International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 259-266
Published Online August 2020 in IJEAST (http://www.ijeast.com)

AUTOMATIC HINDI TEXT SUMMARIZATION USING SELECTION AND ELIMINATION APPROACH

Mili Supreet
Department of CSE
Thapar Institute of Engineering and Technology, Patiala, Punjab, India.

Kanu Goel
Department of CSE
Thapar Institute of Engineering and Technology, Patiala, Punjab, India.

Madhuri Gupta
Department of CSE
Thapar Institute of Engineering and Technology, Patiala, Punjab, India.

***Abstract*— In recent years, the shoot up of web information has required to make intensive research in the field of automatic text summarization which is the part of the Natural Language Processing (NLP). This paper concentrates on different algorithms for text summarization of Hindi documents and to check which methods among them, are given the best summary. This paper has used selection and elimination based approach in order to make a summarizer for Hindi text documents. To make final summary several features are taken into account, including sentence resemblance to the first 100 and last 100 words and find sentence frequency on the basis of the word frequency, and sentence similarity by using Dice Coefficient and cluster the summary obtained by both methods and the Jaccard's Coefficient of dissimilarity is used to remove the sentences which are highly dissimilar in the summary. The results of the proposed algorithm are compared over different algorithms used for text summarization using Hindi corpus posted by IIT Bombay to identify which approach contains the most important points of the original text.**

Keywords— **Natural Language Processing (NLP), Text Summarizer, Tokenizer, Similarity Index**

2000 Mathematics Subject Classification— **68T50.**

I. INTRODUCTION

As the rapid growth of the Internet, the web is full of redundant data therefore methods and tools are being required to help users to manage large amount of data. With the help of Automatic Text Summarization the same information contained in one or more documents is being condensed and presented in a more concise and simple way as all users want to get information in more concrete manner (Gambhir and Gupta, 2017) (Gunawan, Pasaribu, Rahmat and Budiarto, 2017). Therefore, automatic text summarization can be very useful for the above mentioned purpose. The major points of the original text is contained by the product of this procedure.

The humans accomplish the same task by using the following steps—

1. The content of the document is perceived.
2. Important parts of information contained by the data is being identified.
3. The information is written in a very systematic way.

Given the variety of readily obtainable information, it would be useful to make an automatic text summarizer which is domain-independent. The three important aspects to determine the characteristics of research on automatic summarization are as follows—

1. Summary may be obtained either from a single document or multiple set of documents.
2. The important information contained by the document(s) must be included in the summary.
3. Summary should be concise and concrete.

The Automatic Text Summarizer is made by taking into consideration the above points. Several researches have been done till date on Automatic Text Summarization for different languages (Basiron, Jaya Kumar, Ong, Ngo and C Suppiah, 2016). But this paper carries the research done on Automatic Text Summarization based on Hindi language. The comparison is made with different algorithms and the new proposed algorithm. The selection and elimination-based approach is used in order to make a summarizer for Hindi text documents. Final summary which is an extractive summary has been made using sentence resemblance to the first and last 100 words present in the text and then, find the sentence frequency on the basis of the word frequency, and find sentence similarity by using Dice Coefficient (Binanto, Warnars, Abbas, Heryadi, Sianipar, Sanchez et al., 2018) and the Jaccard's Coefficient of dissimilarity (Niwattanakul, Singthongchai, Naenudorn and Wanapu, 2013) is used to remove those sentences which are highly dissimilar in the summary. Hindi corpus posted by IIT Bombay is used for this purpose and also compare the above algorithm with different algorithms present for text summarization to identify which approach will able to give better results.

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 259-266
Published Online August 2020 in IJEAST (http://www.ijeast.com)

The rest of the paper is organized as follows. Proposed embedding and extraction algorithms are explained in section II. Experimental results are presented in section III. Concluding remarks are given in section IV.

