International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 4, Issue 10, ISSN No. 2455-2143, Pages 414-423
Published Online February 2020 in IJEAST (http://www.ijeast.com)

A NEURO-FUZZY BASED DOCUMENT TRACKING & CLASSIFICATION SYSTEM

Ituma, C.,
Department of Computer Science
Ebonyi State University,
Abakaliki-Nigeria

James, G. G.
Department of Computer Science
Ebonyi State University,
Abakaliki-Nigeria

Onu, F. U.
Department of Computer Science
Ebonyi State University,
Abakaliki-Nigeria

Abstract - The worldwide web is an extraordinary resource for gaining access to information of all kinds. Each day a greater number of resources become available online. The advantages that internet offers researchers are tremendous; so much that some may be tempted to bypass the library entirely and conduct all their research on the web. In spite of this, the enquirers are being confronted with information overloaded with noise which the conventional search methods lack the capability to tackle. Existing search engines such as Google, Yahoo and Bing often return a long list of results which forces users to sift through it to find relevant documents; thereby making search for information difficult. This paper takes a critical study of document tracking and classification systems and presents a Neuro-fuzzy based model for classification of search results based on the strength of words in the query. The structured systems analysis and design methodology and the neuro-fuzzy clustering technique were employed to study and classify the documents into groups of similar topics for specific knowledge. The work built an adaptive intelligent information retrieval system which will cluster internet documents into similar topic using an unsupervised machine learning techniques to reduce the percentage of irrelevant documents that are retrieved and presented to users.

Keywords: Fuzzy Logic, Neural Networks, Neuro-fuzzy technique, Document Retrieval.

I. INTRODUCTION

a. Background

Searching for information on the internet to study state of the art for research purpose to a large extent depends on our ability to track all related topics and classify them into groups of similar topics. As information grows rapidly over the web, it becomes difficult for researcher to find the information they are looking for. After providing a query to a conventional search engine, a long list of search results divided into a lot of pages is presented: similar results are scattered in the list, appearing in different pages. Without a proper arrangement of search results, finding the desired query result among ranked list of document snippets is usually difficult for most users as a result of information overload which conventional search methods do not seem to be able to tackle. The problem is further aggravated when query belongs to a general topic which contains documents from a variety of subtopics. This has led to the need for the development of a novel technique using the combination of neural network and fuzzy logic tagged neuro-fuzzy to assists users to effectively navigate, trace and organize the available web documents, with the ultimate goal of finding those best matching their topic needs.

This work is further motivated by the limitations in the following reviews:

The focus of this work was to build an intelligent agent that learn to retrieve and extract information, development of an information retrieval (IR) sub-engine, as well as information extractor (IE) sub-engine.

II. DOCUMENT CLUSTERING

Cluster analysis is an unsupervised learning method aiming to group instances into different subsets, or clusters, such that each subset contains similar observations with respect to some predefined criterion. Clustering has been subjected to intensive studies in different fields, including statistics, machine learning, data mining, image analysis and information retrieval. However, the clustering paradigm has known a fast development in both theoretical and practical domains of computer science due mainly to the ability of computers to cluster huge amounts of data in a short lap of time. Most clustering, approaches fall into three different categories: similarity-based, projection-based and density-based clustering methods [27].

Projection-based clustering approaches make, generally, use of the set of the eigenvalues of a similarity matrix of data to perform dimensionality reduction for clustering in fewer dimensions [16].

Density-based clustering techniques estimate the probability distributions of clusters and then assign each instance to its most probable cluster [5, 2, 17]. Among various density-based clustering algorithms that havç been proposed since the late 70's, topic models have recently received wide attention in both data mining and information retrieval communities.

III. WEB DOCUMENT CLUSTERING APPROACHES

There are many document clustering approaches proposed in the literature. They differ in many parts such as the types of attributes they use to characterize the documents, the similarity measure used, the representation of the clusters etc. Based on the characteristics or attributes of the documents that are used by the clustering algorithm, the different approaches can be categorized into:

i. Text-based, in which the clustering is based on the content of the document.
ii. Link-based, based on the link structure of the pages in the collection.
iii. Hybrid ones, which take into account both the content and the links of the document.

