ISSN: 0976-3104
SPECIAL ISSUE: COMPUTER SCIENCE

IIOAB JOURNAL
ISSN: 0976-3104

ARTICLE

AN APPROACH FOR EFFICIENT RANKING OF XML DOCUMENTS USING USING BPN BASED RANN

Mary Posonia A[1*], Vigneshwari S[1]*, Jyothi V.L[2]

[1]*Dept. of Computer Science & Engineering, Sathyabama Institute of Science & Technology, Tamilnadu, INDIA*

[2]*Dept. of Computer Science & Engineering, Jeppiaar Engineering College, Tamilnadu, INDIA*

ABSTRACT

There is a semantic gap between the implications of the keywords in the recovered documents and the implications of the terms utilized as a part of users' queries. Ranking algorithms are an important step in search engines so that the user could retrieve the pages most relevant to the query. The proposed novel algorithm, Back Propagation Network(BPN) based Ranking algorithm using Neural Networks (RANN) system is used to rank the XML documents, both in a time efficient and cost efficient manner. The existing XML based document indexing approaches focus only on partial input queries which lead to irrelevancy problem. To overcome this issue, efficient document retrieval is achieved with the help of BPN trained RANN and it is proven in the results. The overall performance of RANN is measured in comparison with Vector Space Model (VSM). The results of the devised approach show remarkable improvement in the performance with the use of synthetic records and benchmark dataset with an overall improvement of 7% raise in Precision, Recall and F-Measure rates when compared to the existing VSM based approach.

THE IIOAB JOURNAL

KEY WORDS
BPN, RANN, XML, VSM, Document Retrieval

Received: 20 Nov 2017
Accepted: 22 Dec 2017
Published: 5 Jan 2018

***Corresponding Author**
Email:
vigneshwari.cse@sathyabama.ac.in
Tel.: +91-9941571360

INTRODUCTION

In the usual web based information retrieval frameworks, the user's needs are not met as they are ranked in view of the conventional string matching approach of the user's query. This has led to a semantic gap between the implications of the keywords in the recovered records and the implications of the terms utilized as a part of user's queries. With the rapid growth of the World Wide Web there comes the need for a fast and accurate way to retrieve the information required, which is made possible with the help of Search engines. Ranking algorithms are essential for the users to retrieve the pages that are most relevant to the query.

Information Retrieval (IR) for XML has increased noteworthy consideration and has rose as one of the research subjects that have been examined by Keyword researchers and Query researchers. The objective of this research is to apply the IR utilizing the Ranking Algorithm of Neural Network (RANN) Model on characteristic XML documents to solve the issues in document retrieval. In this paper, IR using RANN is applied for document ranking and retrieval.

This paper is organized as follows: related work, followed by materials and methods discussing about the implementation of BPN based RANN model which is followed by the evaluation of results and conclusion.

RELATED WORK

XML ranking can be supported through the "inverted element, frequency" and "weighted term frequency" as proposed by Chen et al [1].Here, the weight of the term is dependent on the location and frequency in an XML element, and its popularity is also known among the similar elements in an XML dataset. Chen's idea is followed in this paper to find the term-document index of XML documents.

A delay may occur before an information extraction is available based on the updated documents. Shortening of the delay as proposed in [2-6], using a method which recycles the intermediate results of the snapshots taken in the past.

A Vector Space Model was proposed in [7-12] which was known as the orthogonal factorization matrix, and could be used for the retrieval of information from a large database. The VSM model has been used for comparison with the proposed model in our BPN based RANN model.

The authors in [13-20,23,24] presented an information retrieval technique using Vector Space Model (VSM). Firstly, the similarity scores are computed through the use of a weighted average for every item. Later, the cosine measure helps to compute the similarity measure and determines the document vector and the query vector angle, on the basis of geometry. The use of IR and Neural Network (NN) improved the performance of the information retrieval. However, there is a drawback with respect to the IR systems, in conjunction with natural languages, especially of Arabic type. Hence, to overcome those drawbacks a feed forward training Network like BPN is necessary for yielding accuracy which has been proposed in our currnt paper.

A hybrid model and its application were presented by Karegowda et al. [21] and Shanthi et al.[22]. Here, the context of Artificial Neural Networks on the subject of Evolving Connection Weights was applied to the prediction of stroke disease; a comparison was made between the desired and real output of the Hybrid ANN-GA and ANN. The accuracy in classification with respect to the surfaces was found to be improved in

ISSN: 0976-3104

this case. Hence it is inferred from the literature survey that the results of experiments in the existing literature indicate an improvement in the performance upon the utilization of Neural Networks.

