Enhanced Neuro-Fuzzy Based Information Retrieval Technique

S.R. Tandan[1], Priyanka Tripathi[2], Rohit Miri[3], Shagufta Farzana[4]
[134]Dept. of Computer Science Engineering, Dr. C.V.Raman University, Bilaspur (C.G.) India.
[2]National Institute of Technology, Raipur (C.G.) India.

ABSTRACT
In the proposed Neuro-fuzzy based model, fuzzy system have been selected to enhance the capability of document retrieval process. To obtain this, fuzzy parameter or variables that can describe main features of the document are the term Frequency ratio (tfr), Document frequency ratio (dfr), Term frequency (tf), ratio of the number of search terms that occurs in one document to the length of the search term, Inverse document frequency (idf). In the proposed Neuro-fuzzy model for Information Retrieval, three fuzzy values have been used which are Low, Medium and High to represent fuzzy linguistic values. Various membership functions were tested for fuzzy linguistic value to identify the relevancy in document for query term. The query term is tested in three datasets and computed performance of methods using confusion matrix. The performance of the proposed model is compared with existing models, such as cosine similarity, L-shape, and S-shape membership functions. The comparison is based on precision (p), recall (r), and accuracy (a) parameters. Three standard text datasets, viz. Movie Review, Polarity and ACL IMDB (Large movie review) are used to evaluate the comparative models.

Keywords: Information Retrieval, Applications of Neuro-Fuzzy System, Information Processing and Machine Learning

I INTRODUCTION

Increasing demand of digital information along with higher productivity and better quality is on a rise in today's era. To get on time delivery of updated information with accuracy, the industry is turning towards Information and Communication Technology ICT. The digital documents are growing at a very high rate because of the extensive usage of electronic media, social networking and internet. The contents of the documents are mainly text, images, and links to name a few. Almost 80% of the contents are represented in the text form [12][9][7][19]. Today Information Retrieval System is a powerful tool and becoming more and more significant in various aspects of human life, especially in industrial, commercial and scientific applications. As a result of scientific achievements and IT Industry development the number of information retrieval systems currently in use for corporate projects are increasing fast. It has become a must to maintain the same pace for the IR systems too. However, IR has been evolving ever since and there has been an increasing interest in developing information extraction systems, still this area needs to be researched and fasten a lot. IR systems heavily depend on World Wide Web www and vice-versa. Web contains an accumulation of hyperlinks, text and images. Web mining methods consist of incredible framework utilized for data extraction. In information extraction, primary task is query answering (more than question answering) system and it is the backbone of the Information Retrieval and Natural Language Processing NLP systems. Natural language processing tools manage and process the questions for retrieval of answer in the form of natural language [1]. The continuous growth in information technology requires a machine which can handle the system and extract the information correctly is the need of the current generation. The initial objective of information retrieval system is to assist the users while accessing the retrieval environment. The major user group of commercial applications is using a traditional retrieval process of information which is based on crisp and Boolean logic model. The traditional system limitation is a difficulty with dealing uncertain query. The system should be able to process the uncertain query terms. The application of fuzzy system is to provide an environment to handle uncertain data where most of the system failed to process it. The most essential element of an IR system is the Textual Archive which consist of textual units known as Document, and Document Retrieval Engine. The user enters the query with the required document information. The Document Retrieval Engine searches the similar document against query term from the knowledge base and responds with all possible lists of documents which are most relevant for the user. Thus, Information retrieval in general, is the problem of selection of document information from storage in response to search questions, i.e., to match the words or other symbols of the inquiry with those characterizing the individual document and make the appropriate selection. NFS (Neuro-Fuzzy System) and Web Content Mining is natural merging of two activities of recent research. Sandwiched of NFS can handle uncertain query and can provide best possible results. The content mining and World Wide Web can be explained as discovery and analysis of useful and essential information from web [18].

II INFORMATION RETRIEVAL SYSTEM

Information Retrieval (IR) is the activity of obtaining information relevant to an information need from a collection of information resources. Searches can be based on metadata or on full-text (or other content-based) indexing [10]. Enormous growth in data gives genesis to automated information retrieval systems. These are used to reduce the "Information overload". Many universities and public libraries use IR systems to provide access to books, journals and other documents. Web search engines are the most visible, IR applications [17].