II. RELATED WORK

This section contains the work done in the respective field earlier. All the proposed methods which are help to make an effort to introduce a new method in this field to carry out the better results. Text relationship maps had been used to find paragraphs linked along bushy path, segmented path, segmented bushy path and augmented segmented bushy path. Paragraphs linked along each path are extracted to constitute to automatic summary and compare with the manual summary (Mitra et al., 1997).

Table -1 Different approaches for automatic text summarization

Language Used	Description	References
English	Automatic text summarization using different methods of paragraph extraction in comparison with manual summary	Mitra *et al.* (1997) (Mitra, Singhal and Buckley, 1997)
English	Mitra *et al.* (1997) (Mitra, Singhal and Buckley, 1997)	Das and Martins (2007) (Das and Martins, 2007)
English	To make automatic text summarization the use of genetic algorithm (GA), mathematical regression (MR) are studied	Fattah and Ren (2008) (Fattah and Ren, 2008)
English	The summarizer is made using two techniques, first is rule based summarization and second is keyword based summarization	Soumya *et al.* (2011) (Soumya, Kumar, Naseem and Mohan, 2011)
Vietnamese texts	An optimized text summarization method is used which is based on Naive Bayes and topic word for single syllable language	Ha Nguyen Thi Thu (2014) (Thu, 2014)
Hindi	Comparative study of hindi text summarization using genetic algorithm and neural network techniques	Kadam *et al.* (2015) (Kadam, Patil and Gulathi, 2015)
Hindi	To make hindi summarizer an extractive approach is used which selects the significant sentences which are based on a thematic approach.	Kumar and Yadav (2015) (Kumar and Yadav, 2015)
Hindi	Hindi text summarizer is made using rule based approach	Gupta and Garg (2016) (Gupta and Garg, 2016)
Unified Medical Language	Different approaches are used for identifying the important conceptsin probabilistic biomedical text summarization	Moradi and Ghadiri (2018) (Moradi and Ghadiri, 2018)

Machine Learning approach had been used by training manual summaries of training set using negative and positive keywords in the sentence, features sentence position, sentence similarity, etc. and genetic algorithm had been used to find the feature weight. Automatic text summary of the given text had been found by training the model on optimized features and weights (Fattah and Ren, 2008).

Rule based approach was providing the summary which satisfied all the hand-coded rules depending on the kind of text had been used (Gupta and Garg, 2016). Similarity measures like Jaccard's Coefficient, sentence similarity are used to find multiple document for text summarization by finding the summaries of individual documents and then finding the similarities between those summaries to generate final summary (Yasin, Yasin and Yasin, 2011)(Aliguliyev, 2009). Different approaches for automatic text summarization have been described in Table 1

III. ARCHITECTURE AND IMPLEMENTATION OF PROPOSED WORK

This paper proposes the algorithm for automatic text summarization for Hindi documents. The proposed algorithm follows the selection and elimination approach to make text summarizer for Hindi documents. The text has been taken from IIT Bombay repository named "361 utf.txt" whose summary has been made using the proposed algorithm. In Algorithm 1 summaries have been made. First summary has been made by the union of sentence frequency and Dice Coefficient of similarity as described in Algorithm 2. Second summary has been made using Jaccard's Coefficient of dissimilarity which contains those sentences whose dissimilarity coefficient is very low as described in Algorithm 3. Third, the final summary has been made by intersection of first and the second summary as described in Algorithm 4. For making the summary of Hindi document following methods are used:

- *Word Tokenization*

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 259-266
Published Online August 2020 in IJEAST (http://www.ijeast.com)

In this mechanism the sentences are break into the words and store those words into the list.

• *Removal of Stop-word*
The text file at the given link is used which contains all the stop words present in Hindi language. If the word of the text is present in the stop-word list then,that word is removed from the list of words.

• *Stemming*
In this mechanism the stemming is done to cluster the same morphological variant words so that data for summarization becomes less. The stemmer is used from the given link.

