Using Fuzzy Clustering Powered by Weighted Feature Matrix to Establish Hidden Semantics in Web Documents

Dr.Pramod D Patil, Dr.Parag Kulkarni
Department of Computer Engineering
Dr. D.Y. Patil Institute of Technology
Pune, INDIA

***Abstract*—Digital Data is growing exponentially exploding on the 'World Wide Web'. The orthodox clustering algorithms obligate various challenges to tackle, of which the most often faced challenge is the uncertainty. Web documents have become heterogeneous and very complex. There exist multiple relations between one web document and others in the form of entrenched links. This can be imagined as a one to many (1-M) relationships, for example, a particular web document may fit in many cross domains viz. politics, sports, utilities, technology, music, weather forecasting, linked to ecommerce products, etc. Therefore, there is a necessity for efficient, effective and constructive context driven clustering methods. Orthodox or the already well-established clustering algorithms adhere to classify the given data sets as exclusive clusters. Signifies that we can clearly state whether to which cluster an object belongs to. But such a partition is not sufficient for representing in the real time. So, a fuzzy clustering method is presented to build clusters with indeterminate limits and allows that one object belongs to overlying clusters with some membership degree. In supplementary words, the crux of fuzzy clustering is to contemplate the fitting status to the clusters, as well as to cogitate to what degree the object belongs to the cluster. The aim of this study is to device a fuzzy clustering algorithm which along with the help of feature weighted matrix, increases the probability of multi-domain overlapping of web documents. Over-lapping in the sense that one document may fall into multiple domains. The use of features gives an option or a filter on the basis of which the data would be extracted through the document. Matrix allows us to compute a threshold value which in turn helps to calculate the clustering result.**

Keywords*—*Fuzzy; clustering; web document; feature matrix

I. Introduction

Let us now try to understand the need or motivation of the system. With an incredible circulation of several hundred million sites worldwide, the ever changing cluster of documents over the internet is getting bigger and bigger every day. This incorporates some very important and as well very difficult challenges. Over the preceding duration of ten years there has been incredible growth of data on World Wide Web. It has become a major source of information. Internet web generates the new defies of information retrieval [10] as the amount of data on web as well as the number of users using web growing rapidly. It is challenging to quest through this tremendously large catalogue for the information desired by user. Also the traditional clustering algorithms like the k-means, probabilistic algorithms, k-medoid, and density based algorithms, constraint based algorithms and hierarchical algorithms fail to generate a result which render or convey the cross linked relations between the web documents. The other most important aspect was the traditional clustering algorithms use the standard numpy arrays which are very slow and not so effective in time complexity wise processing. Also, these traditional clustering algorithms face the issue of ‘Concentration Measure’ or ‘Curse Dimensionality’. This was the motivation to propose a new algorithm using Weighted Matrix applying the Fuzzy Logic method. This would suffice the end user queries correctly. As explained earlier the amount of information on web is exponential and be termed as information burst, there is critical need to device the system that renders correct classification of data and should fetch correct result to the end user. Let us have a detail overview of components of our system and let us understand what operations it is designated to do.

II. Relevant Terms and Definitions

In this section, the relevant terms, tools, data mining process and techniques which are required for successful implementation of the experimental setup. Let us now start with the Web Crawlers. Search Engines use crawlers to collect data and then store it in database maintained at search engine side. For a given user's query the search engines searches in the local database and very quickly displays the results. The entire Knowledge Discovery System is shown in Fig. 1.

A. Web Crawlers

Web crawling is an imperative method for amassing data on, and custody up with, the speedily intensifying Internet. Web crawling can likewise be baptized as a graph search problem as web is considered to be a large graph where nodes are the pages and edges are the hyperlinks. Web crawlers can be used in various areas, the most prominent one is to index a large set of pages and permit other people to search this index. A Web crawler does not really move all over the place on the computers linked to the Internet, as viruses or bot agents do, as a substitute it only directs entreaties for documents on web servers from a set of already sites. However the web crawlers have progressed, there has remained significant weakness in search engines outstanding to the complex, inter related (linked or cross domain documents) in the document assembly. Polysemies, synonyms, homonyms, phrases, dependencies and spam's act as hindrance to the search engines and therefore hampering the results returned. Also the vagueness or irrelevance of the user probes increases the ambiguous results fetched [2].

B. *Pre-processing*

Data pre-processing as shown in Fig. 2 exists an often neglected step but very important and is of prime importance since data pre-processing forms the foundation step of additional analysis and dispensation of data. Data pre-processing involves following five steps:

1) *Data Cleaning*

This step has operations like to fill values which are missing, smoothen out the noisy data, detecting or eliminating outliers, and deciding discrepancies.

2) *Data Integration*

It involves integrating data using numerous databases, data cubes, or collections.

3) *Data Transformation*

In this step we perform normalization and aggregation operations on data which has been integrated from various data sources.

4) *Data Reduction*

In this step we condense the quantity of data and produce the similar investigative results.

5) *Data Discretization*

I this step of data preprocessing we perform discretization operations like replacing numerical attributes with nominal ones.

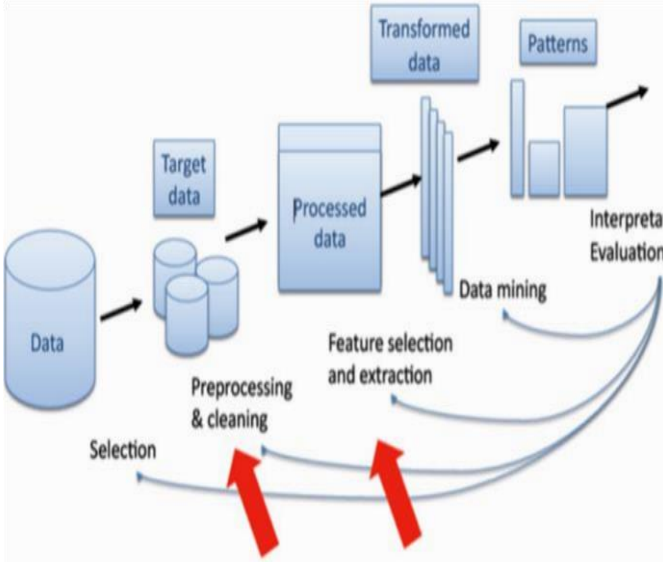


Fig. 1. Knowledge Discovery System [12]

The phrase – If you input the junk data that is 'Garbage In', then you be surely getting the junk output that refers to 'Garbage Out' is particular to the domain of machine learning [1]. Data congregation approaches are often range values, irregular, missing values. Analyzing data that hasn't been properly processed, such data can produce misleading results. Thus, pre-processing is primarily important step formerly running an investigation. Data fetched using a web crawler needs significant amount of processing before it is fed to the 'Fuzzy Clustering Algorithm' (FCA). Data in actual world is unclean which means it is incomplete. Incomplete data means it lacks attribute or the data in which we are interested in. The second part of dirty data is that it contains noise. Noisy data means that there are inaccuracies or outliers in it. The third part of dirty data is that it is inconsistent. Inconsistent means that the facts are not in correct format or the data lacks proper coding and naming format. If there is no good quality of data available then the data that would be eventually loaded in the data warehouse would be of low standards. The mining algorithms would yield a junk result out of the data warehouse. For data to be in correct format for data mining it should possess some valuable qualities where the data mined would be of highest quality [7].