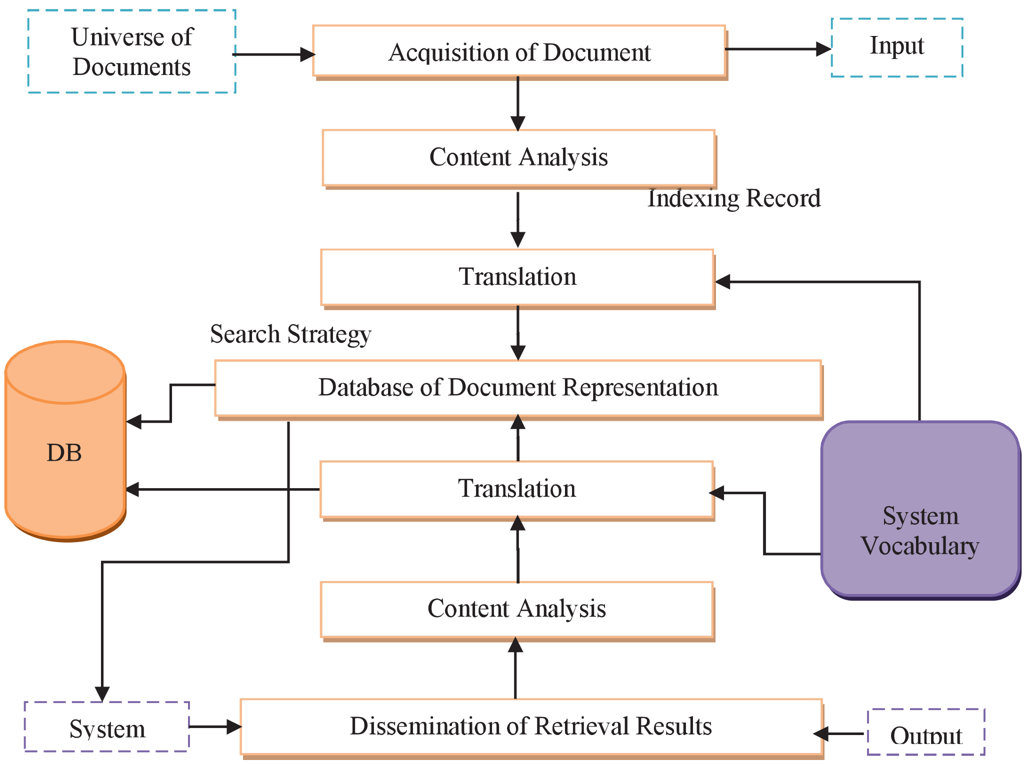

Fig. 1 Functions of an information retrieval system

Information retrieval techniques are extensively used in various applications on www. Broadly divided into two categories, namely General applications and Domain specific applications of information retrieval are listed below.

- **General applications of information retrieval are:** Digital libraries, Information filtering and Recommender systems.
- **Media search:** Blog search, Image retrieval, 3D retrieval, Music retrieval, News search, Speech retrieval and Video retrieval.
- **Search engines:** Site search, Desktop search, Enterprise search, Federated search, Mobile search, social search, Web search.
- **Domain specific applications of information retrieval are:** Expert search finding, Genomic information retrieval, Geographic information retrieval, Information retrieval for chemical structures, Information retrieval in software engineering, Legal information retrieval and Vertical search.
- **Other widely used retrieval methods are:** Techniques in which information retrieval techniques are employed include: Adversarial information retrieval, Automatic summarization, multi-document summarization, Compound term processing, Cross-lingual retrieval, Document classification, Spam filtering and Question Answering.
- **Enterprise applications are:** News Tracking, Customer Care, Data Cleaning, Classified Ads, Medical, Personal Information Management and Scientific Applications.
- **Web oriented applications are:** Citation Databases, Opinion Databases, Community Websites, Comparison Shopping Ad Placement on Web pages and Structured Web Searches.

III RELATED WORK

Data mining as a step of the knowledge discovery process, which can be called with different names, such as: knowledge extraction, information discovery, and data pattern processing. Knowledge discovery is the process of extracting useful information from data according to the user's needs. It is the extraction of interesting patterns from a set of facts in a database [11][13][14][15].

Text mining is combined with other disciplines, too, such as information extraction, data mining, machine learning, text classification, text clustering and natural language processing. Text analysis relates to extracting useful information and knowledge from semi structured data-text that is a combination of neither completely structured nor completely unstructured. Different information retrieval techniques, such as the text index method and the term weighting method have been developed to handle these data [11][13][20].

Intelligent Text Analysis is a process which analyses natural language in documents, e-mail messages and other free-form text by combining computing power with human-like intelligence. It tries to derive meaning from the words and sentences in order to classify documents, route messages appropriately, and to create summaries of content at the same time [11][13][16]. Some of the noteworthy contributions in the literature by various authors are listed below.

(a) Noteworthy Contributions

[24] They showcase the potential of text mining by extracting published protein± protein, disease±gene, and protein subcellular associations using a named entity recognition system, and quantitatively report on their accuracy using gold standard benchmark data sets.

[5] Has given technique for acquiring monolingual sentence level paraphrases from a corpus of temporally and topically clustered news articles composed of thousands of web-based information sources using unsupervised learning. The authors developed two techniques: (1) simple string edit distance, and (2) a heuristic strategy that pairs initial sentences.

[21] Used neural network to support because user query are most often in the form of a string. In that, it is a must to keep the record of matching string of the search key, to find out all the possibilities of the search key. The algorithm had the problem of ambiguity for the scenario where misplaced query string is same as the other (true) query.

[4] Removed the drawbacks of previous works which used monolingual parallel corpora to extract and generate paraphrases. They represented that this task was done using the concept of victimization bilingual parallel corpora, a way additional normally obtainable resource. Victimization alignment techniques from phrase based, mostly applied math computational linguistics. Paraphrases in one language are often known to employ a phrase in another language as a pivot. The tendency to outline a paraphrase likelihood that permits paraphrases extracted from a bilingual parallel corpus to be hierarchal victimization translation possibilities.

