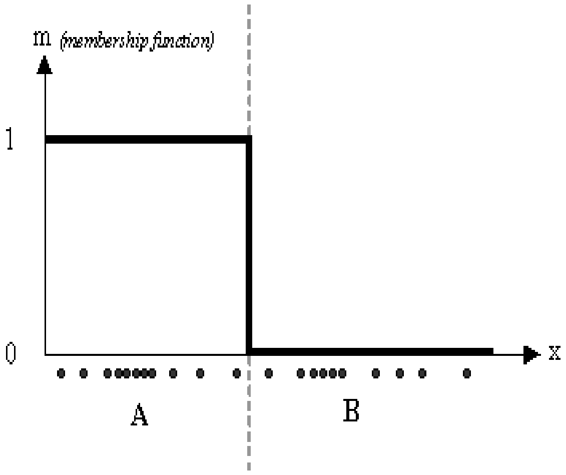

Figure 6.4 Dataset Associated with K-Means Algorithm

In the FCM approach, instead, the same given datum does not belong exclusively to a well defined cluster, but it can be placed in a middle way. In this case, the membership function follows a smoother line to indicate that every datum may belong to several clusters with different values of the membership coefficient.

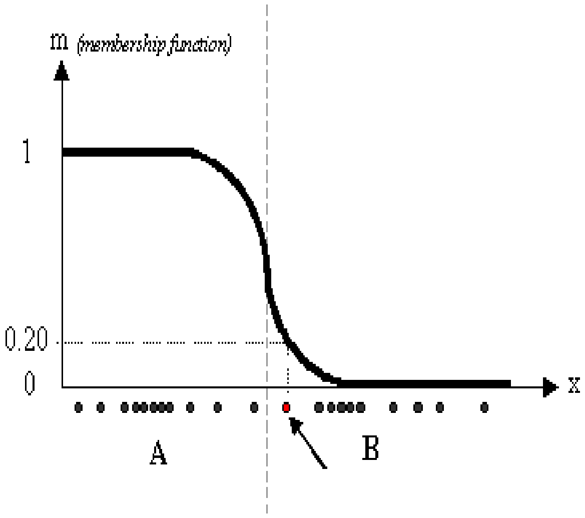

Figure 6.5 Dataset Associated with K-Means Algorithm

In the figure above, the datum shown with arrow spot belongs more to the B cluster rather than the A cluster. The value 0.2 of 'm' indicates the degree of membership to A for such datum. The conclusion can be made that FCM algorithm converge faster than the K-means algorithm.

REFERENCES

1. Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka, "A Web Search Engine-Based Approach to Measure Semantic Similarity between Words", IEEE Transactions on Knowledge and Data Engineering, Vol.23,No.7, July 2011.
2. Yuzhong Qu and Gong Cheng, "Falcons Concept Search: A Practical Search Engine for Web Ontologies", IEEE Transactions on Systems, Man and Cybernetics- Part A: Systems and Humans, Vil.41, No.4, July 2011.

3. John B.Killoran, “How to Use Search Engine Optimization Techniques to Increase Website Visibility”, IEEE Transactions on Professional Communication, Vol.56, No.1, March 2013.
4. Peng-Yeng Yin, Bir Bhanu, Fellow, IEEE, Kuang-Cheng Chang, and Anlei Dong, “Long-Term Cross-Session Relevance Feedback Using Virtual Features”, IEEE Transactions on Knowledge and Data Engineering, Vol.20, No.3, March 2008.
5. Yinghua Lu, TinghuaiMa, Changhong Yin, Xiaoyu Xie, Wei Tian and ShuiMing Zhong, “Implementation of the Fuzzy C-Means Clustering Algorithm in Meteorological Data”, International Journal of Database Theory and Application, Vol.6, No.6, 2013
6. V.P.Sai Suneetha, K.Chandrababu, “Clustering Principles for Inferring User Search Goals with Feedback Sessions”, Proceedings of Twelveth IRF International Conference, 31st August 2014.
7. Sumit Goswami and Mayank Singh Shishodia, “A Fuzzy Based Approach to Text Mining and Document Clustering”.
8. Vishal Gupta, Gurpreet S. Lehal; “A Survey of Text Mining Techniques and Applications”; Journal of Emerging Technologies in Web Intelligence, Vol.1, No.1, August 2009
9. K.Sathiyakumari, V.Preamsudha, G.Manimekalai; “Unsupervised Approach for Document Clustering Using Modified Fuzzy C mean Algorithm”; International Journal of Computer & Organization Trends –Volume 11 Issue3-2011.
10. L.Suganya, Dr.B.Srinivasan, “Effcient Semantic Similarity Based Fcm For Inferring User Search Goals with Feedback Sessions”, International Journal of Computer Trends and Technology (IJCTT), Vol.4, No.9, Sep 2013.

DOI 10.17148/IJARCCE.2015.4179