MODELING SMARTY WEB SEARCH ENGINE USING XML CLUSTERING

1R.Pratheeba, 2R. Purushothaman

1PG Scholar, Dept of CSE, G.K.M. College of Engineering and Technology
2Assistant Professor, Dept of CSE, G.K.M. College of Engineering and Technology

Abstract: Web search engine are today the initial page for most internet users. Searching becomes tedious, since the user provides different keywords to the search engine until they land up with best results. Feature selection used to identify the subset of the most useful feature. Feature clustering is a powerful alternative to feature selection for reducing the dimensionality of dataset. XML clustering formations used to achieve space and language competency. Instead of showing the results one by one, aim to group/cluster the results, so that user selects the group if interested which would reduce the result categories. The FAST algorithm is used to divide the feature into cluster by using clustering methods and they are strongly related to target classes from each cluster to form a subset of features. Data owner can upload the documents from any database format so that it is converted into xml format.

Keywords: Fast algorithm, Feature clustering, XML clustering.


1 Introduction

With the aim of choosing a subset of good features with respect to the target concepts, Feature subset selection is an effective way for reducing dimensionality, removing irrelevant data, increasing learning accuracy and improving result comprehensibility. Many feature subset selection methods have been proposed and studied for machine learning applications. They can be divided into four broad categories: the Embedded, Wrapper, Filter and Hybrid approaches.

XML Keyword search has attracted considerable attentions from research community recently. However, results returned by many XML search engines are still far from satisfactory from user’s perspective. The large number of answers hinders users from identifying relevant query results easily.

Traditional machine language algorithms like decision trees or artificial neural networks are examples of embedded approaches. The wrapper methods use the predictive accuracy of predetermined learning algorithms to determine the goodness of the selected subsets, the accuracy of the learning algorithms is usually high. The filter methods are independent of learning algorithms, with good generality. The wrapper methods are computationally expensive and tend to over fit on small training sets. The filter methods, in addition to their generality, are usually a good choice when the number of features is very large. Based on the MST method; we propose a FAST clustering based feature selection algorithm (FAST).

The hybrid methods are a combination of filter and wrapper methods by using a filter method to reduce search space that will be considered by the subsequent wrapper. Search results clustering have been proven effective in many retrieval tasks in web search field. Mixed results for ambiguous queries. Keyword queries are usually ambiguous and XML search engine are typically unable to disambiguate semantically different answers. These problems can be approached from different perspective.

Efficient Feature Selection via Analysis of Relevance and Redundancy

Feature selection is applied to reduce the number of features in many applications where data has hundreds or thousands of features. Existing feature selection methods mainly focus on finding relevant features. In this paper, we show that feature relevance alone is insufficient for efficient feature selection of high-dimensional data. We define feature redundancy and propose to perform explicit redundancy analysis in feature selection. A new framework is introduced that decouples relevance analysis and redundancy analysis. We develop a correlation-based method for relevance and redundancy analysis, and conduct an empirical study of its efficiency and effectiveness comparing with representative methods.
Modeling Smarty Web Search Engine Using XML Clustering

II. Problem Definition

Web search engines are today the initial page for most internet users. Although very useful to internet users, they present their difficulties: each one has a separate user interface, each interprets queries in its own way, they all support different kinds of advance search functionalities, use different kinds of search algorithm and show different sets of results for the same search conditions (keywords). The search engines do not wish to possess multiple copies of the same internet web pages in their indexes. It takes up a lot of unnecessary room in their databases and slows down how significantly processing they have to do on a regular basis. Google, in particular, has been removing internet web pages from their search engine index that they deem to become a replicate.

In order for internet web pages to become crawled, indexed and ranked nicely within the search results, they need hyperlinks. Lately, the search engines have been relying much more on linkage data in order to determine the search engine rankings not only it is important for your internal navigation to become search engine friendly, it is important that your internet pages have hyperlinks from other web sites. Having links to your web pages assists the search engine spiders find your pages- and the much more links your web pages have the better. Many websites haven’t been optimized well. Usually do not have search engine friendly hyperlinks and they do not have several hyperlinks from other websites.

The search engine cannot display all features in a single page and the clustering formation cannot obtain. The search engine provides some sort of disadvantage on providing the results. The web page displays based on the ranking process. To overcome this, ranking process must be eliminated. The main challenge going to solve is to convert any database format into XML format in which the data owner uploads. The data which are in the XML format can be implemented in high performance. The XML database created does not need more space and the data will be stored in the particular order.

III. Existing Methodology

In the EXISTING METHODOLOGY, the web pages displays according to the number of click (i.e.) ranking process, such that the searching become very tedious. There is no clustering approach is implemented. The features are not clustered to display the results and the process of extracting the result is very tedious. The user will provide the keyword until they land up with the best results. Both irrelevant and redundant features are not handled properly in the existing system.

The disadvantage of this system is time consumption is high and the methods implemented to remove redundant features are failed since they are not effectively implemented.

In the, proposed methodolgy feature selection is avoided and feature clustering is achieved. Feature clustering is performed using FAST algorithm. FAST algorithm works in two steps: in the first step, features are divided into clusters by using graph-theoretic clustering methods. In the second
step, the most representative feature that is strongly related to target classes is selected from each cluster to form a subset of features.