[3] Introduced an Open Information Extraction system from the web. Different from ancient Information Extraction systems that repeatedly incurred the value of corpus analysis with the naming of every new relation. Open Information Extraction, one-time relation discovery procedure permitted a user to call and explore relationships at interactive speeds. They also introduced TEXTRUNNER, a completely enforced Open IE system and demonstrated its ability to extract huge amounts of high-quality info from a 9 million website corpus.

[8] Focused on the use of Neuro-fuzzy based technology for improving the learning capability of the ranking function. Their methodology has been categorized in three important phases. Initially Training for the predicated class label and Calibration of model. Finally, the calculations of score using Byes scoring function. They use LETOR dataset and obtained improved result.

[10] Identified that information retrieval varies from machine to machine and it becomes difficult to integrate it meaningfully. As Search Engine produces hundreds of links, it becomes difficult to manage and identify the relevant one. They then developed the semantic web and common framework that could be reused and shared across the application. The work focused on concept link of pages rather than a hyperlink. Suggested to add Intelligence to the page using Metadata Triples and XML ontologies Tool.

[23] Present technique to the retrieval of unstructured data using English grammar semantic. They introduced an open information extraction

mechanism which is the foundation of question-answering system.

[22] described the several Rel-grams databases facilitates several tasks including: (1) Relational Language Models: it demonstrates a relational language model which encodes the conditional probability of relational tuple R, having observed R0 in the k previous tuples. It is used for discourse coherence, sentence order in summarization, etc. (2) Building event template: To cluster commonly co-occurring relational tuples and use them as the basis for open event templates. (3) Expectation-driven Extraction: the relational language model provides the probabilities output, which may be used to inform an information extractor. The Rel-grams database is freely available to the research community and it is a useful resource for a wide variety of NLP tasks.

IV PROPOSED TECHNIQUE

In the proposed neuro-fuzzy based model, fuzzy system has been selected to enhance the capability of the document retrieval process. To obtain this, fuzzy parameter or variables that can describe main features of the document are the term Frequency ratio (tfr), Document frequency ratio (dfr), Term frequency (tf), ratio of the number of search terms that occurs in one document to the length of the search term and Inverse document frequency (idf).

Let us assume that set of documents D = $\{d_1, d_2, d_3\ldots\ldots, d_N\}$ and set of term frequency T $=\{t_1, t_2, t_3\ldots\ldots, t_m\}$

Table 1
Term frequency matrix

DxT =

D → T ↓		**Documents**				
		d_1	d_2	d_3	…	d_N
Terms	t_1	tf_{11}	tf_{12}	tf_{13}	…	tf_{1N}
	t_2	tf_{21}	tf_{22}	tf_{23}	…	tf_{2N}
	t_3	tf_{31}	tf_{32}	tf_{133}	…	tf_{3N}
	. . .	. . .	. . .	. . .	. . .	. . .
	t_m	tf_{m1}	tf_{m2}	tf_{m3}	…	tf_{mN}

Where N is the total number of documents in the corpus and **m** is the total number of terms in the corpus.

tf_{ij} is the frequency of i^{th} term and j^{th} document.
tf*idf = tf_{term} x (N $_{number\ of\ documents}$ /df $_{term}$) and IDF = log $(1+D/df_i)$

To simplify the retrieval process it is recommended to pre-process the available document and it should be done prior to the start of the document retrieval process. One of the main pre-processing techniques is indexing. While indexing of document the input is all the documents in the corpus and the output is an index that has all the main terms available across all these documents.

Table 2
Pre-processing and indexing of document

Pre-processing (x)	**Indexing (corpus)**
{for each document (d) in x	{preprocessed-corpus=preprocessing(corpus)
{ d = Tokenize(d)	index=an empty file
d=Stop-word-removal(d)	for each document(d) in the preprocessed-corpus
d= Normalization(d)	for each term (t) in d
d=stemming(d)	{if (t in index) then append d to the document list of t
add d to the preprocessed-x}	else add a new index entry to index that has t and d
return (preprocessed-x) }	}return(index)}}

Pre-processing of document and indexing of the corpus is done according to the Table 2.

Table 3
Term occurrence in document

Term	Frequency of the term in Document		
	d_1	d_2	d_3
System	0	2	2
Industry	1	1	0
Information	3	2	1
ICT	1	0	0
and	2	4	2
Term	**Occurrence of the term in Document**		
System	d_2, d_3		
Industry	d_1, d_2		
Information	d_1, d_2, d_3		
ICT	d_1		
and	d_1, d_2, d_3		

.

Document d_1

Increasing demand of digital information along with higher productivity and better quality, on time delivery of updated information, the Industry is turning towards Information and Communication

Document d_2

Today Information Retrieval System (IRS) is a powerful tool and becoming more and more significant in various aspects of human life, especially in Industrial, Commercial and Scientific applications. As a result of scientific achievements and IT Industry development, the number of information retrieval system currently in use in corporate projects is increasing fast.