MATERIALS AND METHODS

General neural network architecture for document ranking

Data retrieval utilizing the RANN Model for XML Document is a strategy to acquire significant measures between a query and the documents recovered. The model consists of three layers; Query Terms Layer, Documents Terms Layer and Documents Layer. Cosine similarity measure is utilized as a part of the RANN model to ascertain the similarity between document query vectors.

Enhanced RANN based information retrieval system

The Back Propagation Network (BPN) follows the delta learning rule of Neural Networks in order to reduce the error by weight adjustments in the hidden and output layers. BPN is preferred in the current paper since the sigmoid function of BPN deals with non-linear models like XML tree structures [Fig. 1].

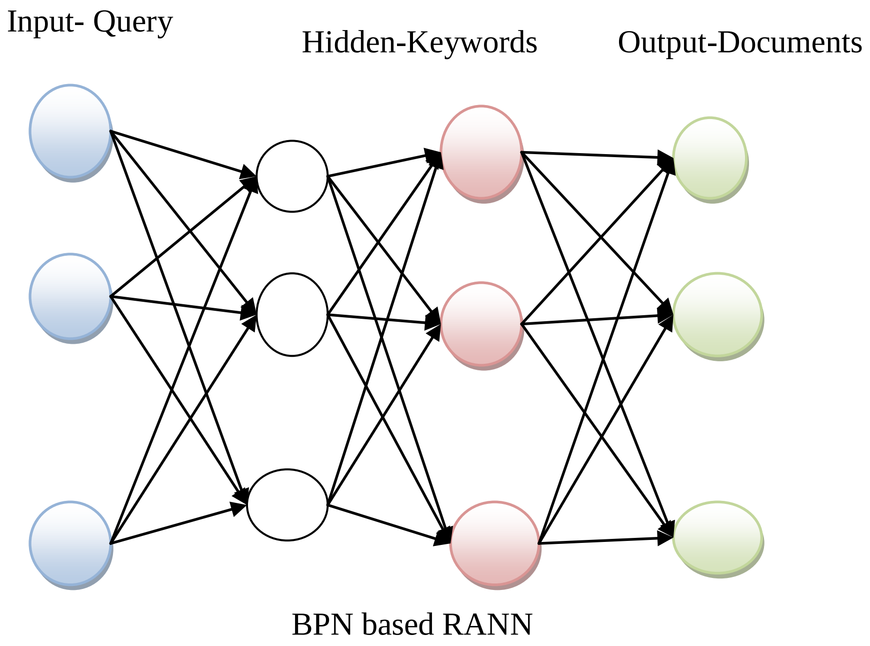

Fig. 1: Enhanced RANN based information retrieval model.

The input interface is added, which allows the user to enter a query. The information retrieval system is then upgraded which empowers the user to locate the significant documents. Towards the end, an output interface is created, which sorts the significant documents and sends them to the user as a result.

When it is necessary to give a query input involving at least two words, it is important to include more input neuron bunches into the first neural network where each gathering depicts a single word. At that point each word is independently mapped cognizant to the keywords of each query.