These desirable qualities are precision, reliability, comprehensiveness, attribute value and most importantly timeliness. The most vital part of Preprocessing is cleaning the data. If the right data is not fed in we cannot expect the right output. Therefore cleansing of data is most vital. Missing data means computing the missing values. Adding missing values means filling the missing values with the average value derived by mean method. It also has a step of removing the noisy data. As discussed above noisy data means the data which comprises errors or outliers. Data cleaning also involves removing of inconsistencies. Inconsistent data removal means removing the data which falls into outlier range. Data can be collected from multiple formats like different databases, different file formats. The utmost vital part is to collate this data and then cleansing this data. It involves converting the dates to one particular format, converting numeric data to proper decimal format. It similarly involves performing binning on the numeric data. Filling the missing data is done by usually adding a tuple or replacing the missing data by a global constant. Applying the data cleansing task in my work the primary step is to remove the stop words from the data which has been fetched by the web crawler [4, 8]. Elimination of stop words and stemming [11]: In this phase, data which has less semantic is removed. Meaning, the full stops, commas, conjunctions etc. are removed. The data of the web documents fetched by the web crawler is equated with a bag of words – Stop words. The matched records are eliminated from the data file. Stemming process is a pre-processing step making the data ready for the next step. It is very important in most of the Information Retrieval systems. The main perseverance of stemming is to decrease diverse grammar pertaining forms or the words like its noun forms, adjective forms, verb forms, adverb forms etc. to its root form. The goal of stemming is to diminish deviational forms and occasionally derived various formations of a word to a conjoint base form. The available data is now further processed [5, 6].

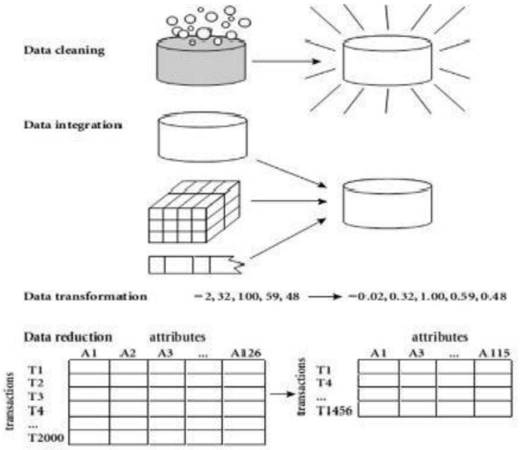


Fig. 2. Data Pre-Processing [12]

The next step of data pre-processing is integrating data from various databases, files, cubes together. Data integration means combining data into intelligible store. It also means schema integration. Schema integration means integrating metadata from different sources. It also includes identifying the attribute mismatch and performing a correcting action. This data is then fed to Data Reduction engine. In data reduction, the data which is redundant is removed. Data redundancy occurs because data is integrated from multiple databases, files etc. Meaning that same attribute might have been referred in a different way or the value of same attribute would have been derived or calculated. Redundant data is recognized by co relational analysis. Data reduction also involves performing numerosity reduction on data. It means to apply the linear regression model on data. This step is followed by Data Transformation. Data transformation means transforming the data in a format which is consistent throughout. It involves normalizing the data and aggregating it. The smoothening process of data transformation removes the noise from data. Aggregation step means aggregating the data into summarized cubes. Normalization activity means scaling the data such that it falls under particular range. It also includes construction of new attributes. It states that the data now has been fully transformed and ready to be loaded in the warehouse. We can conclude that data preparation is a critical issue for both data warehousing and data mining, as actual world data tends to be imperfect, noisy, and unpredictable. Data preparation involves data cleaning, data integration, data transformation, and data reduction. Data cleaning mechanism could be used to fill in missing values, lessen noisy data, detect outliers, and correct data inconsistency. Data integration loads data from multiples sources to form an intelligible data store. Metadata analysis, correlated data analysis, data skirmish detection, and the determination of semantic meanings add to smoothening the data. Data alteration techniques confirm the data into appropriate forms for mining. Data reduction methods such as dimension reduction data cube aggregation, numerosity reduction, data compression and discretization could be used to get a reduced depiction of the data, while minimizing the loss of information content. Concept hierarchies establish the attributes by the values or dimensions into measured levels of abstraction. They are methods of discretization that is predominantly useful in multilevel mining. For numeric data, practices such as data segmentation by divider documentations, histogram analysis, and clustering analysis can be used [9].

C. *Feature extraction*

Data mining is the cumulative task of data analysis and detection algorithms to perform automatic extraction of information from vast amounts of data. This process bonds many practical areas, counting databases, human computer interaction, statistical analysis, and machine learning. A typical data-mining chore is to forecast an unidentified value of circa attribute of a new occurrence when the values of the supplementary qualities of the new occurrence are recognized and a collection of instances with known values of all the attributes is given. Most importantly in numerous applications, data is the subject of analysis and dispensation in data mining, is multidimensional, and presented by a number of topographies. There are moreover many dimensions of data that it is relevant to several machine learning algorithms and denote the extreme raise of computational complexity as well as classification error with data having high expanse of dimensions. Hence, the dimensionality of the feature space is habitually abridged afore cataloguing is commenced [3]. Feature extraction is one of the dimension measures for lessening techniques. Feature extracts a subset of novel features from the unique feature set by means of some functional mapping possessing as much information in the data as possible. Many of the definite world applications has numerous features those are used in an effort to safeguard accurate cataloguing. If all those features are used for buildup classifiers, then they function in high dimensions, and the learning process becomes complex, which leads to high cataloguing error. Therefore, there is a necessity to condense the dimensionality of the features of data before classification. The key objective of dimensionality reduction as shown in Fig. 3 is to convert the high dimensional data samples into the space of low dimensions such that the core information contained in the data is preserved. Once the dimensionality is reduced, it aids us to improve the heftiness of the classifier [11, 22].

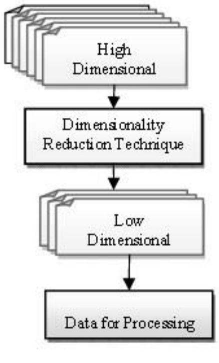


Fig. 3. Dimensionality Reduction Technique

Feature assortment is a technique to find good quality of germane features from the unique dataset using some data reduction and feature extraction measures. Feature extraction involves selection a feature, this is called as Feature Selection, Feature Selection step has turned out to be a thought provoking concern in the field of Pattern Recognition, Data

Mining, Machine Learning and Case Based Reasoning. Feature Selection is process of finding an ideal or suboptimal subset of 'n' features from the unique 'Features. It requires a large search space to get the feature subset. The ideal feature subset is analyzed by evaluation criteria. The key objective of the feature selection is to decrease the amount of features and to remove the irrelevant, redundant and noisy data. Feature Selection includes various steps. These steps are portrayed in a diagrammatic state as below in Fig. 4.

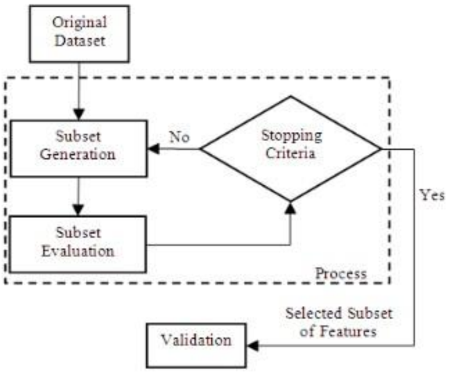


Fig. 4. Feature Extraction Engine

Feature selection mechanism is mostly classified into three types as shown in Fig. 5. They are, Filter Approach, Wrapper Approach and Hybrid Approach.

Feature selection method of 'Filtering an arithmetical measure used as a criterion for choosing the relevant features. This approach is calculated easily and very efficiently.