Most algorithms in the first category were developed for use in static collections of documents that were stored and could be retrieved from a database and not for collections of web pages, although they are used for the latter case too. But, contrary to traditional document retrieval systems, the World Wide Web is a directed graph. This means that apart from its content, a web page contains other characteristics that can be very useful to clustering.

i. Text-based Clustering

The text-based web document clustering approaches characterise each document according to its content, i.e. the words (or sometimes phrases) contained in it. The basic idea is that if two documents contain many common words then it is likely that the two documents are very similar.
The text-based approaches can be further classified according to the clustering method used into the following categories:

ii Partitional Clustering

The partitional or non-hierarchical document clustering approaches attempt a flat partitioning of a collection of documents into a predefined number of disjoint clusters. Partitional clustering algorithms are divided into iterative or reallocation methods and single pass methods. Most of them are iterative and the single pass methods are usually used in the beginning of a reallocation method, in order to produce the first partitioning of the data. The partitional clustering algorithms use a feature vector matrix and produce the clusters by optimising a criterion function. Such criterion functions are the following:

(i) maximize the sum of the average pairwise cosine similarities between the documents assigned to a cluster;
(ii) minimize the cosine similarity of each cluster centroid to the centroid of the entire collection etc.

Zhao and Karypis (2001) compared eight criterion functions and concluded that the selection of a criterion function can affect the clustering solution and that the overall quality depends on the degree to which they can correctly operate when the dataset contains clusters of different densities and the degree to which they can produce balanced clusters.

Scatter/Gather uses two linear-time partitional algorithms, Buckshot and Fractionation which also apply HAC logic. The idea is to use these algorithms to find the initial cluster centres and then find the clusters using the assign-to-nearest approach.

Finally, the single pass method [19] is another approach to partitional clustering which is based on the assignment of each document to the cluster with the most similar representative which is above a threshold. The clusters are formed after only one pass of the data and no iteration takes place. Consequently, the order in which the documents are processed influences the clustering.

The advantages of these algorithms consist in their simplicity and their low computational complexity. The disadvantage is that the clustering is rather arbitrary since it depends on many parameters, like the values of the target number of clusters, the selection of the initial cluster centroids and the order of processing the documents.

iii Hierarchical Clustering

Hierarchical clustering algorithms produce a sequence of nested partitions.
Usually the similarity between each pair of documents is stored in a (n x n) similarity matrix. At each stage, the algorithm either merges two clusters (agglomerative methods) or splits a cluster in two (divisive methods). The result of the clustering can be displayed in a tree-like structure, called a dendrogram, with one cluster at the top containing all the documents of the collection and many clusters at the bottom with one document each. By choosing the appropriate level of the dendrogram we get a partitioning into as many clusters as we wish. The dendrograrn is a useful representation when considering retrieval from a clustered set of documents, since it indicates the paths that the retrieval process may follow]19].

Single link: The similarity between a pair of clusters is calculated as the similarity between the two most similar documents, one of which is in each cluster. This method tends to produce long, loosely-bound clusters with little internal cohesion (chaining effect). The single link method incorporates useful mathematical properties and can have small computational complexity. There are many algorithm based on this method. Their complexities vary from O(nlogn) to O(n^5).

Complete link: The similarity between a pair of clusters is taken to be the similarity between the least similar documents, one of which is in each cluster. This definition is much stricter than that of the single link method and thus, the clusters are small and tightly bound. Implementations of this method are the CLINK algorithm [5] which is a variation of the SLINK algorithm, and the algorithm proposed by Villmann [23].

Group average: This method produces clusters such that each document in a cluster has greater average similarity with the other documents in the cluster than with the documents in any other cluster. All the documents in the cluster contribute in the calculation of the pairwise similarity and thus, this method is a mid-point between the above two methods. Usually the complexity of the group average algorithm is higher than 0(n2). Voorhees proposed an algorithm for the group average method that calculates the pairwise similarity as the inner product of two vectors with appropriate weights. Steinbach et al. (2000) used UPGMA for the implementation of the group average method and obtained very good results.

Ward's method: In this method the cluster pair to be merged is the one whose merger minimizes the increase in the total within-group error in which sum of the squares based on the distance between the cluster centroids (i:e. the sum of the distances from each document to the centroid of the cluster containing it). This method tends to result in spherical, tightly bound clusters and is less sensitive to outliers. Ward's method can be implemented using the reciprocal-nearest neighbour (RNN) algorithm which was modified for document clustering by [7].