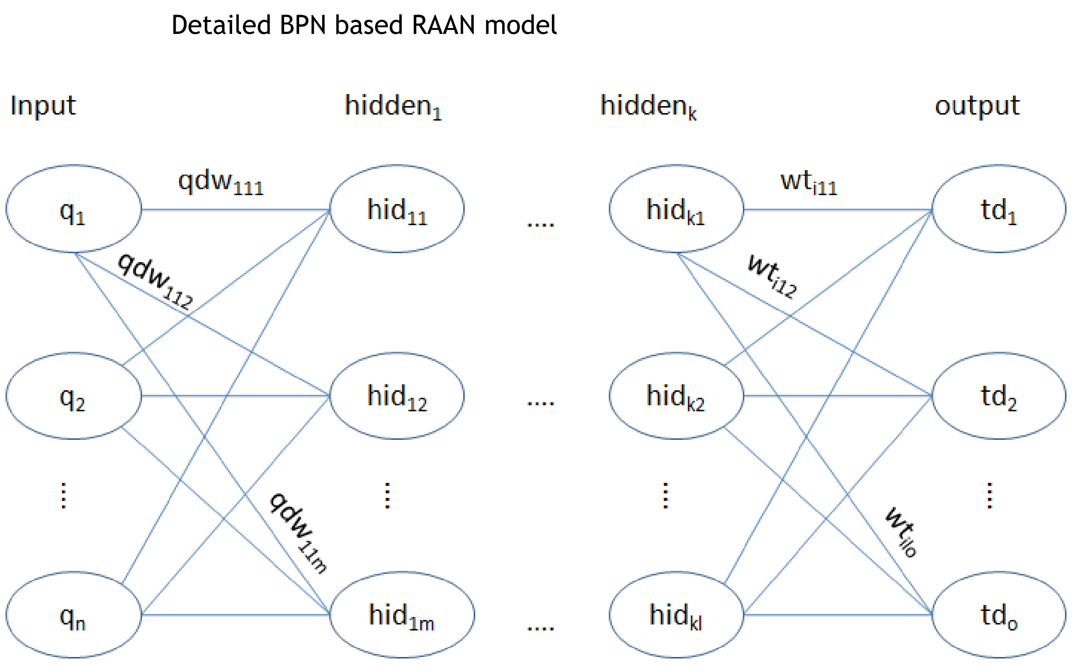

Fig. 2: Detailed BPN based RANN model

[Fig. 2] describes the detailed framework for document retrieval based on BPN algorithm. Here q_i represents the set of input queries i = 1 to n, h_id_{ij} represents the jth node of the ith hidden layer, i = 1 to m and j = k to m. For binary inputs a threshold of 0.5 is set. If the tf-idf measure is greater than 0.5 then the binary input is 1 and 0 otherwise.

And, wt_{ijk} represents the hidden output weights for 'o' output documents. The resultant documents are Trained Document (TD) indexed ones. The TD index is calculated based on the precision rate of the documents which are all True-Positive (i.e based on relevancy in the retrieved documents). For calculating query document similarity cosine similarity is used, for ranking and for
Index, B+ Tree algorithm is used [15].

(1)

Where TP represents True Positiveness and FP represents False Positiveness of the relevant retrieved documents.

BPN based RANN algorithm

Input : 'n' number of input queries
'm' number of hidden nodes in each hidden layer k
'o' number of retrieved documents as output layer
Output: TD indexed documents
Read 'n' number of input nodes
Read 'h' number of hidden nodes
Read 'm' number of output nodes
Step 1: Read the input vector of queries q_i
Step 2: Read the output vector tod_o (Desired output documents with a set of pre-trained documents from the training knowledge base)

Step 3: Read the input hidden weights qdw_{ijk}, where qdw is the query document weight which calculated based on the tf-idf measure.
Step 4: Read the output hidden weights wt_{ijk}
Step 5: Calculate $netvalh_{jk}$ (net value in hidden layer)

$$netvalh[jk] = \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} * hid_{ijk} \; \forall \; k = 1to\; l$$

Step 6: Calculate the f($netvalh_{jk}$) : '$netwalh_{jk}$' in hidden layer (sigmoidal function)

$$\mathrm{f}(\mathrm{netvalh}_{\mathrm{jk}}) = \frac{1}{(1 + e^{-\mathrm{f}(\mathrm{netvalh}_{\mathrm{jk}})})}$$

Step 7: Calculate the net td-output: "td_o "

$$td_{o=} \sum_{i=1}^{n} \sum_{j=1}^{m} td_{ij} * hid_{ijk} \; \forall \; k = 1to\; o$$

Step 8: Calculate the actual output ao_o

IIOAB JOURNAL
ISSN: 0976-3104

$$ao_o = \frac{1}{(1+e^{-td_o})}$$

Step 9: Calculate error in output layer '$ertd_o$' which is the difference between Desired output and Actual output(Delta rule)

$$ertd_o = tod_o - td_o$$

Step 10: Calculate error in hidden layer 'errhid$_{jko}$'

$$\mathrm{errhid_{jko}} = \sum_{k=1}^{n} \mathrm{td_o} \; * \; hid_{jko} \; \forall \; o = 1 to\; o \; and \; j = 1 \; to \; m$$

Step 11: Calculate the adjusted weights (nwo$_{jko}$) for the hidden output layer 'nwo$_{jko}$'
For j=1 to h
For k=1 to m

$$nwt_{jko} = \; wt_{jko} + (\eta * td_o * \mathrm{netvalh_{jk}})$$

Step 12: Adjusted weight for input hidden layer is calculated as follows: 'awh$_{ijk}$'

$$awh_{ijk} = \; wt_{ijk} + (\eta * q_i * \mathrm{netvalh_{jk}})$$

The new weight obtained for hidden output layer is nwt_{jko} and the new weight obtained for input hidden layer is awh_{ijk}

Step 13: The earlier weights are replaced with the adjusted weights in both the hidden-input and output hidden layers and Step 5 to Step 13are continued until saturation is reached.