• *Word frequency*
The word frequency is calculated after the stop word removal and stemming. The word frequency is measured as how many times the same comes in the text. If the word lies in first 100 and last 100 topic words then, their frequency is increased by the maximum frequency of the word.

• *Sentence Frequency*
The sentence frequency is evaluated by summing up the frequencies of all words present in the text. Those sentences are chosen whose sentence frequency is greater than 1.7 * average of sentence frequency of sentences present in the text. Average is calculated by using formula:

$$Average = \frac{Sum(Sentence\ Frequency)}{Total\ no.of\ Sentences} \quad (3.1)$$

In thresh-hold, value 1.7(say as α) is taken randomly because the appropriate number of characters contained by actual summary were obtained by this value as shown in Figure 3 and explained in Results and Discussion section.

• *Dice Coefficient of similarity*

In this mechanism if the sentence is not greater than the 1.7 * average of total number of characters than the similarity of that sentence is compared with all other sentences present in the text with the help of Eq. (3.2). If the similarity coefficient is greater than 0.31 than that sentence should be added into the summary. Hence, in this way first summary has been made.

$$Dice\ Coefficient(X,Y) = 2 * \frac{|X \cap Y|}{|X| + |Y|} \quad (3.2)$$

• *Jaccard's Coefficient of dissimilarity*

The second summary is made with the help of Jaccard's Coefficient of dissimilarity of each sentence with rest of sentences present in the document. Those sentences are included into the summary whose Jaccard's Coefficient is less than of dissimilarity 0.805.

$$Jaccard\ Coefficient\ (X,Y) = 1 - \frac{|X \cap Y|}{|X| + |Y|} \quad (3.3)$$

Algorithm 1 Proposed Algorithm

```
for each word in stw1,stw2,.....,ste do
    if word in (sw1,sw2,.....swt) then
        continue
        if word in (wf1,wf2,.......,wfn) then
            wf[word]= 1+wf[word]
        else
            wf[word]=1
        end if
    end if
end for
for each word tw1, tw2, .....,twf do
    if word in wf1, wf2, .......,wfn then
        wf[word] = max+wf[word]
    else
        wf[word] = max
    end if
end for
for each sentence in (s1,s2,.....,sm) do
    λ[sentence] = 0
    for word in w do
        if word in (sw1, sw2, .....swt) then
            continue
            if sent in λ and μ in (wf1, wf2, .......,wfn) then
                λ[sent] = wf [μ] + λ[sent]
            else
                if μ in (wf1,wf2,.......,wfn) then
                    λ[sent] = wf [μ]
                else
                    λ[sent] = 0 + λ[sent]
                end if
            end if
        end if
    end for
end for
```

Algorithm 2 Summary using sentence frequency and Dice Coefficient of similarity

```
for i in range(0,t) do
    if p[i] in λ and λ[p[i]]≥(1.7 * avg) then
        summary1 += " " + p[i]
        continue
    end if
end for
for j in range((i+1),t) do
    f = 2∗((|X∩Y|) / ((|X|+|Y|)))
    if f≥0.31 then
```

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 259-266
Published Online August 2020 in IJEAST (http://www.ijeast.com)

```
10:          if p[i] not in sum1 then
11:                sum1 = sum1 + " " + p[i]
12:          end if
13:     end if
14: end for
```

Algorithm 3 Summary using Jaccard's Coefficient of dissimilarity

```
1:  for j in range((i+1),t) do
2:       f = 2∗((|X∩Y|)/((|X∪Y|)))
3:       if f≤0.805 then
4:            if p[i] not in sum2 then
5:                 sum2 = sum2 + " " + p[i]
6:            end if
7:       end if
8:  end for
```

Algorithm 4 Final Summary using intersection of sum1 and sum2

```
1:  for sent in sum1 do
2:       if sent in sum2 then
3:            sum3 += " " +sent
4:       end if
5:  end for
```

Finally, the third summary is made using the intersection of first and the second summary which has covered up all the important lines present in the original text.

