International Journal of Grid and Distributed Computing
Vol. 10, No. 11 (2017), pp.99-108
http://dx.doi.org/10.14257/ijgdc.2017.10.11.09

An Approach for Summarizing Hindi Text using Restricted Boltzmann Machine in Deep Learning

J. Anitha[1], N. Thirupathi Rao[2], Debnath Bhattacharyya[2] and Tai-hoon Kim[3*]

[1]Department of Information Technology, Vignan's Institute of Information Technology (A), Visakhapatnam, AP-530 049, India
[2]Department of Computer Science and Engineering, Vignan's Institute of Information Technology (A), Visakhapatnam, AP-530 049, India
[3]Department of Convergence Security, Sungshin Women's University, 249-1, Dongseon-dong 3-ga, Seoul 136-742, South Korea
[1]anithanv28@gmail.com, [2]nakkathiru@gmail.com, debnathb@gmail.com, [3]taihoonn@daum.net*

Abstract

Text summarization plays a crucial role nowadays due to large data available in day to day life. Reduced documents are useful and essential in the busy schedule of our lives. In this paper documents are summarized by four phases. They are preprocessing, feature vector generation, sentence score generation and summary generation. Sentence score is generated by Restricted Boltzmann machine (RBM) to improvise the result accuracy without losing the important information. Each of the sentence in the document undergoes all the phases and final summary generated is better as comparison among the existing (NN +fuzzy) and proposed method (Fuzzy + DL) with α value as 0.25 and β as 0.75. According to the analysis for precision rate at CR 30%, the Fuzzy+DL method is higher compared to the NN+fuzzy method (Fuzzy+DL)-0.875 and (NN+fuzzy)-0.256).The results shows the comparison graph for recall rate at 30%, the Fuzzy+DL method is higher compared to the NN+fuzzy method ((Fuzzy+DL)-0.777 and (NN+fuzzy)-0.6667) and the result also depicts the comparison graph for F-measure at CR 30%, the Fuzzy+DL method is higher compared to the NN+fuzzy method ((Fuzzy+DL)-0.8235 and (NN+fuzzy)-0.36363.

Keywords: *Deep learning, Restricted Boltzmann Machine (RBM), fuzzy, neural networks.*

1. Introduction

With the sensational development of the Internet, individuals are overpowered by the enormous measure of online data and reports. This extending accessibility of reports has requested comprehensive research in the region of programmed content outline. As indicated by Radef et al. [7] a synopsis is characterized as "a content that is delivered from at least one messages, that passes on essential data in the first text(s), and that is no longer than half of the first text(s) and for the most part, altogether not as much as that". Programmed content rundown is the assignment of creating a brief and familiar outline while saving key data substance and general significance. As of late, various methodologies have been created for programmed content synopsis and connected broadly in different areas. For instance, web crawlers produce clip Pets as the sneak peaks of the records [3].

Received (July 20, 2017), Review Result (October 30, 2017), Accepted (November 6, 2017)
* Corresponding Author

ISSN: 2005-4262 IJGDC

Different cases incorporate news sites which deliver dense portrayals of news subjects more often than not as features to encourage perusing or information extractive methodologies [8, 6, 2]. Programmed content synopsis is extremely testing, since when we as people outline a bit of content, we normally read it completely to build up our comprehension, and after that compose a rundown featuring its principle focuses. Since PCs need human learning and dialect ability, it makes programmed content outline an extremely troublesome and non-unimportant errand. Programmed content outline picked up fascination as ahead of schedule as the 1950s. An imperative research of nowadays was [8] for condensing logical reports. Luhn *et al.*, [8] acquainted a technique with extricate remarkable sentences from the content utilizing highlights, for example, word and expression recurrence. They proposed to weight the sentences of an archive as a component of high recurrence words, overlooking high recurrence normal words.

2. Methodology

In Restricted Boltzmann Machine (RBM), there are no direct connections between the hidden layers.

- Get an unbiased sample of $<x_i y_j>$data.
- Give a randomly selected training value x.
- Set a probability 1 to the binary state y_j of each hidden unit j.