Let qi be the input query. Based on the similarity of query with the documents the input values are designed based on Cosine similarity.

$$CosRann(doc, query) = \frac{\sum doc_{ij} \times query_j}{\sum doc_{ij}^2 \times \sum query_j^2} \quad (2)$$

The input weights are determined with the tf-idf value of the input queries with that of the documents mapped.

$$qdw_{ijk} = TermFrequency_i * InverseDocumentFrequency_i \quad (3)$$

$$\mathrm{TermFrequency_i} = \frac{Number\ of\ Document\ Terms\ which\ are\ similar}{Total\ number\ of\ Document\ Terms} \quad (4)$$

$$\mathrm{InverseDocumentFrequency_i} = \log\left(\frac{\text{Number of documents}}{\mathrm{DocumentFrequency_i}}\right) \quad (5)$$

The hidden weight calculation is based on the approximated weights based on the input queries. The documents are trained and the trained document index is calculated as TD index (Equation (5.1)).

Based on the TD index, the documents are sorted based on B+ Tree algorithm[17] and ranked based on the precision value[25]. The time complexity of the devised BPN based RANN algorithm is $O(qdw^k)$ where qdw is the query document weight for k number of hidden layers. Total of 10 sample queries were utilized with total 1800 documents presented in the [Table 1].

Table 1: TD-index calculation table

Queries	True Positiveness	True Negativeness	False Positiveness	False Negativeness	TD index
Q1	980	890	51	43	0.95
Q2	999	865	88	87	0.92
Q3	945	903	243	117	0.80
Q4	1008	976	157	257	0.87
Q5	899	897	47	52	0.95
Q6	907	878	50	80	0.95
Q7	956	856	44	46	0.96
Q8	1013	834	66	68	0.94
Q9	989	912	145	120	0.87
Q10	976	908	67	66	0.94

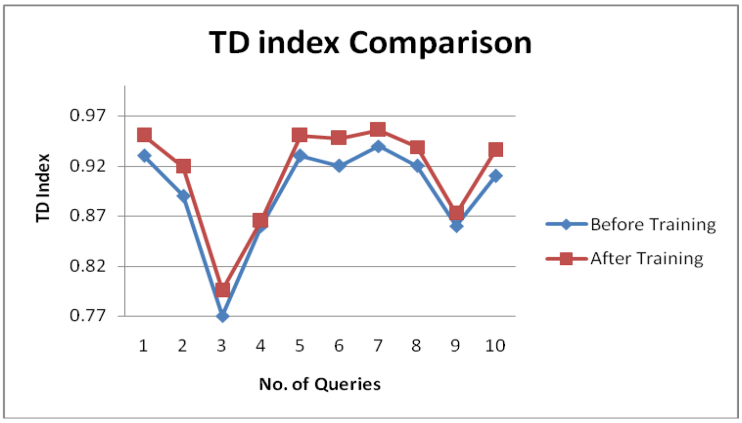

Fig. 3: TD index comparison before training and after training

…….

[Fig. 3] shows a significant improvement of the TD index after training with BPN based RANN with an average improvement of 2%.

Vector space model concepts overview

Vector Space Model (VSM) is a method used to interact with documents and queries as vectors in multidimensional space, whose measurements are the terms which are utilized to build an index to interact with the documents [7]. It is the most widely utilized procedure for information retrieval because of its effortlessness; effectiveness and pertinence over substantial document accumulations. The viability of the VSM depends generally on the term weighting connected to the term of the document vectors. The three phases of VSM are (VSM 2017):

- Document Term extraction
- Document Term weighting
- Ranking of documents based on the query-document similarity measure

Table 2: Difference between VSM and BPN-RANN

VSM	BPN-RANN
Linear Model	Non-linear model
Weights are not binary	Binary weights are allowed
Cosine similarity is followed	Cosine similarity is followed
Allows partial matching	Allows trained document matching based on TD index

RESULTS

Comparison of precision, recall, F-measure between neural network model and vector model

This study was done with the following configuration with Windows 7 operating system, Intel Pentium(R) processor, CPU G2020 with processor speed of 2.90 GHz. The server pre-processes the data and stores it in the database. This process generates keywords, indexes XML documents and rank the documents based on the devised RANN algorithm. The datasets used for comparison is Sigmoid dataset which is freely downloadable from http://aiweb.cs.washington.edu/research/projects/xmltk/xmldata/data/sigmod-record/SigmodRecord.xml.