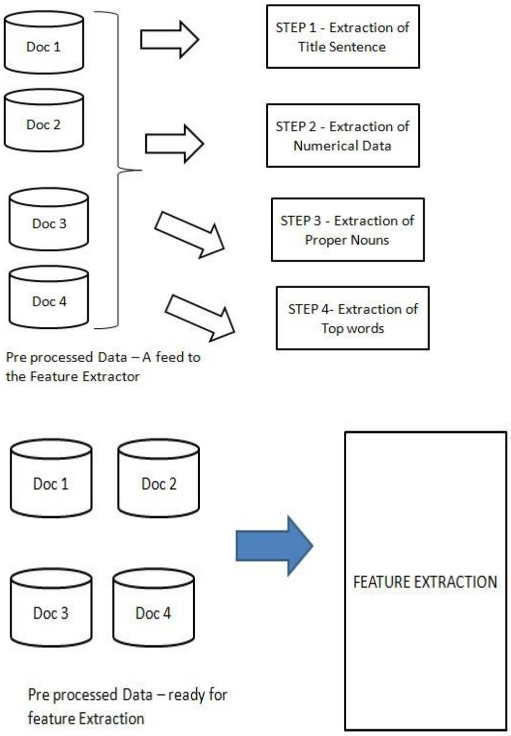

Doc 1
Doc 2
Doc 3
Doc 4
STEP 1 - Extraction of Title Sentence
STEP 2 - Extraction of Numerical Data
STEP 3 - Extraction of Proper Nouns
STEP 4- Extraction of Top words
Pre processed Data – A feed to the Feature Extractor
Doc 1
Doc 2
Doc 3
Doc 4
FEATURE EXTRACTION
Pre processed Data – ready for feature Extraction

Fig. 5. Feature Extraction Process implemented in our work - A high level overview

With the processed data now available, we now extract the important 'features' available. The first step in feature extraction process is to fetch the 'Title Sentence'. The first line of the document is rendered as the 'Title Sentence'. The second step is extracting the numerical data in the data file. A scan is performed and all the numerical data present in the data files is counted. Next, all the nouns present in all the data files are listed. Only proper nouns are to be used. The last scan is done for the 'top words'. Each document now is scanned. The word whose count is highest is regarded as top word. A list of such words is made in descendant order rendering to the web documents.

D. Fuzzy Logic

Fuzzy Logic System are those which produce satisfactory but definite output in rejoinder to imperfect, vague, partial, or imprecise (fuzzy) input. Fuzzy Logic is a technique of perception that it is similar or resembles anthropological reasoning. The methodology of Fuzzy Logic tries to inherit the way of conclusion making in humans that encompasses all transitional possibilities between digital values Yes and No. The predictable logic that a system can comprehend takes exact input and gives a certain output as true or false, which is corresponding to human's YES or NO. The creator of fuzzy logic term, Lotfi Zadeh, detected that dissimilar computers, the human conclusion making embraces a range of likelihoods between YES and NO, such as: CERTAINLY YES, POSSIBLY YES, CANNOT SAY, POSSIBLY NO, CERTAINLY NO [13, 21].

Fuzzy logic contains of four vital phases: A Fuzzfier, Rule Base mapper, An Inference Engine and Defuzzifier.

Fuzzy Logic Systems Architecture is as follows:

1) Fuzzification Module

This unit alters the input to the systems, which are in the form of crisp numbers, into fuzzy sets. For example it transmutes the supplied crisp values to a linguistic variable by making use of the membership functions warehoused in the fuzzy knowledge base. Fuzzy linguistic variable is used to epitomize qualities straddling a particular spectrum or cross domain.

2) Fuzzy Knowledge Base Module

It stocks the conditions established on the If and then rules provided by experts. The fuzzy knowledge base is constructed on linguistic and membership functions.

a) Linguistic Variables

Linguistic variables act as input or output for the system. Their values are articulated in a natural language as an alternate to numerical values. A linguistic variable exists as a generally disintegrated into a group of linguistic terms.

b) Membership Functions

It is used for 'quantifying' the linguistic term. Membership functions are used in the fuzzification and defuzzification phase to plot the non-fuzzy variable as input to fuzzy linguistic terms as well as the reverse way round.

3) Inference Engine Module

It feigns the human cerebral method by creating fuzzy interpretation on the inputs and IF-THEN rules.

4) Defuzzification Module

It transmutes the fuzzy variable set gained by the corollary engine to a definite value [20].

The exponential growth of the Web has led to extensive expansion of web content. The enormous area of product data on the internet poses inordinate task to both users and online commerce. More users are turning towards online shopping because it is relatively convenient, reliable, and fast; yet such users usually experience difficulty in probing for merchandises on the internet due to information overload. Online selling has often been stunned by the rich data they have collected and find it challenging to endorse merchandises suitable to precise users. There is also the problem of futile consumption of the available huge amount of merchandise data from online transactions to support better decision making by both consumers and suppliers. To discourse these information overload problems, e-learning, e-commerce, e-newspapers data stores are now smearing mass customization ideologies not to the merchandises but to their staging in the online.

Fuzzy Logic System as shown in Fig. 6 could be understood as a system which maps nonlinear data as an input to a scalar output data set. Fuzzy sets obligate powerful decision making ability and hence attracted rising consideration and curiosity in recent IT, data generation method, decision building, pattern acknowledgement, and diagnostics and data analysis among others. When a problem has vibrant or evolving behavior, fuzzy logic is a suitable contrivance that deals with such problem. In short to say, fuzzy logic has métier in providing precise solutions to problems

That encompasses the manipulation of numerous variables.

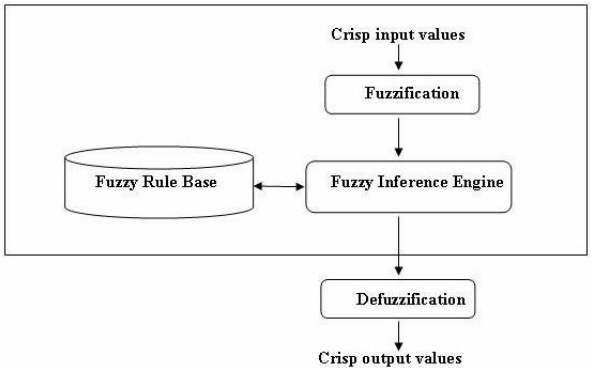


Fig. 6. The depiction of the Fuzzy Logic System [14]

The method of fuzzy logic systems are as follows:

a) Define input and output crisp variables.

b) Define the membership function.

c) (iii) Convert crisp input data into linguistic fuzzy values, using membership function, called fuzzification.

d) Evaluate the rules, using inference engine .

e) Construct the output crisp data, from fuzzy linguistic values, called defuzzification.

Fuzzy logic bids several unique features that make it a predominantly decent choice for many control problems.

a) It is inherently robust since it does not require precise, noise-free inputs and could be programmed to fail safely if a feedback sensor quits or is destroyed. The output regulator is a smooth control function notwithstanding an extensive assortment of input variations.

b) Since the Fuzzy logic checker processes the user-defined rules prevailing the target control system, it can be altered and tweaked easily to improve or radically alters system performance. New sensors can straightforwardly be fused into the system merely by engendering apt governing rules.

c) Fuzzy logic is not restricted to a few feedback inputs and control outputs, nor is it essential to measure or calculate rate-of-change restrictions in order to implement. This permits the sensors to be economical and imprecise thus keeping the inclusive system cost and intricacy low.

d) Due to the rule-based process, any equitable number of inputs can be administered and numerous outputs engendered.

e) Create Fuzzy logic membership functions that express the implication (values) of Input / Output relationships used in the rules.