Centroid/Median Methods: Each cluster as it is formed is represented by the group centroid/median. At each stage of the clustering, the pair of clusters with the most similar mean centroid/median is merged. The difference between the centroid and the median is that the second is not weighted proportionally to the size of the cluster.

iv Neural Network based Clustering

The Kohonen's Self-Organizing feature Maps (SOM) [11] is a widely used unsupervised neural network model. It consists of two layers: the input layer with n input nodes, which correspond to the n documents, and an output layer with k output nodes, which correspond to k decision regions (i.e. clusters). The input units receive the input data and propagate them onto the output units. Each of the k output units is assigned a weight vector. During each learning step, a document from the collection is associated with the output node, which has the most similar weight vector. The weight vector of that 'winner' node is then adapted in such a way that it will become even more similar to the vector that represents that document, i.e. the weight vector of the output node 'moves closer' to the feature vector of the document. This process runs iteratively until there are no more changes in the weight vectors of the output nodes. The output of the algorithm is the arrangement of the input documents in a 2- dimensional space in such a way that the similarity between the input documents is mirrored in terms of topographic distance between the k decision regions.

v Fuzzy Clustering

All the aforementioned approaches produce clusters in such a way that each document is assigned to one and only one cluster. Fuzzy clustering approaches, on the other hand, are non-exclusive, in the sense that each document can belong to more than one cluster. Fuzzy algorithms usually try to find the best clustering by optimizing a certain criterion function. The fact that a document can belong to more than one cluster is described by a membership function. The membership function computes for each document a membership vector, in which the i-th element indicates the degree of membership of the document in the i-th cluster.

vi Neuro-Fuzzy

In the field of artificial intelligence, neuro-fuzzy refers to combinations of artificial neural networks and fuzzy logic. It is a hybridization which results in a hybrid intelligent system that synergizes the two techniques by combining the human-like reasoning style of fuzzy systems with the learning and connectionist structure of neural networks. This combination is widely termed as Fuzzy Neural Network (FNN) or Neuro-Fuzzy Systems (NFS) in the literature. Neuro-Fuzzy system incorporates the human-like reasoning style of fuzzy systems through the use of fuzzy sets and a linguistic model consisting of a set of IF – THEN Fuzzy rules.

IV. RELATED WORK

Varlamis *et al*. (2000), proposed a system called THESUS, which clusters web documents that are characterized by weighted keywords of an ontology. The ontology used is a tree of terms connected according to the IS-A relationship. Given this ontology and a set of document characterized by keywords, the algorithm proposes a clustering scheme based on a novel similarity measure between sets of terms that are hierarchically related. Firstly, the keywords that characterize each document are mapped onto terms in the ontology. Then, the similarity between the documents is calculated based on the proximity of their terms in the ontology. In order to do that, an extension of the Wu and Palmer similarity measure is used [24].The algorithms described above, most often rely on exact keyword matching, and do not take into account the fact that the keywords may have some semantic proximity between each other. This is, for example, the case with synonyms or words that are part of other words (whole-part relationship). For instance, a document might be characterized by the words "camel, desert" and another with the word "animal, Sahara". By using traditional techniques, these documents would be judged unrelated. Using an ontology can help capture this semantic proximity of the documents. Ontology, in this context, is a structure (a lexicon) that organizes words in a net connected according to the semantic relationship that exists between them. More on ontologies can be found in [5].The advantage of using ontology in clustering is that it provides a very useful structure not only for the calculation of document similarity, but also for dimensionality reduction by abstracting the keywords that characterize the documents to terms in the ontology.

Kleingberg (1997) developed a link-based clustering model which takes into account information extracted by the link structure of the collection. The underlying idea is that when two documents are connected via a link, there exist semantic relationships between them, which can be the basis for the partitioning of the collection into clusters.

The use of the link structure for clustering a collection is based on citation analysis from the field of bibliometrics [25]. Citation analysis assumes that if a person creating a document cites two other documents then, these documents must be somehow related in the mind of that person. In this way, the clustering algorithm tries to incorporate the human judgment when characterizing the documents. Two measures of similarity between two documents p and q based on citation analysis that are widely used are: co-citation, which is the number of documents that co-cite p and q and bibliographic coupling, which is the number of documents that are cited by both p and q. The greater the value of these measures the stronger the relationship between the documents p and q is. Also, the length of the path that connects two documents is sometimes considered when calculating the document similarity.

Pageet et al. (1998) also proposed an algorithm for the ranking of the search results. Their approach, PageRank, assigns at each web page a score, which denotes the importance of that page and depends on the number and importance of pages that point to it. According to Kleinberg (1997), This algorithm is used for the identification of mutually reinforcing communities called hubs and authorities. Pages with many incoming links are called authorities and are considered very important. The hubs are pages that point to many important pages.

Botafogo and Shneiderman (1993) also proposed another link-based algorithm which approached was based on a graph theoretic algorithm that found strongly connected components in a hypertext's graph structure. The algorithm uses a compactness measure, which indicates the interconnectedness of the hypertext, and is a function of the average link distance between the hypertext nodes. The higher that compactness the more relevant the nodes are. The algorithm identifies clusters as highly connected sub-graphs of the hypertext graph. Later, Botafogo (1993) extended his idea to include also the number of the different paths that connect two nodes in the calculation of the compactness. This extended algorithm produces more discriminative clusters, with reasonable size and with highly related nodes.

Larson (1996), applied co-citation analysis to a collection of web documents. Co-citation analysis begins with the construction of a co-citation frequency matrix, whose ij-th entry contains the number of documents citing both documents i and j. Then, correlation analysis is applied to convert the raw frequencies into correlation coefficients. The last step is the multivariate analysis of the correlation matrix using multidimensional scaling techniques (SAS MDS), which mirrors the data onto a 2-dimensional map. The interpretation of the 'map' can reveal interesting relationships and groupings of the documents. The complexity of the algorithm is $O(n^2/2\text{-}n)$.

Pirolli *et al.,* (1996) proposed a method that represents the pages as vectors containing information from the content, the linkage, the usage data and the meta-information attached to each document. The method uses spreading activation techniques to cluster the collection. These techniques start by 'activating' a node in the graph (giving a starting value to it) and 'spreading' the value across the graph through its links. In the end, the nodes with the highest values are considered much related to the starting node. The problem with the algorithm proposed by Pirolli *et al.,* is that there is no scheme for combining the different information about the documents. Instead, there is a different graph for each attribute (text, links etc.) and the algorithm is applied to each one, leading to many different clustering solutions.

The text similarity is computed as the normalized dot product of the term vectors representing the documents. The link similarity is a linear combination of three parameters:

(i) the number of Common Ancestors (i.e. common incoming links);

(ii) the number of Common Descendants (i.e. common outgoing links) and;

(iii) the number of Direct Paths between the two documents.

The strength of the relationship between the documents is also proportional to the length of the shortest paths between the two documents and between the documents and their common ancestors and common descendants.

V. DOCUMENT CLUSTERING FOR INFORMATION RETRIEVAL

Document clustering has initially been investigated in Information Retrieval mainly as a means of improving the performance of search engines by pre-clustering the entire corpus [20; 3]. The cluster hypothesis [23] stated that similar documents will tend to be relevant to the same queries, thus the automatic detection of clusters of similar documents can improve recall by effectively broadening a search request. However, this approach has not been shown to be superior to standard near-neighbour searches.

Teufel and Moens (2002) and Siddharthan and Teufel (2007) introduced a scientific attribution task which tries to attribute scientific work to citations. They describe Argumentative Zoning which is a discourse analysis technique that labels sentence according to their role in the authors' argument for example contrasting, background. The aim in this case is to identify the novel claim or contribution of a cited paper by analysing its citations using this technique. Their experiments were conducted on conference articles in computational linguistics and their evaluation which used comparison to human- annotated attribution that showed a very high agreement (around 80%) with human gold standard annotation. Another interesting work based on citation contexts is introduced [6]. They provided a quantitative analysis of the benefits of citation contexts with regards to other applications such as summarization and information retrieval. In particular, they examined the relationship between the abstract and citation contexts of a given scientific paper.

Table 1: Features of a conventional search engine over Software agent

	Search Engine Features	Improvement(s) Intelligent Software Agent can offer:
1	An information search is done, based on one or more keywords given by a user. This presupposes that the user is capable of formulating the right set of keyboards to retrieve the wanted information.	Agents are capable of searching information more intelligently, for instance because too (such as a thesaurus) enable them to search a related term as well, or even on concepts.
2	Information mapping is done by gathering (meta-) information about information and documents that are available on the internet. This is a very time-consuming method that causes a lot of data traffic, it lacks efficiency	Individual user agents can create their own knowledge base about available information sources on the Internet, which is updated an expanded after every search. When information (i.e. documents) has moved to another location agents will be able to find them, and update their knowledge base accordingly.
3	The research for information is often limited to a few internet services, such as the www. Ending information that is offered through other services, often means that user is left to his or her own devices;	Agents can relief their human user of the need to worry about "clerical details", such as way the various Internet service have operated. Instead, he or she will only have worry about the question what exactly is being sought.
4	Search engines cannot always be reached: the server that a service resides on may be 'down', or it may be too busy on the internet to get a connection.	As a user agent resides on a user's computer is always available to the user.
5	Search engines are domain-independent in the way they treat gathered information and, in the way, they enable users to search in it.	Software agents will be able to search information based on contexts. They will deduce this context from user information (in a built-up user model) or by using other services, such as a thesaurus service.
6	The information or internet is very dynamic: quite often search engines refer to information that has moved to another, unknown	User agents can adjust themselves to the preferences and wishes of individual users he/she is looking for, by learning from performed tasks (i.e. searches) and the users react to the results of them.

a. Experimental Observations with Search Engines

Since different search engines provide different services and features, comparison among them is an important matter for users and the parameters that can be used to compare search engines are experimentally reviewed in details in table 2, 3 and 4 respectively

Table 2: The Evaluation Parameters of search engines from "searching features" perspective

Evaluation Parameter	Description
Default search	How does the engine put the keywords together, for example 'AND' between the words (inclusive), or 'OR' between them (alternative).
Keyword/Concept default	Concept searching occurs when the engine not only search for the exact character string, but also for word forms, and even synonyms and other words that statistically appear with the

	typed word.
Exclusion possibility	Ability to exclude web pages (results) including special terms, search engines represent it by putting a minus or 'NOT' in front of excluded term
Truncation	Possibility of finding various form of a word by adding a truncation symbol (such as '*') on the end of the word.
Search restrictors	Ability to search for terms or values contained only in certain portions of a page, rather than anywhere in the entire page or within special kind of pages (sound, image, video) or in special site domains (.com, .edu).
Date searching restrictor	Try to place a date restriction in search query. Date restrictions can be useful to locate newly created or recently updated web pages, weeding out older results.
Phrase searching	Ability of using quotation marks around some terms or a kind of Boolean connector such as ADJ between the terms for phrase searching.
Nesting	Support the use of parentheses to nest various parts of a search query, for example (apple or blueberry) ADJ pie that means apple pie OR blueberry pie.
Multi-Level search	Ability to first casting a wide net then narrowing by searching only within that set of results.
Case sensitive	If the search engine is case sensitive or not?
Language restrictor	Ability to search the web pages in various languages such as English, German, ….
Natural Language support	Can it handle queries in natural language

Table 3: The Evaluation Parameter of search engine coverage, database and manner of search

Evaluation parameter	Description
Content size	How big is its database, i.e how many web pages are indexed in its database.
Search parts	If they search full text of web page or a specific part of it such as keywords, titles, headings, links of web pages,…..
Various Kind of web resources indexing	Indexing the document from other internet sources such as Usenet, peoples, emails texts,….
Focused topic	Whether the search engine focuses on a specific or document type or it is a general-purpose one.
Web crawling strategy	The manner that search traverses the web link's graph, for example breadth-first-search, according to priority queue and some parameters, such as Hub and authority score of page, page-ranking.

Table 4: The Evaluation Parameter of Search Engine Results from Returned Result Perspective

Evaluation parameter	Description
Web pages Ranking methods	Different parameters used to specify the rank of web pages in returned result list, such as site popularity.
Various display option	If various options are available to rank the returned result, such as by date, by site,…
Suggested search	Suggestions for further searching based on the initial search are provided, these suggestions can be simple, such as synonyms or alternative search terms, or may be more sophisticated, such as suggestions for searching in different specialized databases.
Similar searches	If someone locates a web page that is highly relevant to his research issue, he might be interested in finding more pages that are very similar, is it available?
Translated results	Possibility of offering a tool to translate a given result page from one language to another

VI. ANALYSIS OF THE EXISTING SYSTEM

The existing system in Lee and Kim (2012) is a fuzzy Web Information Retrieval System. In their work, a fuzzy web information retrieval system was developed using many of the tools and methods involved in fuzzy logic and fuzzy set theory, along with standard algorithms involving information retrieval on the web. Along with the fuzzy inference engine, the other subsystems of a search engine are also developed. A crawler is developed that adheres to standard internet etiquette rules. An inverted index database is used to store the internet document information retrieved by the crawler and an interface is also provided to allow users to search for gathered information located within the database. The results of the ranking algorithm used the fuzzy relational BK-products, fuzzy thesauri, and fuzzy closure properties for purposes of retrieving relevant documents to a user query. These results of the developed system were compared with existing search engines. The benefits of using a fuzzy information retrieval system are discussed, along with the issues that arise from using these techniques. The goal of the existing system was to rank documents retrieved from the internet that are more relevant to a user query with a higher value than those documents that are less relevant to a user query. The system used new tools to rank documents within a repository to retrieve better search results. The believe was that using the tools in fuzzy information retrieval would aid in the retrieval and ranking of relevant documents from the web.

VII. ARCHITECTURE OF THE EXISTING SYSTEM

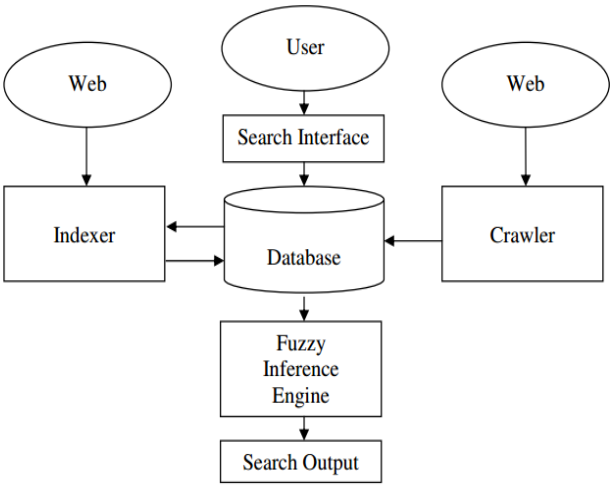

Fig. 1: Existing System Architecture (Lee and Kim, 2012)

The Existing System was divided into four individual components. These components are the crawler, the indexer, the fuzzy ranking settings, and the user's search component. The crawler allowed for a maximum link retrieval count. This enables the user to stop the crawler after it has retrieved up to a specific amount of web documents from a particular seed URL. The crawler will also adhere to standard web etiquette defined by the *robots.txt* standard.

- The steps involved in the indexing of documents and terms within this system are to retrieve a URL from the list of valid URLs from the database, extraction of document information, removal of insignificant words, insertion of document in relation to specific terms into the inverted index, and update inverted index with new weights on documents. This portion of the system is dedicated primarily to the flow of query evaluation and document ranking. This subsystem has two primary functions—the generation of the fuzzy relational matrix of terms to documents, and the generation of the fuzzy thesaurus which gives relationship from search terms to other terms, and the ranking of documents. These matrices are the primary tools used to rank documents in relation to a user query. The steps involved in defining the ranking of the fuzzy information retrieval system are as follows:
- The matrix defining the associations between terms and documents is generated from the information gathered by the indexer that is stored in the database.
- The thesaurus that defined the relationships from terms to terms is generated based on information gathered from the database. The user chooses the implication type used in generating this matrix, along with the fuzzy criteria used to evaluate the results of the fuzzy product. A transitive function can be chosen to compute the transitive closure of the search.
- The fuzzy implication, criteria, and product are chosen to define how the association between the two matrices is computed.
- The results are flattened into a single membership relation from the user query to the documents within the database.
- An α–Cut is applied to the list of retrieved documents to retrieve only relevant documents. An α-cut is a chosen value where any membership value of a document that is less than the α-cut is unrelated, and any membership value equal to or greater than that the α-cut is considered relevant to the user query. The remaining documents are ordered by rank and displayed to the user

VIII. CHALLENGES OF THE EXISTING SYSTEM

The system in Lee and Kim (2012) above suffers the following limitations:

- Using fuzzy relational methods and fuzzy logic in information retrieval on the web is the magnitude of the terminology base, and the scope of the documents to search from on the web. The concepts in fuzzy information retrieval are deeply involved with the use of matrix computations. This implies expensive spatial and temporal calculations which may not always be feasible in certain computing environments. High performance computing is mandatory for larger volumes of terms and documents.
- Tests that are limited in size however, can still be performed and evaluated on machines with fewer resources and good result sets can still be generated
- The existing system make use of Fuzzy Logic (FL), and FL lacks the capability to learn from previous data.
- There was no ANFIS architecture or well define neuro-fuzzy based model to enhance the search, training of data as well as well as classification of the web document.

- There was no hybrid intelligent search system based on neuro-fuzzy paradigm that can enhance effective tracking and classification of the documents.

IX. COMPONENTS ANALYSIS OF INTELLIGENT BASE SEARCH AND CLASSIFICATION SYSTEMS

Table 5 shows the components analysis of available intelligent base document tracking, clustering and classification systems. The analysis is based on the type of work, intelligent tool(s), author, as well as limitation(s).

Table 5: Table of analysis of available intelligent Base Search and classification Systems and their intelligent tools

Work	Author /Date	Intelligent Tools	Limitations
Classification of Arabic Documents	Taher et al., 2010	Fuzzy Proximity with a Radial Basis Function	Inability to learn from training data
Classification of Web Documents	George et al., 2010	Fuzzy Logic Categorical Data Clustering	Inability to effectively tract the required Document
Text classification	Mohan et al., 2012	Novel Fuzzy based Clustering Algorithm	The aim of classifying text not met
Arabic Document Classification	Ali & Adnan 2014	Fuzzy Logic	Challenges on how the feature can be selected, reduced and weight. And limited number of available clusters.
Text-Based Infor-mation Retrieval System	Hameed, 2008	Fuzzy Logic	There was no sophisticated hash function to enhance indexing system that can handle efficiently the matching process on a selective bases
Classification Techniques for Sentiment Analysis	Soundarya and Manjula, 2015	Neuro-Fuzzy	No hybrid intelligent para-digm to enhance effective tracking and classification of documents
Information retie-val From text documents in slovak language	Mokriš and Skovajsová	Neural Network Model	The paradigm was not suitable to track document in other Languages.
Intelligent search system for topic tracking & classification of Document	Ejiofor, 2014	Fuzzy Clustering	The system was not able to support advanced search needs; Inability to automatically expand user query without user intervention
Event-based multi document summa-rization	Lus Cartos, 2015	fuzzy fingerprint	Difficulty to relate differ-rent descriptions of the same event due to differ-rent lexical realizations
Survey of Topic Tracking Techniques	Kermaldeep et al., 2012	A conventional topic tracking system through the 'bag-of-words'	No competent mathematical model to measure to-pic – story similarity and do threshold comparison that determine if similarity of topics is higher than predefined threshold so as to fridge is if the topic is on – topic as off-topic

X. ANALYSIS OF THE PROPOSED SYSTEM

The proposed system is a neuro-fuzzy based model for searching and classification of document using library of e-books as a case study. FL lacks the capability to learn from previous data and NN equally lacks the capability to handle imprecise and incomplete data. This makes Neuro-Fuzzy systems one of the best option for document tracking and classification as the weakness of Fuzzy Logic and Neural Network are complimented whilst the strength of the individual components are enhanced. The document that shall be used as case study is collected from TREC of 2011. The system shall have a four (4) input variables and two (2) outputs. The input variables to be considered are: File size, Search Words (keywords), search index and multiplicity. The system shall be made up of two sections; that is, the fuzzy Logic section and then the neural network section. The fuzzy section shall handle the tracking while the neural network will do the classification and retrieval and as well optimized fuzzy retrieval system.

XI. BENEFITS OF THE PROPOSE SYSTEM

The propose system have the following benefits:

- The system has the capability of handling imprecise, incomplete and vague information, as well as ability to represent partial truth.
- The system has the capabilities of fault tolerance, parallelism, learning from training data, recalling memorized information and generalizing to the unseen patterns.
- The hybridization will help, the weakness of Fuzzy Logic and Neural Network are complimented whilst the strength of the individual components are enhanced to enhanced an effective and efficient tracking and classification of web document.
- The system has an inbuilt hybrid intelligent search model based on neuro-fuzzy paradigm that has the capability to cluster internet documents into similar topic using an unsupervised machine learning techniques to reduce the percentage of irrelevant documents that are retrieved and presented to users.
- There is available ANFIS architecture and well-defined neuro-fuzzy based mathematical models to enhance the intelligent searching, tracking, clustering, ranking and classification of the relevant web documents.
- Availability of a vector model that enhances feature selection, reduction and weighting.
- It has the ability to classify documents into similar group for domain knowledge.
- It has the ability to automatically infer plausible training e.g. by observing users' normal use of the browsers.
- The system is able to support advanced search needs.
- It has the ability to automatically expand user query without user intervention.

XII. CONCLUSION

The document that was used as case study were retrieved from Information Retrieval Conference (TREC) of 2011 INFILE collections. Data such as Term Weighting, Lexical Density, represent Similarity Vector and Word Ratio respectively which represent the dataset was divided into three parts; training, validation and testing dataset in the ratio of 3:5:1 respectively and translates to 300 records for training, 484 data points for validation and 107 records for testing of the system. The system is made up of two sections; that is, the fuzzy Logic section and then the neural network section. The fuzzy section shall handle the tracking while the neural network will do the classification and retrieval and as well optimized fuzzy retrieval system. The essence of applying the neuro-fuzzy techniques is to build an adaptive intelligent information retrieval system which will cluster electronics library documents into similar topic using an unsupervised machine learning technique to reduce the percentage of irrelevant documents that are retrieved and presented to users.

XIII. REFERENCES

1. Botafogo, R. (1993). Clsuter analysis for hypertext systems. Proc. ACM SIGIR Conference on Research and Development in Information Retrieval 116-125.

2. Celeux, G. and Govaert, G. (1992). A classification based algorithm for clustering and two stochastic version. Computational Statistics and Data Analysis, 14(3):315-332.

3. Croft, W. (1993). Retrieval strategies for hypertext. Information Processing and Management, 29:313-324.

4. Cutter M., Deng H., Mannicam S., and Meng W. (1999). A new study on using html Structure to improve retrieval. The eleventh IEE Conference on tools with AL.

5. Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society. series B (Methodological), 39(1):1-38.

6. Ejiofor, C. (2014), An Intelligent search system for topic tracking and classification of documents, Ph.D dissertation of University of Port Harcourt, Nigeria, June 2014.

7. Hearst, M. (1996). Improving full-text precision on short queries using simple constraints. Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval, Univ. Nevada, Las Vegas, NV, USA, 217-228.

8. Kamadeep, K., and Vishal, G. (2012), A survey of topic tracking techniques, international journal of Advanced Research in computer science and software engineering, volume 2, Issue 5, pp. 384-385.

9. Kaufman, L. and Rousseeuw, P. (1990). Finding Groups in Data: An Introduction to cluster Analysis: Wiley.

10. Kleinberg, J. (1997). Authoritative sources in a hyperlinked environment. Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms.

11. Kohonen, T. (1995). Self-organizing maps. Springer-Verlag, Berlin.

12. Kumar, S., Raghavan, P., Rajagopalan, S. and Tomkins, A. (1999). Traveling the web for emerging Cyber-Communities. Proceedings 8th www. Conference.

13. Larson, R.., (1996). Bibliometrics of the World Wide Web: An Exploratory Analysis of the Intellectual Structure of Cyberspace. Proceedings of the 1996 American Society for Information Science Annual Meeting.

14. Looney, C. (1999). A Fuzzy Clustering and Fuzzy Merging Algorithm, Technical Report, CS-UNR-101-1999 (http://www.cs.unr.edu/looney/cs479.newfzclst2.pdf) Accessed: 10/04/2012.

15. LusCarlus (2015), Event-based Multi-document summarization, Arinzona state university Ph.D dissertation.

16. Michael S., and Leven-Ertoz, V. (2003). The challenges of clustering high-dimensional data. Springer-Verlag.

17. Pavlov, D., Balasubramanyan, R., Dom., B., Kapur, S., and Parikh, J. (2004). Document preprocessing for naïve bayes classification and clustering with mixture of multinomials. In Proceedings of the 10th ACM SIGKDD International Conference of Knowledge Discovery and Data mining, pages 829-834.

18. Pirolli, P., Pitknow, J., and Rao, R. (1996). Silk from a sow's ear: Extracting usable structures from the Web. Proceedings of the ACM SIGCHI Conference on Human Factors in Computing.

19. Rasmussen, E. (1992). Clustering Algorithms. Information Retrieval, W. B. Frakes and R. Baeza-Yates, Prentices Hall PTR, New Jersy.

20. Salton G., (1989). Automatic Text Processing: the Transformation, Analysis and Retrieval of Information by Computer, Addison Wesley, Reading, MA.

21. Steinbach, M., Karypis, G., and Kumar, V. (2002). A Comparative of Document Clustering Techniques. KDD Workshop on Text Mining.

22. Tina, E. (2001), Building intelligent Agents that learn to retrieve and extract information, Ph.D dissertation, 2001.

23. Villmann, T., Michael, H., and Verleysen, M. (2009). Similarity –based Clustering. Springer.

24. Weiss, R., Velez, B., Sheldon, M., Nemprempre, C., Szilagyi, P., and Gifford, D. K., (1996). HyPursuit: A Hierarchical Network Search Engine that Exploits Content-Link Hypertext Clustering Proceedings of the Seventh ACM Conference on Hypertext.

25. Willett, P. (1988). Recent Trends in Hierarchic document clustering: a critical review. Information and Management, 24(5):577-597.

26. Zhao, Y., and Karypis, G. (2001). Criterion Functions for Document Clustering: Experiments and Analysis. Technical Report #01-40. University of Minnesota, Computer Science Department. Minnesapolis, MN (http://www. user.cs.umn.edu/karypis/publicaitons/ir/html) Accesses: 04/05/2016.

27. Zhong, S. (2005). Generative Model-based document
    a. clustering: A comparative study. Knowledge and Information Systems, 8(3):374-384.