Document d_3

It has become a must to maintain the same pace for the IR systems too. However, IR has been evolving ever since and there has been an increasing interest in developing information extraction system still this area needs to be researched and fasten a lot.

The Table 3 shows the occurrence of term in available document d_1, d_2 and d_3 for example the term system is present in document d_2 and d_3. Further simplification of document can be possible that is determining the occurrence of the term in its associated document. Let us assume document d_1, d_2 and d_3 contains information are shown in paragraph. The term frequency given in Table 4 term frequency pattern can be useful while calculating the fuzzy score of query terms in the documents.

V CALCULATION OF SCORE

To calculate the score of documents lets interpret using the vector space model. To find documents with higher term frequency and relevancy with a query can be determined as shown in Table 5. In this Table weight of query terms have been calculated which is useful to decide the relevancy of a document for the selected query term. Weight calculation is done by multiplying term frequency with inverse document frequency (idf) to find the similarity to documents of the corpus.

Document d_3

It has become a must to maintain the same pace for the IR systems too. However, IR has been evolving ever since and there has been an increasing interest in developing information extraction system still this area needs to be researched and fasten a lot.

Table 5
Weight calculation of documents against query

Term Vector Model Based on $W_i = tf_i * idf_i$											
Query, Q: "Today Information Retrieval System is a powerful tool" DOCUMENT d_1 DOCUMENT d_2 DOCUMENT d_3 TOTAL DOCUMENT = 3, IDF = log (1+D/df_i)											
		Counts, tf_i						Weights, $W_i = tf_i * idf_i$			
Term	**Q**	**d_1**	**d_2**	**d_3**	**df_i**	**D/df_i**	**IDF**	**Q**	**d_1**	**d_2**	**d_3**
a	1	0	2	2	4	0.75	0.24	0.24	0	0.48	0.48
is	1	1	2	0	3	1	0.30	0.30	0.30	0.60	0
tool	1	0	1	0	1	3	0.60	0.60	0	0.60	0
today	1	0	1	0	1	3	0.60	0.60	0	0.60	0
system	1	0	2	2	4	0.75	0.24	0.24	0	0.48	0.48
powerful	1	0	1	0	1	3	0.60	0.60	0	0.60	0
retrieval	1	0	1	0	1	3	0.60	0.60	0	0.60	0
information	1	3	2	1	6	0.5	0.18	0.18	0.54	0.36	0.18
industry	0	1	1	0	2	1.5	0.40	0	0.40	0.40	0

In the proposed Neuro-fuzzy model for Information Retrieval, three fuzzy values have been used which are Low, Medium and High represent fuzzy linguistic values. Various membership functions were tested for fuzzy linguistic value to identify the relevancy in document for query term. Following are the fuzzy membership methods to find the relevancy of a document are

Cosine Similarity

(1) $$sim(d_j, d_k) = \frac{\vec{d}_j \cdot \vec{d}_k}{\left|\vec{d}_j\right|\left|\vec{d}_k\right|} = \frac{\sum_{i=1}^{n} w_{i,j} w_{i,k}}{\sqrt{\sum_{i=1}^{n} w_{i,j}^2}\sqrt{\sum_{i=1}^{n} w_{i,k}^2}}$$

S – Shaped Membership Function

(2) Relevancy
$$\begin{cases} 0 & \text{if } tfr \leq a \\ 2\,[(tfr-a)/(b-a)]^2 & \text{if } tfr \in [a, m] \\ 1-2\,[(tfr-b)/(b-a)]^2 & \text{if } tfr \in [m, b] \\ 1 & \text{if } tfr \geq b \end{cases}$$

L – Shaped Membership Function

(3) Relevancy
$$\begin{cases} 0 & \text{if } tfr \leq a \\ (tfr-a)/(b-a) & \text{if } a < tfr < b \\ 1 & \text{if } tfr \geq b \end{cases}$$

In the existing S and L shape membership function based implementation model value a =0.3 and b = 0.7 were used. The proposed Mamdani triangular membership function model to estimate the relevancy ratio of query term is -

Triangular – Shaped Membership Function

(4) Relevancy
$$\begin{cases} 0 & \text{if } tfr \leq a \\ (tfr-a)/(m-a) & \text{if } tfr\ \varepsilon\ [a, m] \\ (b-tfr)/(b-m) & \text{if } tfr\ \varepsilon\ [m, b] \\ 1 & \text{if } tfr \geq b \end{cases}$$

Linguistic variable range selected for Low from 0.0 to 0.1, Medium from 0.1 to 0.7 and to High above 0.7

$$\text{(5)} \quad \text{Relevancy} = \begin{cases} 0 & \text{if } tfr \leq 0.1 \\ (tfr-0.1)/(m-0.1) & \text{if } tfr \in [0.1, 0.4] \\ (0.7-tfr)/(0.7-m) & \text{if } tfr \in [0.4, 0.7] \\ 1 & \text{if } tfr \geq 0.7 \end{cases}$$

6. Experimental Setup

The proposed triangular membership model based information retrieval system has been implemented using python 2.7.3 with three standard dataset viz. Movie Review, Polarity and ACLIMDB (Large movie review dataset) are used to evaluate the proposed model and compare it with existing models.