Fuzzification is conversion of crisp variables into linguistic variables, and it is the central unit for fuzzy logic system. Variable pertaining to linguistic sense such as age might obligate a value such as 'young' or 'old'. However, the noteworthy efficacy of linguistic variables is that they can be amended via linguistic verges pragmatic to primary terms. Prof. Zadeh has recommended the notion of fuzzy variables. Although variables in arithmetic typically gross data which in the form of numbers, if the data which is not numeric then linguistic variables are often used to simplify the countenance of rules and facts. The usage of linguistic variables in numerous applications cuts the overall computation complexity of the application. Linguistic variables obligate to be predominantly useful in complex non-linear applications.

III. Literature Survey

Ojokoh et al. in their work in the paper titled *“A Fuzzy Logic Based Personalized Recommender System” [15]* communicates to apply Fuzzy logic algorithms to the e-commerce space to drill to exact customer requirement. They carried out the experiments using laptops of various brands and configurations that customers usually search on various e-commerce sites. It defines the Fuzzy near compactness concept is engaged to measure the resemblance between customer needs and merchandise features. The ever-increasing figure of E-retail, e-commerce websites on the internet has led to data overload with over hundreds and thousands of customers. So it is challenging for customers of certain merchandises to discover information regarding merchandises in an attempt to purchase products that best satisfies them. This has led in reduction of the amount of product sales in the e-commerce domain. The work in this paper highlights a personalized recommender system motivated by fuzzy logic method. The offered system intelligently mines data about the features of laptop computers and offers professional services to potential consumers by endorsing ideal merchandises grounded on their distinct requirements. They measured the result of the offered system by means of fifty laptop computers brands and configurations from Acer, HP, Sony, Dell and Toshiba. We studied the Fuzzy Logic implementation done in this paper. We also got to know how large data sets can help in an efficient fuzzy classification.

ChrisCornelius, Jie Lu et al. in their paper titled *"One and Only Item recommendation with Fuzzy Logic Techniques" [16]* implement a Collaborative Filtering method which is the abstract framework for endorsing one and-only items. It practices fuzzy logic, which permits to reflect the graded/uncertain data in the domain, and to range the CF paradigm, overcoming limitations of existing practices. The conceivable use of this Collaborative Filtering is in the e-government application. There is a personalization of e-government facilities intended at custom tailoring the content government made available to the end user. In several countries, e-government applications are increasing speedily and the quantity of e-government websites, as well as the assets and services provided, are dynamically increasing. This has caused a delinquent wherein citizens may find it more and tougher to locate relevant data from these websites. Matching specific citizens and businesses interests and needs is therefore one of the main trials for e-government services, and intelligent decision support. This paper gave us the idea of a fuzzy framework where a recommender system was constructed. It also gave an idea to construct a fuzzy algorithm which can be generically applied to all the cross domains.

Andreas Meier, KuisTeran in their work titled *"A Fuzzy Recommender System for eElections" [17]* describe the recommender system which is grounded on fuzzy logic and fuzzy clustering mechanism. It related to construction of an architecture for recommender system which can be used in e-Democracy and e-Elections applications. The use of this system enhances and succor voters in making verdicts by providing data about contenders close to the voter's preferences and tendencies. The usage of recommender systems for e-Government is used to decrease data overload, which might help to advance self-governing processes. Fuzzy clustering investigation differs from classic clustering where the interpretations belong to only one cluster. Moreover, classic clustering makes no use of plodding membership. The recommender system approach fluctuates from collaborative filtering. The later one is built on historical experiences. It is suitable in the one and only scenario where events such as voting and election processes occur only once. This paper was a crucial reference as it implemented a filtering based fuzzy clustering technique.

Tung-Cheng, T-zone-I Wang et al. in their work titled *"A Fuzzy Logic based Personalized Learning System" [18]* shed light on use of fuzzy logic clustering the e-Learning domain. It employs fuzzy insinuation mechanisms, reminiscence cycle updates, apprentice preferences and systematic hierarchy process. The system has been used to cram any language. By using fuzzy corollaries and personal reminiscence cycle updates, it is possible to find an editorial best suited for both a learner's ability and their need to review vocabulary. After reading an article, a test is instantaneously provided to enhance a learner's reminiscence for the words newly learned in the editorial. The methodology uses a questionnaire to realize a learner's predilections and then uses fuzzy inference to find editorial of suitable exertion levels for the learner. It then employs review values to compute the fraction of editorial vocabulary that the learner must evaluate. It has cartels these three parameters to establish the article's suitability formulae for computing the suitable level of articles for the learner. It uses memory to update the words so that the person seeking learning learns for the first time and also the words that appear that need to be reviewed based on the learner's learning feedback. The consequences of these experiments vitrine that with intensive reading of pupilages as recommended by the approach, student can reminisce together new words and the words learnt in past easily and for longer time, thus competently enlightening the vocabulary ability of the learner.

A research conducted by JieZang in the field of a *Social Media based Personalized Recommender system [19]* based on Fuzzy logic describes a recommender systems which are built on intelligent computational abilities. From the topical past with the rise of data balloon on the internet, there is a consistent demand for the data processing engine for solving the problem of information overloading and information filtering. Present-day recommender systems hitch context-awareness with the personalization to deal the most accurate endorsements about diverse merchandises, services, and possessions. However, such systems arise across the issues, such as cold start, sparsity, and scalability that lead to vague endorsements. Computational Intelligence means not only improve endorsement accuracy but also markedly mitigate the above-mentioned issues. Computational Intelligent system as based on practices, such as: (i) fuzzy sets (ii) Artificial Neural Networks (iii) Evolutionary Computing, (iv) Swarm Intelligence, (v) Artificial Immune Systems.

IV. Proposed and Implemented System

The main motivation was to propose an algorithm which uses the efficiency of the Matrix and weighted features with an application of fuzzy logic. This is a first ever attempt to create a fuzzy weighted matrix to extract the features of the data and then to form the overlapping logical clusters. Here in this section we are giving comprehensive emphasis on the design of the system. Each and every stage of the offered system is well narrated here. Along with the elucidation the complete system is well presented using the architecture. We are proposing a new algorithm using Fuzzy matrix and by using the weighted methods. The complete system as shown in Fig. 7 is dissected in four steps as discussed below.

The process proposed is: A web crawler would fetch number of web documents and store them in a folder. A web crawler would mainly act as a Data Source or Data Collector in our project work. The data fetched from the web crawlers is fed as an input to the pre-processing engine. The pre-processing engine follows the pre-designed steps of data cleansing and gives out a bag of words corresponding to the document as output. The pre-processing engine is devised with three algorithms. That means it is a three step process. The first step is to 'Remove the special symbols'. The second step is 'Removing the stop words' and the third step is 'Removing the Stemming and deriving the root form of the word'. The output of this step is then fed as input to the further module. This is then fed to the feature extractor. The Feature extractor extracts the features from the bag of words pertaining to web document. It extracts the semantic features such as Numerical Data, Nouns, Title Sentence and Highest occurring word. It is actually based on accepting the accuracy and number of top words constructs a matrix. A weighted feature matrix is build up and using fuzzy logic the overlapping structures of the web documents are revealed as a final output.

1) *Web Crawler*

A web crawler plays an important part in this project. It mainly acts as a Data Source / Data Collector. The web crawler would fetch in number of web documents and parse them using the open source Google parser and store them in a folder called as WEBPAGES_REPOSITORY. This data set of web documents is then used as an input to the Pre-Processing Engine.

2) *Pre-Processing*

Pre-processing is vital step in data mining systems as it condenses the scope of the data required for processing. This condensed size minimizes the cost and space complexity of the system as fewer quantities of data are needed to be processed.

We have devised three algorithms for Pro-processing module:

- Special Symbol Removal

A special symbol removal algorithm is devised. It scans the bag of words in the array list and removes the special symbols from it e.g.!,@,#,$,% etc. These special symbols do not contribute in result generation; hence it is worth to remove all the special symbols.