The performance of RANN is compared with the traditional VSM techniques, and the results prove that the retrieval rate and relevancy of documents using RANN is more efficient with that of VSM is shown in the [Table 3].

Table 3: Comparison of RANN and VSM for Precision, Recall and F-measure

Model	Query	Number of Documents	Precision	Recall	F-Measure
RANN	Q1	150	89.84%	98%	93.74%
	Q2	300	88.78%	97.20%	92.80%
	Q3	450	85.04%	92.16	88.46%
	Q4	600	84.04%	92.16%	87.89%
	Q5	750	86.04%	92.16%	88.99%
VSM	Q1	150	79%	85%	81.89%
	Q2	300	76.00%	86.15%	80.76%
	Q3	450	75.71%	87.50%	81.18%
	Q4	600	75.71%	87.50%	81.18%
	Q5	750	75.71%	87.50%	81.18%

[Table 3], illustrates the Precision and Recall values of Ranking Algorithm of Neural Network and Vector Space Model keeps on decreasing as the total number of documents increases. The effect is caused by the keyword expansion from the ranking process. This causes the results to be error bounded which leads to inaccuracies in the user search. The system search is relevant even though the accuracy is not excelling. Also, the F-Measure values of Ranking Algorithm of Neural Network and Vector Space Model decreases when total number of documents increases. These issues are caused by the keyword expansion from the ranking process, thus giving results which are not accurate with respect to the user search but relevant to the systematic search. However, it is evident from [Table 3] that the Precision-recall and F value of Ranking Algorithm of Neural Network is higher when compared to the Vector Space Model. The results showed increased precision, recall, accuracy and F-measure rates and reduced response time and memory utilization with RANN is illustrated in [Table 4].

Table 4: Performance analysis of RANN compared with VSM and ORA in terms of Response Time and Accuracy

	Response Time		Accuracy	
Query	RANN	VSM	VSM	RANN
Q1	0.0061	0.00695	0.006919	0.006836
Q2	0.0054	0.0064	0.006133	0.006094
Q3	0.0071	0.00795	0.007802	0.00793
Q4	0.0064	0.0069	0.006422	0.005664
Q5	0.0074	0.0075	0.007043	0.00558

CONCLUSION

Urbanization In this paper, a novel algorithm called BPN based RANN algorithm for ranking the retrieved documents is introduced. It is a non linear model and therefore binary weights are allowed which is important for performing ranking on the dynamic incoming real time data. The proposed work has been compared with existing VSM model based on precision, recall and f-measure percentages and the results prove that the proposed algorithm shows an average of 7% improvement in the performance when compared with the existing VSM based approach.

CONFLICT OF INTEREST

The authors have no conflict of interest regarding this manuscript.

ACKNOWLEDGEMENTS

The authors would like to thank Sathyabama Institute of Science & Technology and its School of Computing for providing this great opportunity to do our research in a successful manner.

FINANCIAL DISCLOSURE

None

REFERENCES

[1] Cekstere Chen Fei, Xixuan Feng, Christopher Re and Min Wang [2012] Optimizing Statistical Information Extraction Programs over Evolving Text. In Data Engineering (ICDE), 28th International Conference on IEEE, 870-881.

[2] Neumann Thomas and Gerhard Weikum [2010] x-RDF-3X: Fast Querying, High Update Rates, and Consistency for Rdf Databases. Proceedings of the VLDB Endowment, 3(2):256-263.

[3] Ren Chenghui, Eric Lo, Ben Kao, Xinjie Zhu, Reynold Chen [2011] On Querying Historical Evolving Graph Sequences. Proceedings of the VLDB Endowment, 4(11):726-737.

[4] Tomasic Anthony, Hector Garcia-Molina and Kurt Shoens [1994] Incremental Updates of Inverted Lists for Text Document Retrieval. ACM, 23(2):289-300.

[5] Margaritis Giorgos and Stergios V Anastasiadis [2009] Low-Cost Management of Inverted Files for Online Full-Text Search. In Proceedings of the 18th ACM conference on Information and knowledge management, 455-464.

[6] Keyaki Atsushi, Jun Miyazaki, Kenji Hatano, Goshiro Yamamoto, Takafumi Taketomi and Hirokazu Kato [2012] Fast and Incremental Indexing in Effective and Efficient Xml Element Retrieval Systems. In Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services, ACM, 157-166.

[7] Isara Nakavisute and Kanyarat Sriwisathiyakun [2015] Optimizing Information Retrieval (Ir) Time with Doubly Linked List and Binary Search Tree (BST). ACM Transaction, 3(12), 1-9.

[8] Akewaranukulsiri P, Prompoon N. [2013] Semantic and Cross-Language Information Retrieval for Thai Herbs and Modern Medicine. In Information Science and Applications (ICISA), 2013 International Conference on IEEE, 1-5.

[9] Hassani K, Lee WS. [2015] Adaptive Animation Generation using Web Content Mining. IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS), 1-8.

[10] Zhang Liang, Bingpeng Ma, Guorong Li, Qingming Huang and Qi Tian [2017] Cross-Modal Retrieval using Multiordered Discriminative Structured Subspace Learning. IEEE Transactions on Multimedia, 19(6): 1220-1233.

[11] Lee C, Kawahara T. [2012] Hybrid Vector Space Model for Flexible Voice Search. Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), IEEE, 1-4.

[12] Deng S, Gao K, Du C, Ma W, Long G, Li Y. [2016] Online Variational Bayesian Support Vector Regression. International Joint Conference on Neural Networks (IJCNN), IEEE, 3950-3957.

[13] Ogheneovo EE, Japheth RB. [2016] Application of Vector Space Model to Query Ranking and Information Retrieval. International Journal of Advanced Research in Computer Science and Software Engineering, 6(5):42- 47.

[14] Wedyan Mohammad, Basim Alhadidi, Adnan Alrabea [2012] The Effect of using a Thesaurus in Arabic Information Retrieval System. Int. J Computer Science, 9:431-435.

[15] Zhao Chongchong, Zhiqiang Zhang, Xiaoqin Xie, .Tingting Liang [2010] A New Keywords Method to Improve Web Search. In High Performance Computing and Communications (HPCC), 12th International Conference on IEEE, 477-484.

[16] Mokriš Igor, Lenka Skovajsová. [2005] Development of Neural Network Information Retrieval System from Text Documents. Acta Electrotechnica et Informatica, 5(3):357-366.

[17] Scarselli Franco, Sweah Liang Yong, Markus Hagenbuchner ,Ah Chung Tsoi. [2005] Adaptive Page Ranking with Neural Networks. In Special interest tracks and posters of the 14th international conference on World Wide Web, ACM, 936-937.

[18] Dharmistha Vishwakarma, D. [2012] Genetic Algorithm Based Weights Optimization of Artificial Neural Network. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 1(3):206-211.

[19] Venkatesan P, Premalatha V. [2012] Genetic-Neuro Approach for Disease Classification. International Journal of Science and Technology, 2(7):473-478.

[20] Khan Koffka, Ashok Sahai. [2012] A Comparison of BA, GA, PSO, BP And LM for Training Feed Forward Neural Networks in E-Learning Context. International Journal of Intelligent Systems and Applications, 4(7):23-26.

[21] Karegowda Asha Gowda, Manjunath AS, Jayaram MA. [2011] Application of Genetic Algorithm Optimized Neural Network Connection Weights for Medical Diagnosis of Pima Indians Diabetes. International Journal on Soft Computing, 2(2):15-23.

[22] Shanthi D, Sahoo G. Saravanan N. [2009] Evolving Connection Weights of Artificial Neural Networks using Genetic Algorithm with Application to the Prediction of Stroke Disease. International Journal of Soft Computing, 4(2):95-102.

[23] Valle Marcos Eduardo. [2014] An Introduction to Complex-Valued Recurrent Correlation Neural Networks. In Neural Networks (IJCNN), International Joint Conference on IEEE, 3387-3394.

[24] Dong Fang, Junao Wang [2015] Personal Information Extraction of the Teaching Staff Based on CRFs. In Network and Information Systems for Computers (ICNISC), International Conference on IEEE, 615-661.

[25] Archana Shree S, Vigneshwari S.[2016] Enhancing access of archives and ranking in web search, ARPN Journal of Engineering and Applied Sciences-ISSN 1819-6608, 11(9), MAY 2016/5926-5932