A. ***Architecture Design –***

This design includes the basic workflow of the proposed system. Figure 1 explain the basic steps used to carry out the result. Figure 2 explains the block diagram of the proposed algorithm.

B. ***Implementation Design –***

This section contains the proposed algorithm used for finding the summary of the Hindi text. Table 2 shows the various symbols with their description which are used in the algorithm.

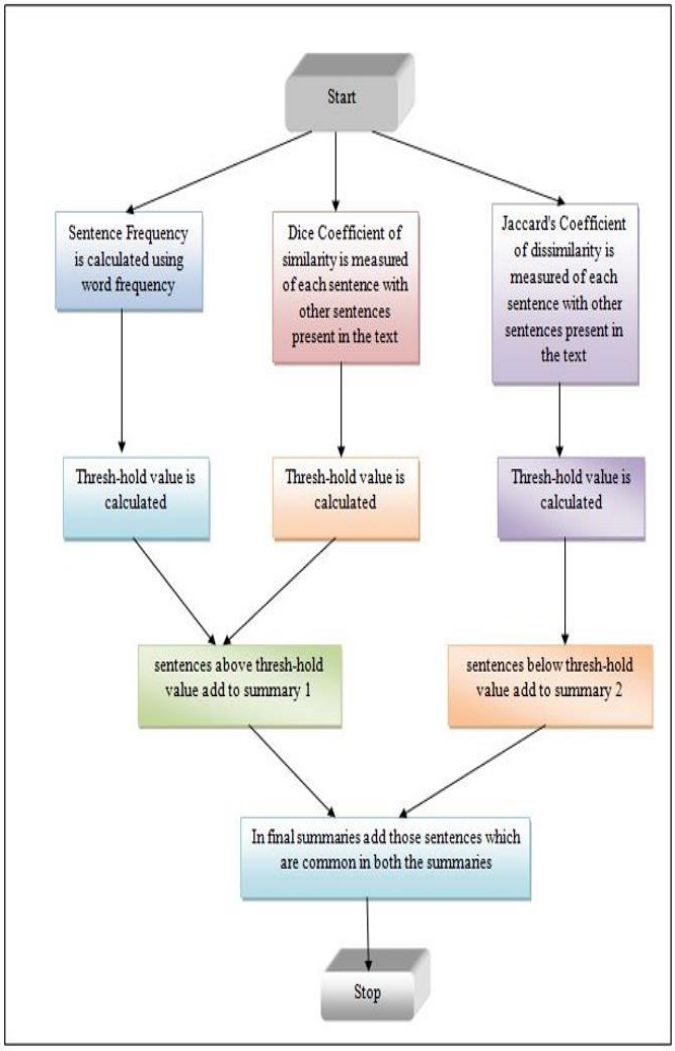

Fig. 1. Basic workflow of Algorithm

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 259-266
Published Online August 2020 in IJEAST (http://www.ijeast.com)