Algorithm for special symbol removal

```
Step 0: Start
Step 1: Read string
Step 2: divide string into words on space and store in a vector V
Step 3: Identify the duplicate words in the vector and remove them
Step 4: for i=0 to N (Where N is length of V)
Step 5: for i^th word of N check for its occurrence in Special Symbols repository
Step 6: if present then remove the special symbols
Step 7: else return the remaining words
Step 8: stop
```

- Stop Words Removal

Stop words are the words used as a supporting word in content to bring the semantics in the sentence; however, after this discarding the meaning of the sentence is not changing too much extent. Hence they are removed here by maintaining one repository for comparison. This repository contains the 500+ stop words.

Algorithm to find stop word

```
Step 0: Start
Step 1: Read string
Step 2: divide string into words on space and store in a vector V
Step 3: Identify the duplicate words in the vector and remove them
Step 4: for i=0 to N (Where N is length of V)
Step 5: for i^th word of N check for its occurrence in Stop words repository
Step 6: if present then remove the stop words
Step 7: else return remaining words
Step 8: stop
```

- Stemming

Stems are used to derive word. Generally the words are derived for making the correct use of tenses. Unnecessarily this stems increase the system costing hence they are removed over here. No stemming algorithm is there which gives 100% accuracy.

Algorithm for stemming

```
Step 0: Start
Step 1: Read string
Step 2: divide string into words on space and store in a vector V
Step 3: Identify the duplicate words in the vector and remove them
Step 4: for i=0 to N (Where N is length of V)
Step 5: for i^th word of N check for its occurrence in Stemming extensions repository
Step 6: if present then process the word back to its root form
Step 7: else do nothing
Step 8: stop
```

3) *Feature Extraction*

As data contains tons of features it's not worth to consider the complete content for the further operations. Feature extraction is essentials step in data mining. It is used for fetching the required data i.e. features from the huge set of data. In our proposed work four features are extracted.

- Title Sentence

Title sentences are the one which represents the first sentence of the file content. The reason behind this extraction is to give a proper name to the cluster because each cluster is named by the title sentences.

- Numeric Data

Numeric data plays vital role in file content as the most of the important data are represented using numerical values only. So by considering this thing we extracted numerical values from the file content.

- Proper nouns

Proper nouns are the words which represent the person or place. For extraction of this feature a dictionary is used. So to access this dictionary jxlapi offers all the necessary functionalities.

Algorithm to find noun

```
Step 0: Start
Step 1: Read string
Step 2: divide string into words on space and store in a vector V
Step 3: Identify the duplicate words in the vector and remove them
Step 4: for i=0 to N (Where N is length of V)
Step 5: for ith word of N check for its occurrence in Dictionary (Open source dictionary api used)
Step 6: if present then return true
Step 7: else return false
Step 8: stop
```

- Top Words

Top words are the important words of the sentence. Here in this feature the frequency of the each word are found out. The word which repeat more time is needed to consider as it have the more weightage in the file content.

Algorithm to find Term weight words

```
Step 0: Start
Step 1: Read string
Step 2: divide string into words on space and store in a vector V
Step 3: Identify the duplicate words in the vector and remove them
Step 4: for i=0 to N (Where N is length of V)
Step 5: for $i^{th}$ word of N check for its frequency
Step 6: Add frequency in List Called L
Step 7: end of for
Step 8: return L
Step 9: stop
```

4) *Master Matrix Creation*

Here in this step all the extracted features are taken as an input. From these entire features a one matrix is created. This is inspired from *'Vector Space Model', VSM* which is an algebraic model for representing documents. So, particular feature of each file is compared with the respective feature of the other file. In this way all the four features are compared with four features of other file. This comparison led to a score of each file with other file.

Matrix Creation Process:

A weighted matrix is built, feature values are calculated against the every document in following way as shown in Table I:

TABLE I. WEIGHTED FEATURE MATRIX CALCULATIONS (FEATURES EXTRACTED ARE: TOP WORDS, NUMBER DATA, PROPER NOUNS, TERM WEIGHT)

	Feature Extracted	**D 1**	**D 2**	**D 3**
		(T,N,P,Tw)	(T,N,P,Tw)	(T,N,P,Tw)
D 1	(T,N,P,Tw)	0		
D 2	(T,N,P,Tw)		0	
D 3	(T,N,P,Tw)			0
D n	(T,N,P,Tw)			

Fuzzy Logic

The generated score from matrix is taken as input. The smallest and biggest score is calculated. Exactly five ranges are calculated starting from smallest value and end to largest value. Now the score is assigned to each of the scores calculated in master matrix step by checking the occurrence of the score in these five ranges. Once score is calculate a threshold of 2 is set. The file having threshold more than 2 is added to cluster and discards the file which fails to satisfy the condition.

Algorithm for Document clustering using Fuzzy matrix Weighted method

```
Input: Merged Feature vectorFv
       User Accuracy as Ua
Output: Cluster Set C= {c1, c2, c3….cn}
Step 0: start
Step 1: create matrix M of length Fv
Step 1: For i=0 to Fv length (for each row)
Step 2: For j=0 to Fv length (for each column)
Step 3: Fvr= element of one row
Step 4: Fvc=element of one column
Step 5: Compare features and get score as Sc
Step 6: Average Score as Asc=Sc/4
Step 7: add Average to matrix M
Step 8: End Inner For
Step 9: End Outer For
Step 10: for every file in M'sRows if (Asc>=Ua) then add into cluster Ci
Step 11: return cluster set C
Step 12: Stop
```

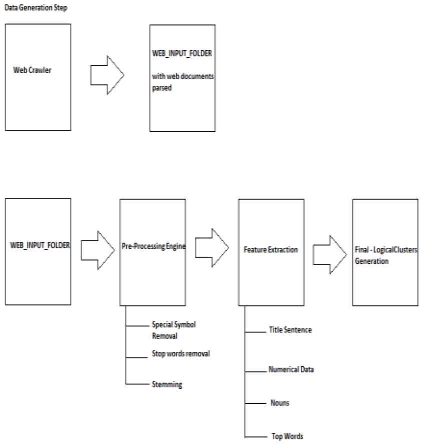


Fig. 7. Our Proposed System Architecture Overview

V. Experimental Results and Discussions

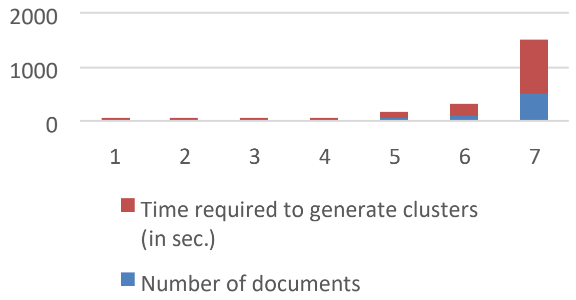
To show the efficiency of the system on experiment is conducted on java 1.6 based machine using Net beans as an IDE on windows machine having 2GB ROM and 500GB HDD. After doing the experiment by providing the files from different categories such as text, pdf, and doc the following observation is led.

We have considered the cross domain documents pertaining to various fields like Sports, medicine, finance, insurance, travel, music, etc. This data set was taken from the world renowned news channel which is an open source for data set and which is available to use for research work. The size of each document was about 2-3 MB text file. Furthermore we also provided the input as .mp3 files, .docx files, video files which were successfully handled and ignored by the system as currently we do not support these file types.