It is given by,

$$p(y_j = 1 | x) = \sigma(t_j + \sum_j x_j w_{ij}) \quad (1)$$

$\sigma(x)$ → Logistic sigmoid function 1/ (1+exp (-x))

x_i y_j →Unbiased sample

Due to no direct connections between visible units in an RBM, it is also very easy to get an unbiased sample of the state of a visible unit,

$$p(x_i = 1 | y) = \sigma(s_i + \sum_j y_j z_{ij}) \quad (2)$$

Getting an unbiased sample of $<x_i y_j>_{model}$ however, it is more difficult. It can be done by starting at any random state of the visible units and performing alternating Gibbs sampling for a very long time.

- In an iteration of alternating Gibbs sampling, update the update the hidden units in parallel using equation 1 followed by update all the visible units in parallel using equation 2.
- Assign the training vector by the states of the visible units.
- Compute the binary states of the hidden units are all computed in parallel using equation 1.
- For the hidden layer, choose the binary states.

The change in weight is given by,

$$\Delta z_{ij} = \in (< x_i y_j >_{data} - < x_i y_j >_{recon}) \quad (3)$$

The fitness can be given by,

$$\text{Fitness} = \sum_{i=1}^{n} x - y \quad (4)$$

x→Output value from deep learning evaluation

y→original score value of each sentence

n→number of sentences

2.1. Input Hindi Text Document

The input which is to be summarized is a Hindi text document. It comprised of totally 10 to 15 sentences and around 280 words. Then these sentences are processed with the pre-processing technique. According to this process it divides the text into sentences and removes the stop words.

2.2. Feature Values

The below table shows the feature values for all sentences at the compression ratio 30, alpha as 0.25 and beta as 0.75. After pre-processing, the text subjected to the feature vector generation process to extract the feature values. It contains totally ten feature vectors.

CR=30, **α**=0. 25, **β**=0.75

Table 1. Feature Score

Doc. No.	*Para. No.*	*Line No*	*Sen.wt*	*Num. Data*	*Cue-phrase*	*Sen. Len*	*Pos. Value*	*Term Wt.*	*Title Val*
1	1	10	4.666667	0	0	0.841471	0.82283	25	0
1	1	2	3.8	0	0	0.756802	0.80548	43	0
1	1	3	5	0	0	0	0.52058	13	0
1	1	8	4	0.076923	0	0.141120	0.98498	25	0
1	1	16	3	0	0	0	0.97938	15	0
1	1	18	1	0	0	0	1	15	0
1	1	12	3.4	0	0.333333	0	0.18735	45	0
1	1	9	4	0	0	0	0.97688	10	0
1	1	5	3.4	0	0.333333	0	0.22835	27	0
1	1	1	0	0	0	0	1	2	0
1	1	4	4.75	0	0	0	0.15791	30	0
1	1	6	6.25	0	0	0.841471	0.58050	41	0
1	1	15	3.4	0	0	0	0.82961	31	0
1	1	7	4	0	0	0.756802	0.84593	19	0
1	1	11	2	0.181818	0	0.841471	0.54586	8	0
1	1	14	2	0	0	0	0.55591	13	0
1	1	17	0	0	0.333333	0	0.98283	4	0
1	1	13	5.8	0	0	0	0.19915	77	0
				0	0	0			
						0			

2.3. Sentence Score

The table represents the different score values generated from the feature values provided to generate the summary. The sentence scores are then analyzed to produce the synopsis from the text document. The key constraint utilized for the synopsis production is the mixture score and the weights of the sentences.

Table 2. Sentence Score

Fuzzy score	*DL Score*	*Hybrid Score*
0.388283	0.964164	0.820194
0.388283	0.935065	0.800253
0.388283	0.838765	0.732798
0.388283	0.821356	0.730833
0.388283	0.692933	0.631018
0.388283	0.638793	0.615585
0.388283	0.558013	0.532228
0.388283	0.549358	0.529149
0.388283	0.492799	0.47786
0.388283	0.409559	0.357169
0.388283	0.235421	0.29632
0.388283	0.245213	0.287032
0.388283	0.243567	0.279005

0.388283	0.228338	0.275134
0.388283	0.169186	0.208246
0.388283	0.155656	0.200665
0.388283	0.146946	0.19162
0.388283	0.012057	0.142869

2.4. Summary

CR:30 alpha:0.25 beeta :0.75
Summary: 30.0 %

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ (□□□□□□)
□□
□□
□□
□□
□□□
□□□,
□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ 1500
□□
□□□
□□
□□□□□□□□
□□□
□□
□□□□□□
□□.□□□□□□□□□□□□□□□□□
□□
□□□□□□□□, □□□□□□□□□□□□□□□□□□□□,
□□

The text summarization is based on the Hindi language. The performance analysis based on three parameters which are accuracy, recall and f-measure. The generated summary associated with parameters like compression rate, alpha and beta. The table shows the precision, recall and f-measures values at different compression ratios.

Table 3. Precision, Recall and F-measure Values

	CR	***α=0.25,β=0.75***	***α=0.5,β=0.5***	***α=0.75,β=0.25***
Precision	***30***	*0.875*	*0.875*	*0.875*
	40	*0.7142*	*0.8571*	*0.7777*
	50	*0.6666*	*0.744*	*0.8333*
	60	*0.5333*	*0.75*	*0.75*
recall	***30***	*0.7777*	*0.7777*	*0.875*
	40	*0.5556*	*0.6666*	*0.777*
	50	*0.6666*	*0.63*	*0.8333*
	60	*0.53333*	*0.3333*	*0.75*
F-measure	***30***	*0.825*	*0.8235*	*0.8235*
	40	*0.625*	*0.75*	*0.743*
	50	*0.533*	*0.533*	*0.6666*
	60	*0.30769*	*0.4615*	*0.5*

(a) Analysis on precision

The figure shows the precision rate at various alpha and beeta values (α=0.5, 0.25, 0.75 and β=0.5, 0.25, 0.75) as well as different compression ratios such as 30, 40, 50 and 60. The graph depicts that the precision rate is high at the compression ratio 30% and low at 60%.

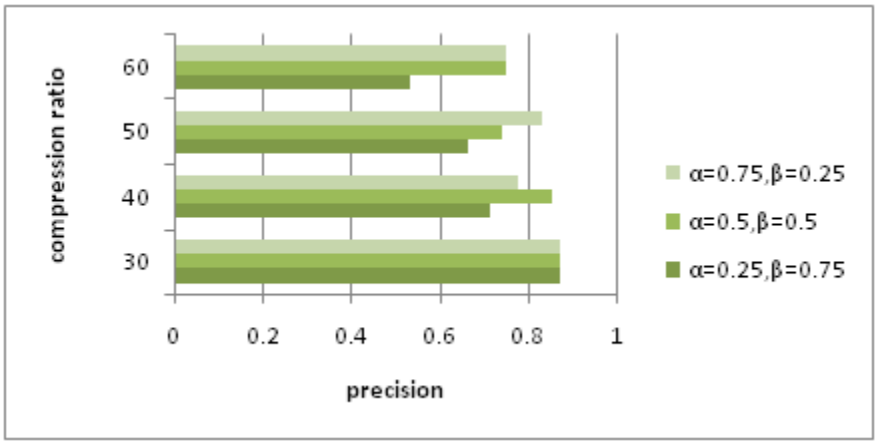


Figure 1. Analysis on Precision

(b) Analysis on recall

The figure depicts the recall rate at various alpha and beeta values (α=0.5, 0.25, 0.75 and β=0.5, 0.25, 0.75) as well as at different compression ratios such as 30, 40, 50 and 60. The graph depicts that the precision rate is high at the compression ratio 40%. The graph depicts that the precision rate is high at the compression ratio 30% and low at 60%.

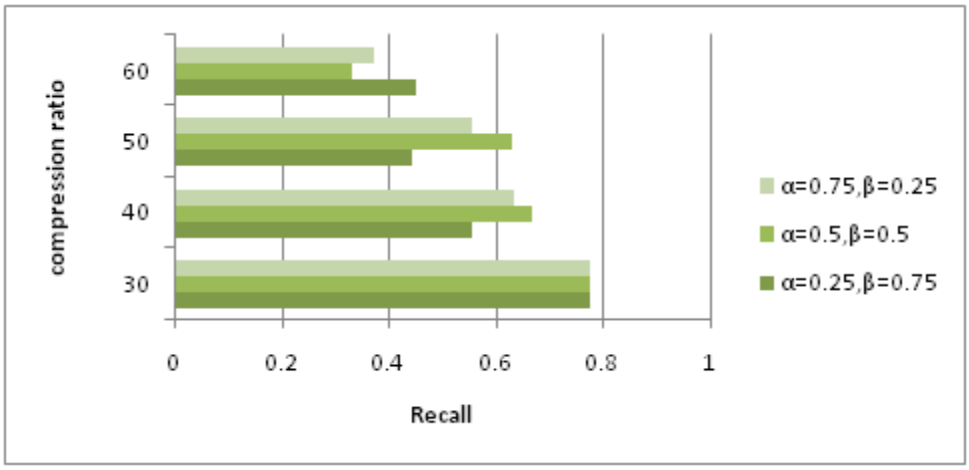


Figure 2. Analysis on Recall

(c) Analysis on F-measure

The figure represents the f-measure rate at various alpha and beeta values (α=0.5, 0.25, 0.75 and β=0.5, 0.25, 0.75)as well as at different compression ratios such as 30, 40, 50 and 60.

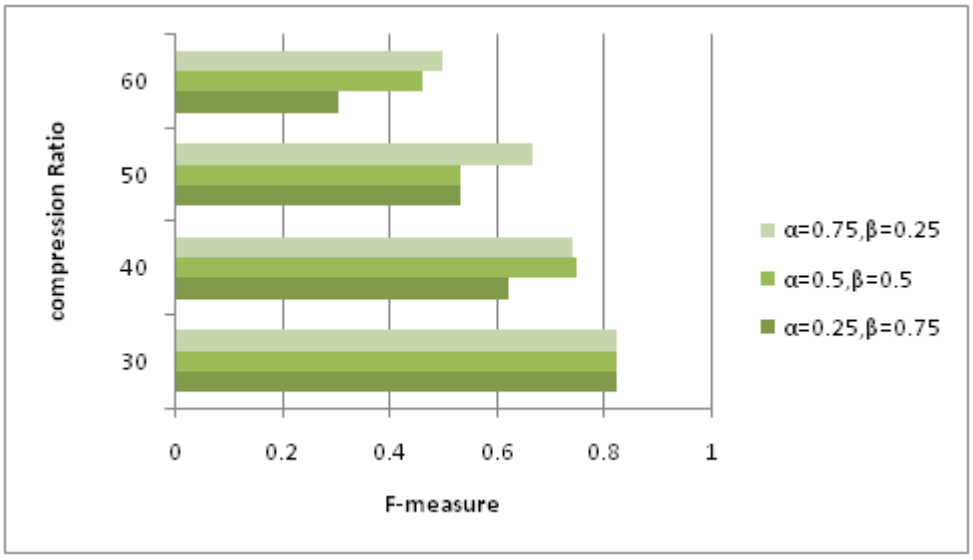
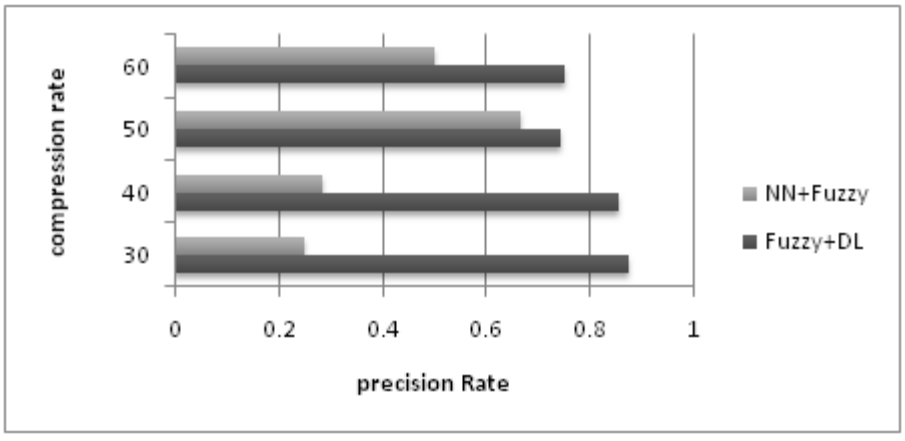


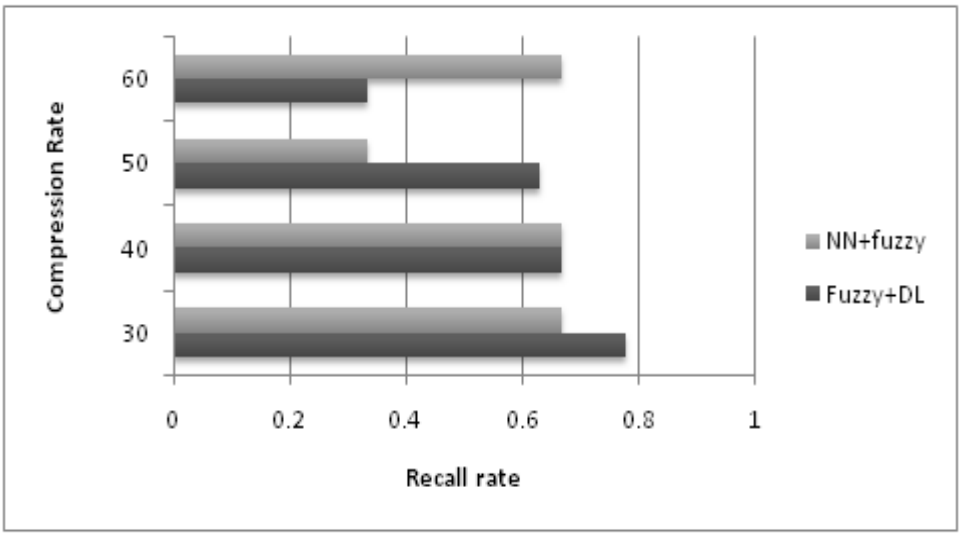
Figure 3. Analysis on F-measure

3. Comparison Analysis

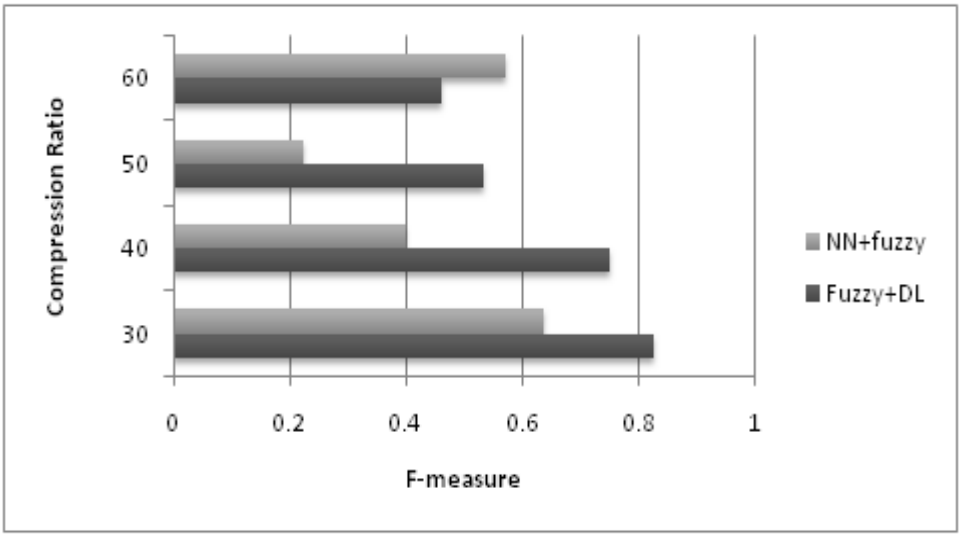
The graph in figure shows the comparison among the existing (NN +fuzzy) and proposed method (Fuzzy + DL) with α value as 0.5 and β as 0.5. According to the analysis, the figure shows the comparison graph for precision rate. At CR 30%, the Fuzzy+DL method is higher compared to the NN+fuzzy method ((Fuzzy+DL)-0.875 and (NN+fuzzy)-0.256). The figure shows the comparison graph for recall rate. At 30%, the Fuzzy+DL method is higher compared to the NN+fuzzy method ((Fuzzy+DL)-0.777 and (NN+fuzzy)-0.6667) and the figure depicts the comparison graph for F-measure. At CR 30%, the Fuzzy+DL method is higher compared to the NN+fuzzy method ((Fuzzy+DL)-0.8235 and (NN+fuzzy)-0.36363).



(a) Compression Rate Vs Precision Rate



(b) Compression Rate Vs Recall Rate



(c) Compression Rate Vs F-Measure

4. Conclusion

Boltzmann Machines (BMs) are a form of log-linear Markov Random Field (MRF), *i.e.*, for which the energy function is linear in its free parameters. To make them powerful enough to represent complicated distributions, some variables are hidden known as hidden units. To enhance the output, RBM restricts to only visible-visible and hidden-hidden connections. Finally, the output produced by undergoing four phases is yielding a better summarized output. It uses two main algorithms such as deep learning algorithm and the fuzzy classifier. The E-GSO is incorporate to optimize the weights and the combination of deep learning and the fuzzy generates the score for every sentence which is also called as the hybrid score. As per the experimental analysis, the Fuzzy+DL achieved and average precision rate of 0.7607, average recall rate of 0.54604 and average f-measure rate of 0.63221 at alpha is 0.25 and beeta as 0.75 and compression ratio of 30%, 40%, 50% and 60%. The comparative analysis likewise generated sensible summary to demonstrate that the proficiency of the Fuzzy+DL technique.

References

[1] Li Cun-he and Zhang Pei-ying, "Automatic text summarization based on sentences clustering and IEEE International Conference on extraction", Computer Science and Information Technology, PP 167 - 170, 2009.
[2] Giuseppe Di Fabbrizio, Amanda J. Stent and Robert Gaizauskas, "A Hybrid Approach to Multi-document Summarization of Opinions in Reviews", Proceedings of the 8th International Natural Language Generation Conference, pages 54–63, 2014.
[3] Chetana Thaokar and Dr. Latesh Malik, 'Test Model for Summarizing Hindi Text using extraction Method", Proceedings of 2013 IEEE Conference on Information and Communication Technologies, pp: 1138 – 1143, 2013.
[4] Vishal Gupta, G. Sl Lehal, "A Survey of Text Mining Techniques and Applications", Journal of Emerging Technologies in Web Intelligence, Vol. 1, No. 1, 60-76, AUGUST 2009.
[5] Aristoteles, Yeni Herdiyeni, Ahmad Ridha and Julio Adisantoso, "Text Feature Weighting for Summarization of Documents in Bahasa Indonesia Using Genetic Algorithm", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No 1, May 2012.
[6] Artificial Intelligence Review 16, Kluwer Academic Publishers (2001) 177-199], Freitas, A.A., Data Mining and Knowledge Discovery with Evolutionary Algorithms (forthcoming book). Springer-Verlag, 2002.
[7] Arman Kiani .B and Arman Kiani .T, "Automatic Text Summarization Using: Hybrid Fuzzy GA-GP", IEEE International Conference on Fuzzy Systems, PP 16-21, July 2006.
[8] H. Zha, "Generic summarization key phrase extraction using mutual reinforcement principal and sentence clustering", Proc.25TH Ann.1 INT'ACM SIGIR Conf. research and development in information retrieval, pp.113-120, 2002.
[9] "An Approach for Summarizing Hindi Text Through a Hybrid Fuzzy Neural Network Algorithm", J. Anitha, Prasad Reddy P.V.G.D. and Prasad Babu M. S., Journal of Information & Knowledge Management, Vol. 13, No. 4 (2014) 1450036(1-18), World Scientific Publishing Co.
[10] "Abstractive text summarization using sequence to sequence RNNs and beyond", Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL), pages 280–290.