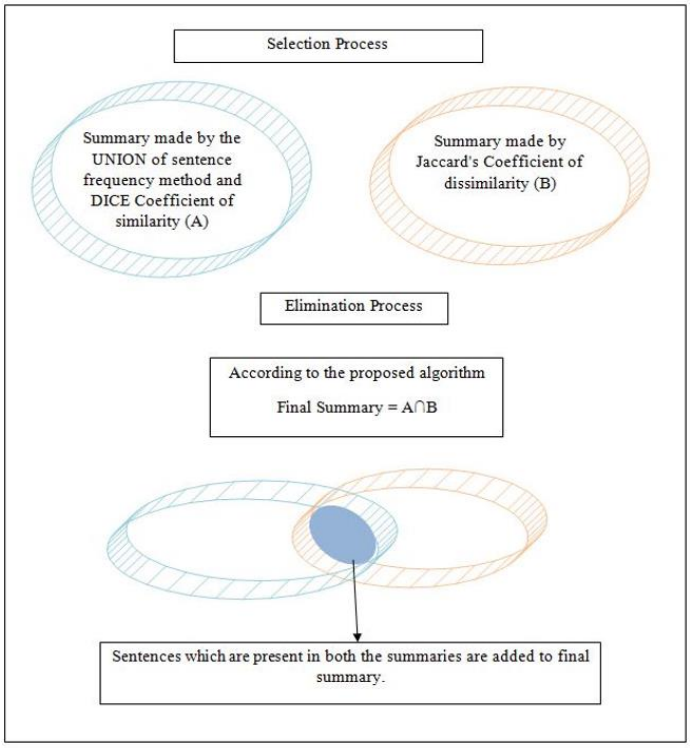

Fig. 2. Block Diagram of proposed work

Table -2 Notations used in the Proposed Algorithm

Symbol	Description
n	number of words in the text
m	number of sentences in the text
t	number of stop-words present in stop-word file.
e	total number of stem words
f	total number of topic words after stop-word removal
i	variable
J	variable
p[i]	sentence at index i where 0<=i<=m
X	unique words in sentence i
Y	unique words in sentence j
λ	value of sentence
μ	value of word
(w1, w2, ...wn)	words tokenizer
(s1, s2, ..,sm)	sentence tokenizer
(sw1, sw2, ...swt)	stop-words
(stw1, stw2, ..., ste)	stem words
(tw1, tw2, .., twf)	topic words
(wf1, wf2, ..., wfn)	words frequency
sum1	summary made by the union of sentence frequency & Dice Coefficient of similarity
sum2	summary made by Jaccard's Coefficient of dissimilarity
Sum3	final summary made by the intersection of sum1 and sum2

IV. EXPERIMENTAL SET-UP

The experiment has been conducted on the system having i5 processor and 8 GB RAM. The code to carry out the results has been written in Python language. Software used are Python 3.6. NLTK (Natural Language Processing Toolkit) is used to perform basic functions like word tokenization, sentence tokenization, import self-made corpus, etc (Lobur, Romanyuk and Romanyshyn, 2011). Also, codecs package is used to import files in the code. The text whose summary is made has been taken from IIT Bombay repository named "361 utf.txt". Stop-word file has been taken from the given link. Hindi Stemmer file has been taken from the respected given link. Summary of different Hindi corpuses used in given in Figure 3.

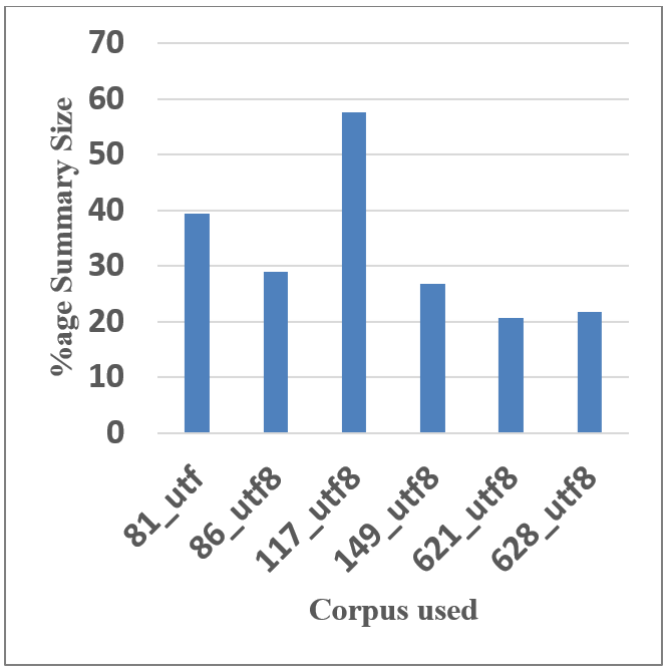

Fig. 3. Block Diagram of proposed work

V. RESULTS AND DISCUSSIONS

The proposed text summarizer is implemented on the Hindi language text document belonging to the IIT Bombay repository. The results proposed by the system are promising as the final summary contains all the important lines present in the original text and also gave the summary between 30%-40% of the original text which is the necessary conditions to make the good summary for any given text. To get summary