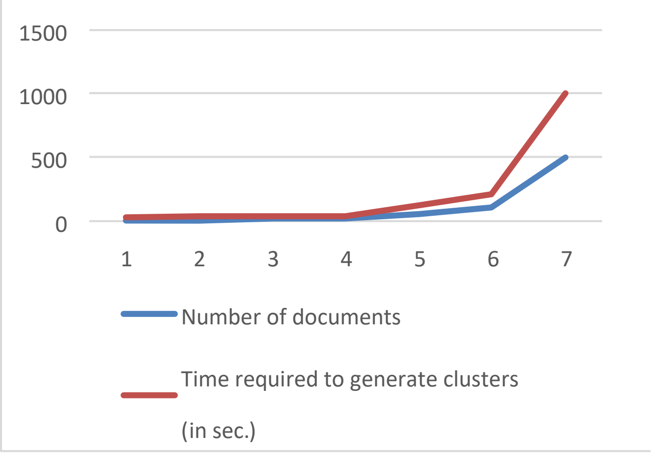
This system can be used as an overnight process, where you feed the system with huge volumes of documents and the system would successfully run and form the logical cluster groups of the documents. Logical clusters then would determine the relationships of various documents with each other.

TABLE II. Time Required To Cluster VS Numbers of Documents

Number documents of (Cross Domain)	Time (seconds) to generate overlapping clusters
5	30
10	34
15	41
20	47
50	117
100	209
500	1004



Graph 1: Performance measurement – Bar Graph representation of Number of Docs vs Time Required to Generate Clusters



Graph 2: Performance Measurement – Linear exponential depiction of Number of Documents to be clustered w.r.t Time Required

Graphs 1and 2 signifies the clustering time. From the graph we can determine that as the numbers of documents increase exponentially the required time to generate the clusters marginally increases in folds as shown in Table II.

Application screen shots and Comparative study

Compared the Fuzzy clustering by Weighted Feature Matrix algorithm with the traditional clustering algorithms.

A comparative study was conducted using the Dataset obtained from the BBC news website. The data set comprised of documents pertaining to various domains like 'Banks', 'Loans', 'Sports', 'Insurance', 'Weather', 'Politics', 'Music', 'Films', 'Geography', 'History', 'Literature' etc. We found that K-Means, Hierarchical algorithms do not perform well due to the inability to recognize the semantic meaning of the document. Therefore there was a need to propose a new algorithm which you carry out the clustering task as per the hidden semantics meaning. Thus we have tried our best to propose a new algorithm which extracts the features from the web documents and then follows the weighted matrix for generating the logical clusters. The implemented results are depicted from Fig. 8 to Fig. 18. The login screen shown in Fig. 8.

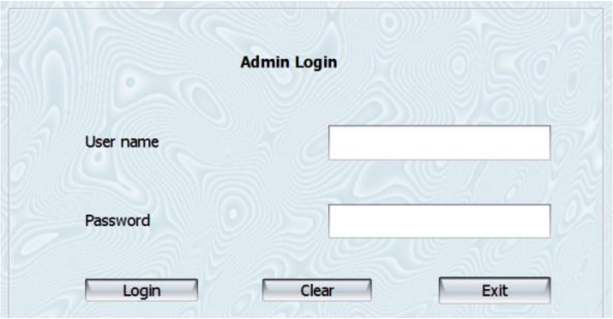
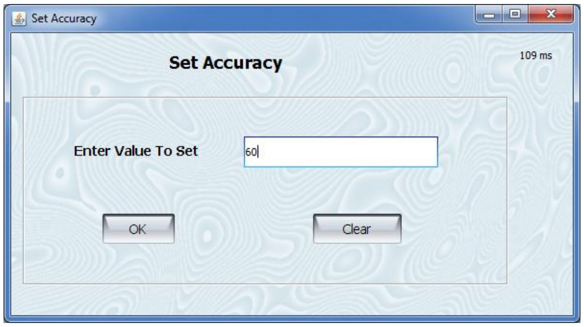


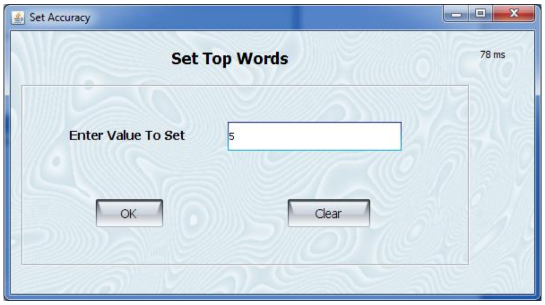
Fig. 8. Login Screen.

Now, Navigate to “System Settings”> Set “Accuracy”> Enter valid percentage from 0%-99% as shown in Fig. 9.



Enter Value To Set 60

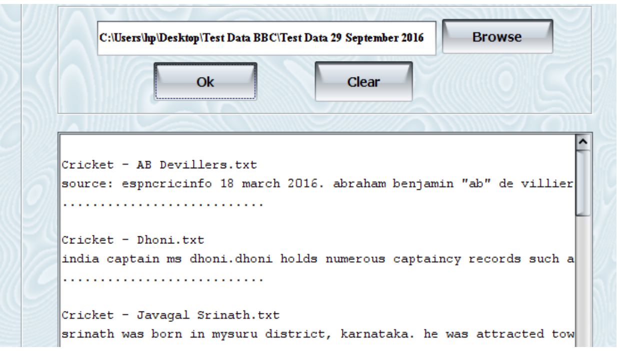
Fig. 9. Input Accuracy in %



Enter Value To Set 5

Fig. 10. Top Words

The process to set the value for Top words is depicted in Fig. 10. Now, navigate to “Folder Input”. Select the web pages repository which needs to be fed to our system as shown in Fig. 11.



C:\Users\hp\Desktop\Test Data BBC\Test Data 29 September 2016

```
Cricket - AB Devillers.txt
source: espncricinfo 18 march 2016. abraham benjamin "ab" de villier
..........................

Cricket - Dhoni.txt
india captain ms dhoni.dhoni holds numerous captaincy records such a
..........................

Cricket - Javagal Srinath.txt
srinath was born in mysuru district, karnataka. he was attracted tow
```

Fig. 11. Data Set Selection

Now, we will apply the “Pre-Processing” algorithms on the data imported from Web pages repository as shown in Fig. 12, 13 and 14.

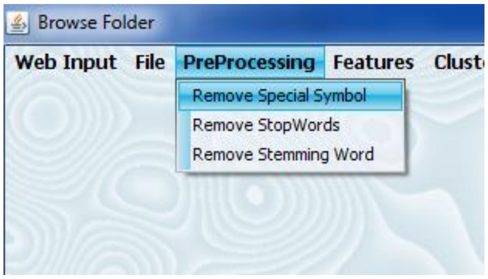


Fig. 12. Removal of special symbols

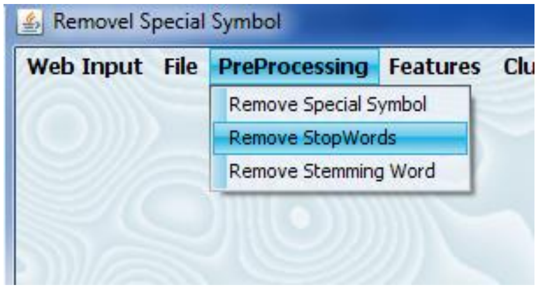


Fig. 13. Removal of stop words

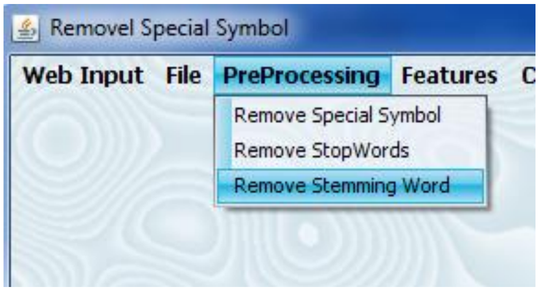
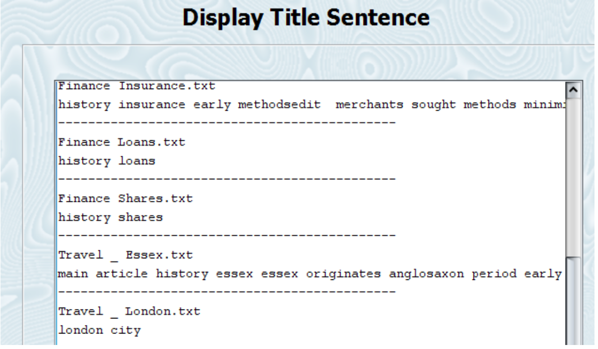