XML clustering is a new approach implemented in this paper to cluster all types of features. This makes the searching process easy and quick. The advantage of using XML clustering is low time consuming and effective search is achieved based on feature search. The search engine created with XML will be time efficient.

IV. Modules

1. User query request
2. Clustering of features
3. XML database-server
4. Data retrieval of different formats

Clustering Of Features
Clustering is a data mining (machine learning) technique used to place data elements into related groups without advance knowledge of the group definitions. Popular clustering techniques include k-means clustering and expectation maximization (EM) clustering. Here the Clustering technique is used to group the data according to the user query. So that User Selects the Group if interested which would reduce the Result Categories. Also we Propose is Data Owner can Upload the Documents from any Database Format So that it is converted into XML Format.

Passive Clustering: A straightforward approach to implement the KMP-based clustering proceeds in three steps 1) obtain all answers of a given query; 2) derive KMPs from answers; 3) cluster search results based on the KMPs.

Active Clustering: The passive approach is a blocking approach, because clusters can be fulfilled only after results have been generated. In this approach, we infer the labels of clusters and then generate clusters using the labels. We call this approach as active approach.

Our approach is implemented in two steps:
Step1: Extract and encode structural information: This step scans the documents, computes their s-graphs, and encodes them in a data structure.
Step2: Perform clustering on the structural information: This step applies a suitable clustering algorithm on the encoded information to generate the clusters.
Initially, the s-graphs of all the documents are computed and stored in a structure called SG. An s-graph can be represented by a bit string which encodes the edges in the graph. In general, be small enough to fit into the memory. In the extreme case, a general approach such as sampling can be used. Clustering is performed on the bit strings. Therefore, we transform the problem of clustering XML documents into clustering a smaller set of bit strings, which is fast and scalable.

**Xml Database**

Server The dataset containing all the data will be uploaded to the xml database. They are created by XML language. When the keyword given it enters into the database and retrieves the data based on the keyword. When the keyword does not match, the result will not be provided. Clustering is performed for each and every format.

![Fig: 5 Example for clustering method](image)

Here in this module, the user’s query is processed by sending the request to the main data server. The main data server will contain the data it in it. The query is processed in the server, and the route is searched via the route. In the search process, the data is searched via the route, for example, the data searched is “Customers who are interested in Street Arts”, the Search Via is the route towards exact Key word. Here in this Query the Key word is the Street Art, so the Search engine has to route via the XML Database for Customer, then to their Interest, then to their Street Art Table. This process is called as Search Via.

![Fig: 5. XML Database](image)

Data Retrieval of Different Format Here the data of different formats which is clustered will be displayed for a particular keyword. Only the relevant information will be displayed. The server forwards the query to the XML database and searches each and every cluster formed using XML. The user can get the related information of different formats as a result from the search engine. When the query is given, it searches in the database and gathers all the related information about the keyword and forms the cluster. The image file, video file, PDF file, Document file, PowerPoint are extracted at the same time for the particular keyword given by the user.

**V. Xml Documents**

Semi structured data has not been a popular data format until the appearance of XML files, conventional clustering techniques do not have special emphasis on this data type. We can treat the elements of a document as attributes and convert the document into a transaction of binary attributes. Among the various other similarity measures, can be used to measure the similarity between documents. However, many structurally different documents have almost the same set of elements. In doc1 and doc2 have only one different element, but they should be in two different clusters according to the semantics, assuming that many applications would be interested in posting queries to journal and conference papers separately. In other words, doc2 and doc3 should be separated from doc1 to form a cluster.

![Fig:6 Representation of XML Documents](image)
VI Conclusion

In this paper, we investigate the problem of returning cluster-based search results for XML keyword search. We propose new answer semantics for XML keyword query, which is based on a proposed conceptually related relationship between nodes. Then, we propose a novel clustering methodology based on the notion of keywords matching pattern. 2) Constructing a minimum spanning tree from relative ones, and 3) partitioning the MST and selecting representative features. In the proposed algorithm, a cluster consists of features. Each cluster is treated as a single feature and thus dimensionality is drastically reduced. we present two approaches: the first one is a conventional one, which does clustering in a post phase; the second one is novel in that it performs clustering in an active way, i.e., it first computes KMPs, then generates clustered search results using the KMPs.

Generally, the proposed algorithm obtained the best proportion of selected features, the best runtime, and the best classification accuracy for Naive Bayes, C4.5, and RIPPER. The Win/Draw/Loss records confirmed the conclusions. We also found that FAST obtains the rank of 1 for microarray data, the rank of 2 for text data, and the rank of 3 for image data in terms of classification accuracy of the four different types of classifiers, and CFS is a good alternative. At the same time, FCBF is a good alternative for image and text data. Moreover, Consist, and FOCUS- SF are alternatives for text data. For the future work, we plan to explore different types of correlation measures, and study some formal properties of feature space. The generated clusters can be further improved by organizing clusters into a hierarchy. Experimental results verify the effectiveness and efficiency of our methods.

References