VI DATASET DESCRIPTION

- **Movie Review Dataset [6]** It contains 1000 positive documents and 1000 negative documents.
- **Polarity Dataset [6]** It contains 700 positive documents and 700 negative documents.
- **ACLIMDB Dataset [2]** It contains 25000 training and 25000 testing documents, the combination of this dataset contains 50000 documents in which 25000 documents are of positive category and 25000 are of negative category.

The proposed model has calculated the term score on the basis of fuzzy inference rule which defined in a triangular membership function and retrieve the similar documents list which are most relevance to the query.

In experimental work both categories of query have been tested and retrieved the documents containing positive and negative class label.

Table 6
Positive query information

Search query=**('liked', 'movie', 'good', 'beautiful') -** positive query in movie review dataset
tf-idf=term frequency X log(N/document frequency of term)
Fuzzy Triangular membership function is used

Table 7
Retrieved documents against positive query term

Term	Movie review documents	Tf-idf weight	Fuzzy score
Positive Class Label			
liked	cv746_10147.txt	0	0
movie	cv746_10147.txt	1.767451736	1
good	cv746_10147.txt	1.578048643	1
beautiful	cv746_10147.txt	0	0
		Fuzzy Score =	2
liked	cv089_11418.txt	0	0
movie	cv089_11418.txt	0	0
good	cv089_11418.txt	0.526016214	0.579945953
beautiful	cv089_11418.txt	0	0
		Fuzzy Score =	0.579945953
liked	cv723_8648.txt	0	0
movie	cv723_8648.txt	0.252493105	0.50831035
good	cv723_8648.txt	1.052032428	1
beautiful	cv723_8648.txt	0	0

		Fuzzy Score =	1.50831035
Negative Class Label			
liked	cv011_13044.txt	0	0
movie	cv011_13044.txt	0.757479315	1
good	cv011_13044.txt	0.526016214	0.579945953
beautiful	cv011_13044.txt	0	0
		Fuzzy Score =	1.579945953
liked	cv007_4992.txt	0	0
movie	cv007_4992.txt	1.767451736	1
good	cv007_4992.txt	0	0
beautiful	cv007_4992.txt	0	0
		Fuzzy Score =	1
Term	**Movie review documents**	**Tf-idf weight**	**Fuzzy score**
liked	cv042_11927.txt	0	0
movie	cv042_11927.txt	0.757479315	1
good	cv042_11927.txt	0	0
beautiful	cv042_11927.txt	0	0
		Fuzzy Score =	1

Table 8
Negative query information

Search query=(**'bad', 'movie', 'worst', 'terrible' ,'awful') –** Negative query in movie review dataset
tf-idf=term frequency X log(N/document frequency of term)
Fuzzy Triangular membership function is used

Table 9
Retrieved documents against negative query term

Term	**Movie review documents**	**Tf-idf weight**	**Fuzzy score**
Positive Class Label			
Bad	cv090_0042.txt	0	0
movie	cv090_0042.txt	0.50498621	0.645004596
worst	cv090_0042.txt	0	0
terrible	cv090_0042.txt	0	0
awful	cv090_0042.txt	0	0
		Fuzzy Score =	0.645004596
Bad	cv003_11664.txt	0	0
movie	cv003_11664.txt	1.00997242	1
worst	cv003_11664.txt	0	0
terrible	cv003_11664.txt	0	0
awful	cv003_11664.txt	0	0
		Fuzzy Score =	1
Bad	cv001_18431.txt	0.950476665	1
movie	cv001_18431.txt	0.50498621	0.650045966
worst	cv001_18431.txt	0	0
terrible	cv001_18431.txt	0	0

awful	cv001_18431.txt	0	0
		Fuzzy Score =	0.650045966
Negative Class Label			
bad	cv008_29326.txt	0.950476665	1
movie	cv008_29326.txt	0	0
worst	cv008_29326.txt	0	0
terrible	cv008_29326.txt	0	0
awful	cv008_29326.txt	0	0
		Fuzzy Score =	1
bad	cv718_12227.txt	0	0
movie	cv718_12227.txt	1.514958631	1
worst	cv718_12227.txt	2.127033024	1
terrible	cv718_12227.txt	0	0
awful	cv718_12227.txt	2.793291305	1
		Fuzzy Score	3
bad	cv698_16930.txt	0.950476665	1
movie	cv698_16930.txt	1.00997242	1
worst	cv698_16930.txt	0	0
terrible	cv698_16930.txt	0	0
awful	cv698_16930.txt	0	0
		Fuzzy Score =	2

List of retrieved documents is shown in Table 7 and Table 9. Three positive and three negative class labels have been given for both categories of query terms.

VII RESULT ANALYSIS

Observations have been made on the set experiment of proposed Neuro-Fuzzy based IR Model and evaluation of the query has been done at the same time. The query term is tested in three datasets and computed performance of methods using confusion matrix.