Fig. 14. Removal of stemming words

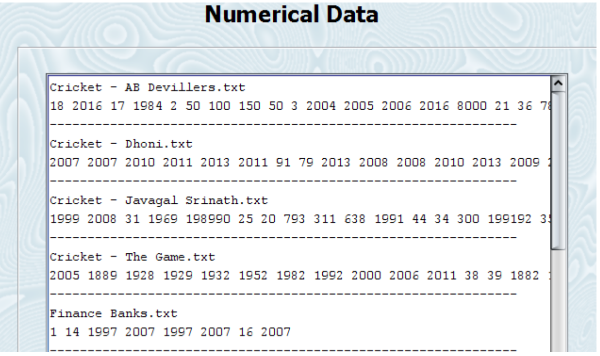
After the data has been pre-processed, we now feed this data to the “Feature Extractor engine”.

The following features are extracted in the given order as shown in Fig. 15, 16 and 17.



```
Finance Insurance.txt
history insurance early methodsedit  merchants sought methods minim
---------------------------------------------
Finance Loans.txt
history loans
---------------------------------------------
Finance Shares.txt
history shares
---------------------------------------------
Travel _ Essex.txt
main article history essex essex originates anglosaxon period early
---------------------------------------------
Travel _ London.txt
london city
```

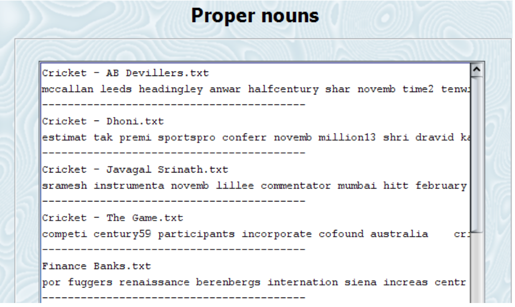
Fig. 15. Title Sentence



```
Cricket - AB Devillers.txt
18 2016 17 1984 2 50 100 150 50 3 2004 2005 2006 2016 8000 21 36 7
--------------------------------------------------------------
Cricket - Dhoni.txt
2007 2007 2010 2011 2013 2011 91 79 2013 2008 2008 2010 2013 2009
--------------------------------------------------------------
Cricket - Javagal Srinath.txt
1999 2008 31 1969 198990 25 20 793 311 638 1991 44 34 300 199192 3
--------------------------------------------------------------
Cricket - The Game.txt
2005 1889 1928 1929 1932 1952 1982 1992 2000 2006 2011 38 39 1882
--------------------------------------------------------------
Finance Banks.txt
1 14 1997 2007 1997 2007 16 2007
--------------------------------------------------------------
```

Fig. 16. Numerical Data

A dictionary scan is run, every word in the document is matched against the words in dictionary and only nouns are processed.



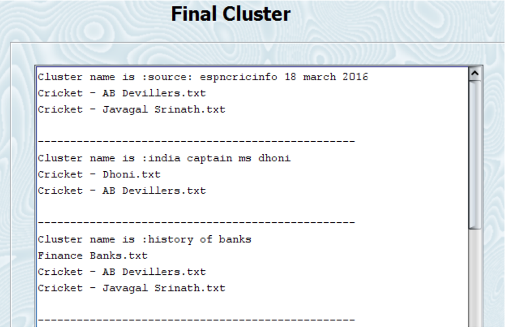
```
Proper nouns

Cricket - AB Devillers.txt
mccallan leeds headingley anwar halfcentury shar novemb time2 tenwi
----------------------------------------
Cricket - Dhoni.txt
estimat tak premi sportspro conferr novemb million13 shri dravid ka
----------------------------------------
Cricket - Javagal Srinath.txt
sramesh instrumenta novemb lillee commentator mumbai hitt february
----------------------------------------
Cricket - The Game.txt
competi century59 participants incorporate cofound australia    cri
----------------------------------------
Finance Banks.txt
por fuggers renaissance berenbergs internation siena increas centr
----------------------------------------
```

Fig. 17. Proper Noun

Now, generate the Weighted Matrix

```
Cricket - AB Devillers.txt
[0, 0.14, 0.76, 0.24, 0.04, 0.0, 0.08, 0.0, 0.01, 0.19, 0.26, 0.02]
---------------------------------------
Cricket - Dhoni.txt
[0.64, 0, 0.42, 0.34, 0.06, 0.0, 0.02, 0.0, 0.02, 0.27, 0.16, 0.02]
---------------------------------------
Cricket - Javagal Srinath.txt
[0.47, 0.08, 0, 0.23, 0.03, 0.0, 0.1, 0.0, 0.03, 0.18, 0.25, 0.01]
---------------------------------------
Cricket - The Game.txt
[0.39, 0.12, 0.46, 0, 0.02, 0.0, 0.06, 0.0, 0.0, 0.18, 0.23, 0.02]
---------------------------------------
Finance Banks.txt
[0.91, 0.31, 1.44, 0.28, 0, 0.0, 0.22, 0.14, 0.14, 0.44, 0.06, 0.09]
---------------------------------------
Finance Derivatives.txt
[0.0, 0.0, 0.0, 0.0, 0.0, 0, 0.0, 0.0, 0.06, 0.0, 0.0, 0.0]
---------------------------------------
Finance Insurance.txt
[0.62, 0.06, 1.31, 0.06, 0.1, 0.0, 0, 0.02, 0.03, 0.29, 0.2, 0.15]
---------------------------------------
Finance Loans.txt
[0.0, 0.0, 0.0, 0.0, 0.15, 0.0, 0.12, 0, 0.12, 0.12, 0.0, 0.0]
---------------------------------------
              Finance Shares.txt
              [0.6, 0.07, 1.36, 0.05, 0.13, 0.05, 0.13, 0.12, 0,
              0.18, 0.35, 0.0]
              ---------------------------------------
              Travel _ Essex.txt
              [0.43, 0.17, 0.62, 0.25, 0.05, 0.0, 0.13, 0.02,
              0.03, 0, 0.34, 0.13]
              ---------------------------------------
              Travel _ London.txt
              [0.26, 0.04, 0.4, 0.15, 0.0, 0.0, 0.06, 0.0, 0.01,
              0.2, 0, 0.06]
              ---------------------------------------
              Travel_surrey.txt
              [0.38, 0.09, 0.47, 0.1, 0.09, 0.0, 0.19, 0.0, 0.0,
              0.38, 0.18, 0]
              ---------------------------------------
```

The final logical clusters formed are shown in Fig. 18.



```
Final Cluster

Cluster name is :source: espncricinfo 18 march 2016
Cricket - AB Devillers.txt
Cricket - Javagal Srinath.txt

-------------------------------------------------
Cluster name is :india captain ms dhoni
Cricket - Dhoni.txt
Cricket - AB Devillers.txt

-------------------------------------------------
Cluster name is :history of banks
Finance Banks.txt
Cricket - AB Devillers.txt
Cricket - Javagal Srinath.txt

-------------------------------------------------
```

Fig. 18. Logical Clusters

TABLE III. EXPERIMENT CONDUCTED ON SYSTEM HAVING CONFIGURATION AS INTEL I7-970 PROCESSOR WITH 4 GB RAM *HARD CLUSTER: DOCUMENT BELONGS STRICTLY TO ONE CLUSTER*

	No. of documents as input	**No. of Hard clusters**	**No. of overlapping clusters (Cross Domain)**	**Time required to form cluster (in seconds)**
Proposed algorithm	120	0	**32**	157
K-Means algorithm	120	11	**0**	160

The number of clusters formed and time required in forming clusters using our proposed algorithm and K-means algorithm is shown in Table III. The proposed algorithm outperforms.