International Journal of Engineering Applied Sciences and Technology, 2020
Vol. 5, Issue 4, ISSN No. 2455-2143, Pages 259-266
Published Online August 2020 in IJEAST (http://www.ijeast.com)

between 30%-40% of the original text some values have been calculated during the implementation of the algorithm proposed in this paper. The proposed algorithm works on 3 summaries in which first summary has been made using the union of sentence frequency and Dice Coefficient of similarity, second summary has been made using Jaccard's Coefficient of dissimilarity, and the third but the final summary has been made by the intersection of the first and the second summary.

In the first summary the sentence chosen for sentence frequency are those whose value is greater than α * average of sentence frequency of sentences present in the text. The value α is chosen 1.7 by iteratively finding the summary of a text at various values of α ranging between 1-2 at the interval of 0.1. At α equals to 1.7, the summary is approximately 33% of the original text is obtained as shown in Figure 4.

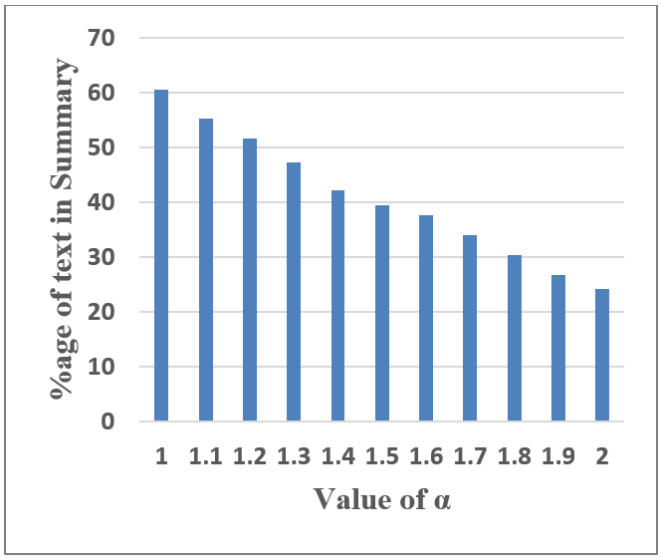

Fig. 4. Block Diagram of proposed work

The sentences whose Dice Coefficient of similarity is equal to or greater than 0.31 are chosen in the summary because they are more similar to the rest of sentences present in the original text. Therefore, when the thresh-hold value of Dice coefficient of similarity is equal to or greater than 0.31 then, all the important points sum up in the summary text which is approximately 33% of the original text as shown in Figure 5. The thresh-hold value is obtained by varying its limits ranging between 0.1-0.5.

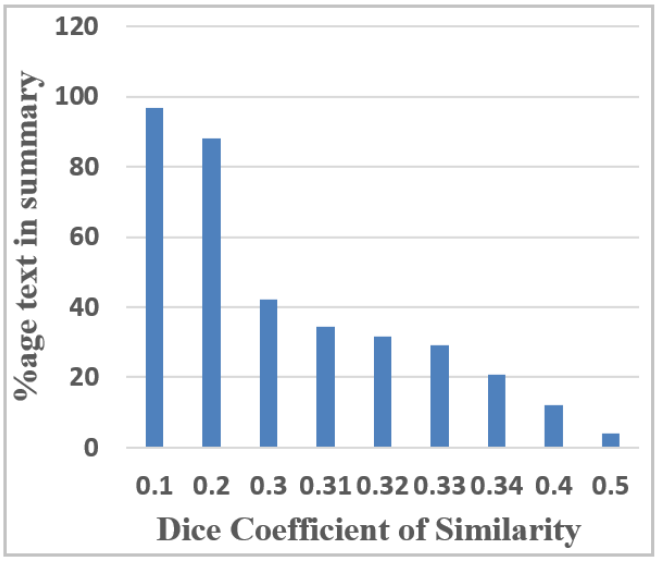

Fig. 5. Block Diagram of proposed work

The sentences whose Jaccard's Coefficient of dissimilarity is equal to or less than 0.805 are chosen in the summary because they are more similar to the rest of sentences present in the original text. Therefore, when the thresh-hold value of Jaccard's Coefficient of dissimilarity is equal to or less than 0.805 then, all the important points sum up in the summary text which is approximately 33% of the original text as shown in Figure 6. The thresh-hold value is obtained by varying its limits ranging between 0.1-0.5.

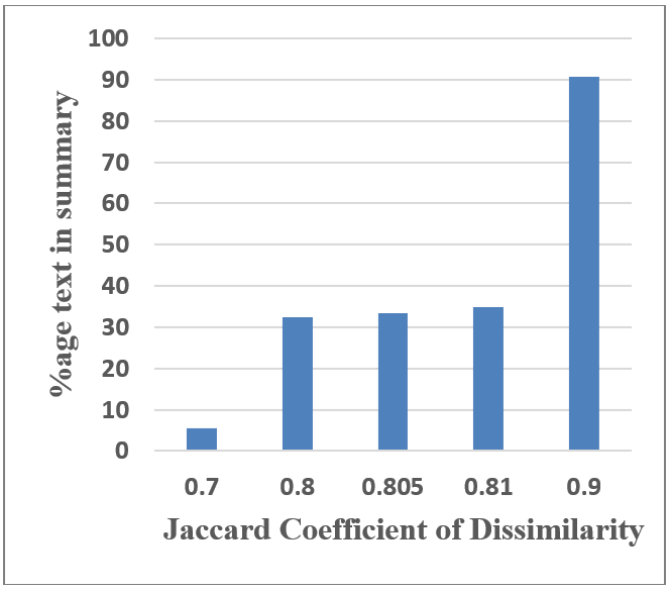

Fig. 6. Block Diagram of proposed work

First the union of summaries obtained by sentence frequency and Dice coefficient of similarity has been done. This is because some of the important sentences which are not taken by the sentence frequency method, through union with Dice method those sentences are able to include into the summary. Afterwards, the obtained summary is matched with the summary obtained by Jaccard's coefficient because to eliminate all those sentences which are less important in the final summary. Therefore, the final summary contains all the important sentences present in the text.

This approach is useful because all the important sentences are included into the summary and the final summary is 33% of the original text as shown in Table 3 which fulfil the necessary condition that the summary must be 30%-40% of the original text.

Table -3 Outcomes of Experiment

Description	Length	Parameters
Length of text	12258	
Length of summary obtained by sentence frequency method	4163	$\alpha = 1.7$
Length of summary obtained by Dice Coefficient of similarity method	4206	Thresh-hold value of Dice method = 0.31
Length of summary obtained by the union of sentence frequency method and Dice Coefficient of similarity method	6027	$\alpha = 1.7$, Thresh-hold value of Dice method = 0.31
Length of summary obtained by Jaccard's Coefficient of dissimilarity method	4082	Thresh-hold value of Jaccard's Coefficient = 0.805
Length of final Summary	4173	$\alpha = 1.7$, Thresh-hold value of Dice method = 0.31,Thresh-hold value of Dice method=0.805

VI. CONCLUSION

The Automatic text summarizer is very useful to get simple, concrete, and exact information from the large number of redundant data present on the web. The approach given in this paper to make text summarizer for Hindi language is very much promising as it includes all the important lines of the original text in the very systematic way in the summary. As a good summary contains 30%-40% of the original text, this approach also able to fulfil this condition. Hence, this approach is one of most accurate approach to make text summarizer. For future work further study will be done to make cumulative weighted text summarizer using more set of Hindi stop-words and more appropriate Hindi stemmer.

VII. REFERENCE

[1] Aliguliyev, R. M. (2009). A new sentence similarity measure and sentence based extractive technique for automatic text summarization, Expert Systems with Applications 36(4): (Pg 7764– 7772).

[2] Basiron, H., Jaya Kumar, Y., Ong, S. G., Ngo, H. C. and C Suppiah, P. (2016). A review on automatic text summarization approaches, Journal of Computer Science 12:(Pg 178–190).

[3] Binanto, I., Warnars, H. L. H. S., Abbas, B. S., Heryadi, Y., Sianipar, N. F., Sanchez, H. E. P. et al. (2018). Comparison of similarity coefficients on morphological rodent tuber, 2018

[4] Indonesian Association for Pattern Recognition International Conference (INAPR), IEEE, (pp. 104–107).

[5] Das, D. and Martins, A. F. (2007). A survey on automatic text summarization. language technologies institute.

[6] Fattah, M. A. and Ren, F. (2008). Automatic text summarization, World Academy of Science, Engineering and Technology 37(2): (Pg 192).

[7] Gambhir, M. and Gupta, V.(2017). Recent automatic text summarization techniques: a survey, Artificial Intelligence Review 47(1): (Pg 1–66).

[8] Gunawan, D., Pasaribu, A., Rahmat, R. and Budiarto, R. (2017). Automatic text summarization for indonesian language using textteaser, IOP Conference Series: Materials Science and Engineering, Vol. 190, IOP Publishing, (Pg. 012048).

[9] Gupta, M. and Garg, N. K. (2016). Text summarization of hindi documents using rule based approach, (2016) international conference on micro-electronics and telecommunication engineering (ICMETE), IEEE, (pp. 366–370).

[10] Kadam, D. P., Patil, N. and Gulathi, A. (2015). A comparative study of hindi text summarization techniques: Genetic algorithm and neural networks,

International Journal of Innovation and Advancement in Computer Science 4.

[11] Kumar, K. V. and Yadav, D. (2015). An improvised extractive approach to hindi text summarization, Information Systems Design and Intelligent Applications, Springer, (pp. 291–3000).

[12] Lobur, M., Romanyuk, A. and Romanyshyn, M. (2011). Using nltk for educational and scientific purposes, (2011) 11th international conference the experience of designing and application of CAD systems in microelectronics (CADSM), IEEE, (pp. 426–428).

[13] Mitra, M., Singhal, A. and Buckley, C. (1997). Automatic text summarization by paragraph extraction, Intelligent Scalable Text Summarization.

[14] Moradi, M. and Ghadiri, N.(2018). Different approaches for identifying important concepts in probabilisticbiomedicaltextsummarization,Artificialintelligenceinmedicine84:(Pg 101–116).

[15] Niwattanakul, S., Singthongchai, J., Naenudorn, E. and Wanapu, S. (2013). Using of jaccard coefficient for keywords similarity, Proceedings of the international multiconference of engineers and computer scientists, Vol. 1, (pp. 380–384).

[16] Soumya, S., Kumar, G. S., Naseem, R. and Mohan, S. (2011). Automatic text summarization, International Conference on Computational Intelligence and Information Technology, Springer, (pp. 787–789).

[17] Thu, H. N. T. 2014. An optimization text summarization method based on naive bayes and topic word for single syllable language, Applied Mathematical Sciences 8(3): (Pg 99–115).

[18] Yasin, H., Yasin, M. M. and Yasin, F. M. (2011). Automated multiple related documents summarization via jaccard's coefficient, International Journal of Computer Applications 13,(Pg 3).