In the Table 10 description of experimental data is given to verify the correctness (relevancy) of a query term. In the implementation section both (positive and negative) category query term is tested and computed weight as well as fuzzy score of each term. On the basis of the fuzzy score relevancy of the document is decided and classifies the query label (positive or negative) document label (positive or negative) in certain predicated class.

Table 10
Analysis of query

Query Term	Label of Query	Document Name	Level of Document	Fuzzy Score	Relevancy	Predicated Class
('liked', 'movie', 'good', 'beautiful')	Positive	cv746_10147.txt	Positive	2	Relevance	TP
('liked', 'movie', 'good', 'beautiful')	Positive	cv089_11418.txt	Negative	0.58	Not Relevance	FN
('bad', 'movie', 'worst', 'terrible' ,'awful')	Negative	cv001_18431.txt	Positive	1.65	Relevance	FP
('bad', 'movie', 'worst', 'terrible' ,'awful')	Negative	cv090_0042.txt	Negative	0.64	Not Relevance	TN

Table 11
Binary pattern to representation predicated class of confusion matrix

Query Term	Score Relevance	Class Label	Predicated Class
0	0	0	FP
0	0	1	FP
0	1	0	TN
0	1	1	FP
1	0	0	FN
1	0	1	FN
1	1	0	FN
1	1	1	TP

Table 11. represents the binary pattern of query processed in Table 10. True (or positive) and false (or negative) is represented by 1 and 0 respectively.

A document relevancy range of triangular membership function is defined as

Not relevance if $tfr \leq 0.2$
(6) Query Relevancy
Relevance if $tfr \geq 0.7$

Table 12
Confusion matrix result for movie review dataset

Method	Computed Data			
	TN	FP	FN	TP
Cosine Similarity	925	75	56	944
S Shape	969	31	63	937
L Shape	940	60	50	950
Triangular	980	20	10	990

Table 13
Confusion matrix result for polarity dataset

Method	Computed Data			
	TN	FP	FN	TP
Cosine Similarity	605	95	108	592
S Shape	651	49	86	614
L Shape	647	53	72	628
Triangular	648	52	67	633

Table 14
Confusion matrix result for ACLIMDB dataset

Method	Computed Data			
	TN	FP	FN	TP
Cosine Similarity	21986	3014	4089	20911
S Shape	21789	3211	3562	21438
L Shape	22112	2885	3352	21648
Triangular	22145	2855	2888	22112

Table 15
Performance comparison

Method → Dataset ↓	Cosine Similarity			S Shaped			L Shaped			Proposed (Triangular)		
	Accuracy	Precision	Recall	Accuracy	Precision	Recal l	Accuracy	Precision	Recal l	Accuracy	Precision	Recall
Movie Review	93.4	92.6	94.4	95.3	96.7	93.7	94.5	94.1	95.0	98.5	98.0	99.0
Polarity	85.5	86.2	84.6	91.0	92.2	89.7	90.4	92.6	87.7	91.5	92.4	90.4
ACLIMDB	85.7	87.4	83.6	87.5	88.2	86.6	86.5	86.9	86.4	88.5	88.7	88.4

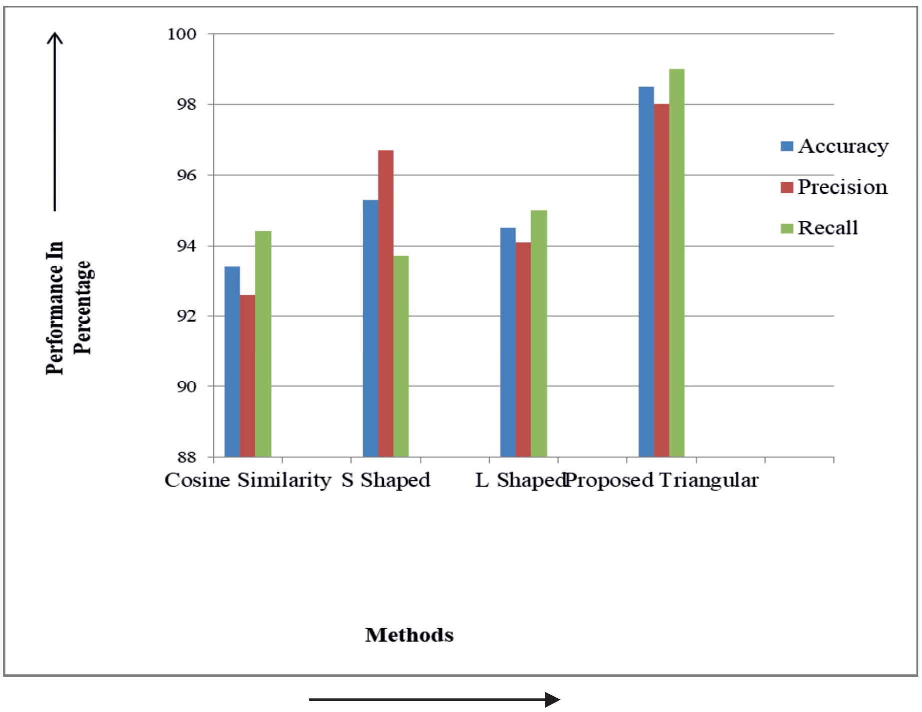

Fig 2. Result Comparisons for movie review dataset

The result of experimental work is shown in Table 15. The experimental work has been performed using various categories of standard dataset. The result computed for the proposed and existing methods is given. The proposed work performance is compared and shown in the Fig. 2, Fig. 3 and Fig. 4 consecutively.

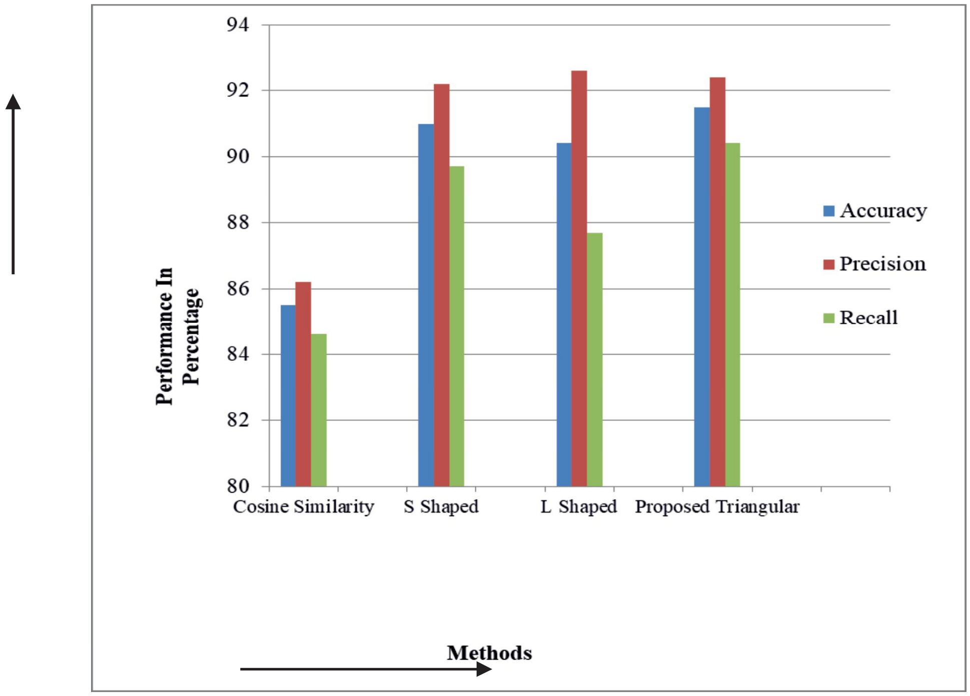

Fig 3. Result Comparisons for polarity dataset

VIII CONCLUSION

The performance parameters - accuracy, precision (positive predicated documents against query) and recall (relevant documents that have been retrieved over total relevant documents) have been compared with the cosine similarity, S Shaped, L Shaped and the Proposed Method (Triangular). The proposed method performance is higher in all three cases. Three standard text datasets, viz. Movie Review, Polarity and ACLIMDB (Large movie review) are used to evaluate the comparative models.

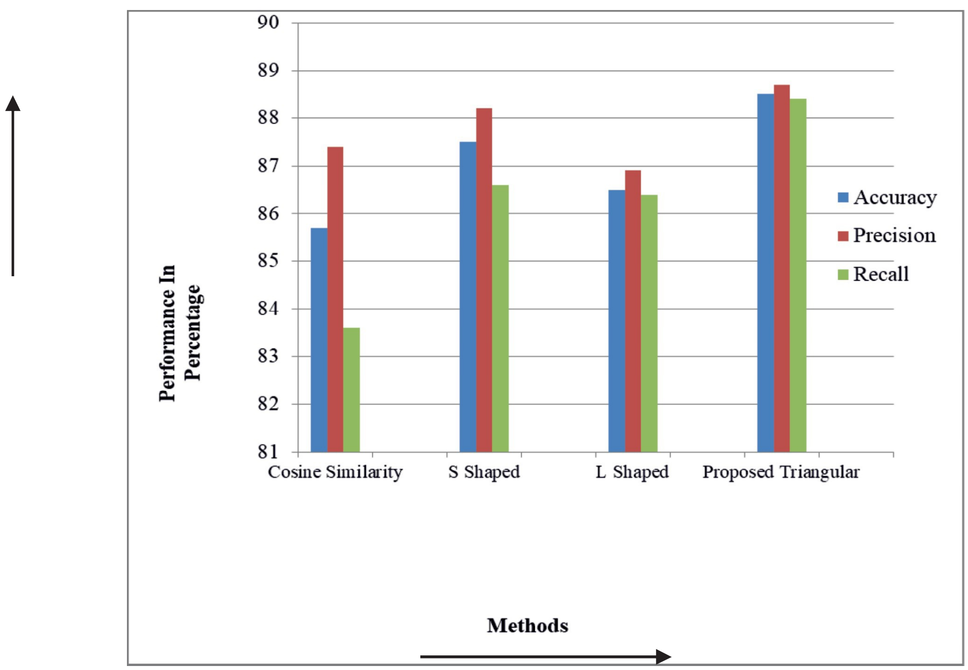

Fig 4. Result comparisons for ACLIMDB dataset

REFERENCES

[1] A. Fader, Zettlemoyer L. and Etzioni Oren. Paraphrase-Driven Learning for Open Question Answering. – Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, Sofia, Bulgaria, 2013, 1608-1618.

[2] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng and Christopher Potts. Learning Word Vectors for Sentiment Analysis. – The 49th Annual Meeting of the Association for Computational Linguistics, 2011.

[3] Banko Michele, Etzioni Oren, Soderland Stephen and Weld Daniel S. Open Information Extraction from the Web. – ACM, Vol. 51, 2008, No. 2, 68-74.

[4] Bannard Colin, Burch Chris Callison. Paraphrasing with Bilingual Parallel Corpora. School of Informatics University of Edinburgh, 2 Buccleuch Place Edinburgh, 2005.

[5] Bill Dolanet, Chris Quirk and Chris Brockett. Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources. – In Proceedings of the 20th International Conference on Computational Linguistics (Coling‘04), Association for Computational Linguistics, Stroudsburg, PA, USA Article 350, 2004.

[6] Bo Pang, Lillian Lee. Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with respect to Rating Scales. – Proceedings of ACL, 2005.

[7] Bo Pang, Lillian Lee et al. Opinion Mining and Sentiment Analysis. – Foundations and Trends in Information Retrieval, Vol. 2, 2008, No. 1–2, 1–135.

[8] Busa Fekete Robert, KeglBalazs, Tamas Elteto and Szarvas. A Robust Ranking Methodology Based on Diverse Calibration of Adaboost. – Joint European Conference on Machine Learning and Knowledge Discovery in Databases ECML PKDD, Vol. 1, 2011, 263-279.

[9] Christopher D Manning, Hinrich Schiitze. Foundations of Statistical Natural Language Processing, 2nd Edition, MIT Press, 1999, ISBN:. 0-262-13360-l.

[10] Dhingra Vandana, Bhatia Komal Kumar. Towards Intelligent Information Retrieval on Web. – International Journal on Computer Science and Engineering (IJCSE), Vol. 3, 2011, No.4, 1721-1726.

[11] F. Sebastiani. Machine Learning in Automated Text Classification. – ACM Computing Surveys, Vol. 34, 2002, No. 1, 1–47.

[12] George Forman. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. – Journal of Machine Learning Research, 2003, 1289–1305.

[13] George Forman. A pitfall and solution in multi-class feature selection for text classification. – 21st International Conference on Machine Learning, Banff, Alberta, Canada, 2004.

[14] George Forman. Chapter: Feature Selection for Text Classification Book: Computational Methods of Feature Selection Chapman and Hall/CRC Press 16, ISBN:. 9-781-584888-789. 2007.

[15] George Forman. BNS feature scaling: An Improved Representation over tf-idf for SVM Text Classification. – Proceedings of the 17th ACM Conference on Information and Knowledge Management, 2008, 263–270.

[16] Gerlof Bouma. Normalized (pointwise) Mutual Information in Collocation Extraction. – Proceedings of GSCL, 2009, 31–40.

[17] Gupta Sachin, Aggarwal Ankit. Study of Search Engine Optimization. – International Journal Research in Engineering & Applied Sciences, Vol. 2, 2012, Issue 2, 1529-1536.

[18] Jai Manral, Mohammed Alamgir Hossain. An Innovative Approach for online Meta Search Engine Optimization. – Computational Intelligence Research Group, The 6th Conference on Software, Knowledge, Information Management and Applications, Chengdu, China, 2012, 1-7.

[19] K Bretonnel Cohen, Lawrence Hunter. Getting Started in Text Mining. PLoS Computational Biology, Vol. 4, 2008, No. 1, 1–4.

[20] Luigi Galavotti, Fabrizio Sebastiani and Maria Simi. Experiments on the use of Feature Selection and Negative Evidence in Automated Text Categorization. – Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Verlag, Vol. 1923, 2000, 59–68, ISBN:.3-540-41023-6.

[21] Murthy SGK, Biswas R N. A Fuzzy Logic Based Search Engine Technique for Digital Liabraries. – Desidoc Bulletin of Information Technology, Vol. 24, 2004, No. 6, 3-9.

[22] Niranjan Balasubramanian, Stephen Soderland, Mausam and Oren Etzioni. Rel-Grams: A Probabilistic Model of Relations in Text. – Proceeding AKBC-WEKEX '12 Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, Montreal, Canada — June 07 – 08, 2012, 101-105.

[23] Etzioni Oren, A. Fader, Janara Christensen, Stephen Soderl and and Mausam. Open Information Extraction: The Second Generation. International Joint Conference on Artificial Intelligence, 2011, 1-8.

[24] Westergaard, D., Stærfeldt, H. H., Tønsberg, C., Jensen, L. J., & Brunak, S. A comprehensive and quantitative comparison of text-mining in 15 million full-text articles versus their corresponding abstracts. – P L o S Computational Biology,2018, 14(2).