A set of documents used for evaluation has following features:

1) *Number of documents per category*

2) *Evenness in number of documents in each category*

3) *Size of each document i.e. the number of words in each document*

4) *Similarity of documents of same category compared to similarity of documents of different categories.*

5) *Number of unique words in all the documents. The quality of the results of the clustering algorithms depends very much on the features of the set of documents on which it is applied. For example, some algorithms may give good results in case of large documents as compared to small documents.*

The documents of BBC dataset are large news articles and thus the names of people, places, organizations, etc. play an important role in them and this gives the consideration of co-occurrence of words a huge importance. For example, 'Tendulkar' and 'cricket' are two different words which co-occur many numbers of times in news articles. Now, if an article contains only 'Tendulkar' then the feature based approach will still put it in the cluster of articles related to cricket or sports but this will not be the case with other algorithms.

VI. CONCLUSION AND FUTURE WORK

The experiment conducted shows that weighted feature matrix when combined with the application of Fuzzy Logic yields accurate results for the overlapping clusters of the web documents. Thus this gives us a deep insight of interlinked or interconnected documents. It also gives us an option of clustering the web document by application of fuzzy logic by using the feature extraction method. Feature extractions enable the cardinality of the data that is to be extracted. We extracted proper nouns, numerical data, top words, term weight for this experiment. By using weighted matrix it gave us the flexibility for the calculations and an ease of computing the results. Matrix formation process on the basis of the features extracted is a unique method and the threshold value would reap us the results of the overlapping clusters. Large sets of web documents which are inter-related could be classified into clusters by using this novel way by the application of the fuzzy logic. The algorithm devised in this work is at very rudimentary stage and there are many possibilities for improvements. Some of the work that can be done on it is elaborated in this section.

This work can be enhanced by application of the Natural language Processing. Documents which are in Marathi, Hindi, Chinese, and Japanese, etc. could be easily classified into the logical overlapping clusters with the application of the fuzzy logic algorithm. Also, another enhancement could be increasing the feature set as per the domain example a feature as a Date could be used in formation of forensic data or historical data. Similarly domain knowledge from various fields like Medicine, Sports, Financial Services, Space Research, Weather Reports, etc. could be applied in this experiment.

This work could be migrated to next level for deep learning and data analytics by migrating it to "R" programming for reaping the best results.

As of now, considering the limitations, we have used the open source APIS for the crawler, parser, dictionary, reading the data from excel files. These open source APIS have a limitation on the volume of data that could be processed, the open source dictionary also does not cater all the noun forms of the words. It may be enhanced by using self-developed APIS.

REFERENCES

[1] Kotsiantis, S. B., D. Kanellopoulos, and P. E.Pintelas. "Data pre-processing for supervised leaning." International Journal of Computer Science 1.2 (2006): 111-117.

[2] The wall, Mike. "A web crawler design for data mining." Journal of Information Science 27.5 (2001): 319-325.

[3] M. Ram swami and R. Bhaskaran, "A Study on Feature Selection

[4] Liu, Jin-Hong, and Yu-Liang Lu. "Survey on topic-focused Web crawler."Appl. Res. Computer 24.2629 (2007)

[5] Frakes, William B. "Stemming algorithms." (1992): 131-160.

[6] Govind MurariUpadhyay, KanikaDhingra,"Web Content Mining: Its Techniques and Uses", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 11, November 2013, pp.610-613

[7] Fayyad, Usama, Gregory Piatetsky-Shapiro, and Padhraic Smyth. "From data mining to knowledge discovery in databases." AI magazine 17.3 (1996)

[8] Munk, Michal, JozefKapusta, and Peter Švec. "Data preprocessing evaluation for web log mining: reconstruction of activities of a web visitor." Procedia Computer Science 1.1 (2010): 2273-2280.

[9] Khasawneh, Natheer, and Chien-Chung Chan. "Active user-based and ontology-based web log data pre-processing for web usage mining." Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence.IEEE Computer Society, 2006.

[10] JH Paik, MandarMitra, Swapan K. Parui, KalervoJarvelin, "GRAS: An effective and efficient stemming algorithm for information retrieval", published in ACM Transaction on Information System (TOIS), Volume 29 Issue 4, December 2011, Chapter 19, page 20-24

[11] M. Bacchin, N. Ferro, and M. Melucci 2005."A probabilistic model for stemmer generation". Inf. Process. Manage. 41, 1, 121–137.

[12] The Text Book of Data Mining, Kimball

[13] Subramanian Appavu Alias Balamurugan, Ramasamy Rajaram "Effective and Efficient Feature Selection for Large-scale Data Using Bayes' Theorem", International Journal of Automation and Computing, Volume6, Issue 1, Feb 2009, pp. 62-71

[14] H. Ying, "A Fuzzy Systems Technology: A Brief Overview" IEE Press, 2000

[15] Ojokoh, B. A., Omisore, M. O, Samuel, O. W, and Ogunniyi, T. O. Department of Computer Science Federal University of Technology, A Fuzzy Logic Based Personalized Recommender System, IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol. 2, No.5, October 2012 1008

[16] Chris Cornelis a, Jie Lu b , XuetaoGuo b , Guanquang Zhang,One-and-only item recommendation with fuzzy logic techniques, Information Sciences, Volume 177, Issue 22, 15 November 2007, Pages 4906-4921

[17] Luis Ter´an and Andreas Meier Information Systems Research Group, University of Fribourg, A Fuzzy Recommender System for eElections, K.N. Andersen et al. (Eds.): EGOVIS 2010, LNCS 6267, pp. 62–76, 2010. c Springer-Verlag Berlin Heidelberg 2010

[18] Tung-Cheng Hsieh, Tzone-I Wang* , Chien-Yuan Su and Ming-Che Lee , A Fuzzy Logic-based Personalized Learning System for Supporting Adaptive English Learning, January 2012, Educational Technology & Society

[19] Aaditeshwar Seth and Jie Zhang School of Computer Science University of Waterloo, ON, Canada, A Social Network Based Approach to Personalized Recommendation of Participatory Media Content, Copyright c 2008, Association for the Advancement of Artificial Intelligence

[20] I-Jen Chiang, Member, IEEE, Charles Chih-Ho Liu, Yi-Hsin Tsai, and Ajit Kumar, Discovering Latent Semantics in Web Documents using Fuzzy Clustering, IEEE Transactions on Fuzzy Systems (Volume: 23, Issue: 6, Dec. 2015), **Page(s):** 2122 - 2134

[21] A. Ali, and N. Mehli, "A Fuzzy Expert System for Heart Disease Diagnosis", Proceedings of the International MultiConference of Engineers and Computer Scientists, Vol. 1, Pp. 134-139, 2010.

[22] Khalid, Samina, Khalil Tehmina,NasreenShamila, A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning, IEEE Science and Information Conference, 2014, pp. 372-378.

Declaration: "The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper."