International Journal of Computational Intelligence Research
ISSN 0973-1873 Volume 13, Number 1 (2017), pp. 113-125

http://www.ripublication.com

Application of Minimum Vertex Cover for Keyword –based Text Summarization Process

Atowar-Ul Islam[1] and Bichitra Kalita[2](Retd.)

[1] Department of Computer Science & IT, Cotton College,
Guwahati, Assam -781001, India.

[2] Department of Computer Application (MCA), Assam Engineering College,
Guwahati - 781013, Assam, India.

Abstract

In this article, a new approach has been discussed to solve text summarization problem using minimum vertex cover for weighted graph on the basis of keyword based approach. It is known that the text summarization problem is an NP-Complete problem.

Keywords: Minimum Vertex Cover, Text summarization, sentence graph, keyword and degree

1. INTRODUCTION:

Text summarization is the process of reducing the content of document with a automated system that retains most important points of the original document. In other words it is the process of automatically creating a compressed version of a given text that provides useful information for the user. The process of text summarization can be break down into three phases and they are found to be analysis, transformation, and synthesis phase. In the analysis phase, the input text is analyzed and a couple of notable features have been chosen. The transformation phase transforms the results of analysis into a summary representation. Finally, the synthesis phase takes the summary representation, and produces an appropriate summary corresponding to user's needs. It has been found that compression rate, which is defined as the ratio between the length of the summary and that of the original, is an important factor influencing the quality of the summary. It is true that when compression rate decreases, the summary will be more concise; however, more information is lost.

While the compression rate increases, the summary will be larger; relatively, more insignificant information is contained. In fact, when the compression rate is 5–30%, the quality of the summary is acceptable [18,19]. In last few decades, there are lot of approach has been carried out by different researcher. These approaches can be classified into two board categories – extractive and abstractive. Extractive method produces summary by selecting a subset of existing words, phrases, or sentences from the original text. Whereas the abstractive method uses natural languages generation techniques to produce the summary. Abstractive method may reshuffle words, sentences or generate new words, sentences that resemble the human summary generation process. The other dimension of classifying the summarization process is the context factor. Jones et al. [12-17] classified summarization process into three different classes – input factor, purpose factor and output factor. Input factors focus on the source material including style and units, purpose factors concentrate on the use and target audience of the process, and output factors focus on reduction and format. An abstractive method uses heavy natural language generation techniques.

We are interested to select a few sentences which will cover the semantics of the given text. The meaning of a text explores from one sentence to next and ripened, established a concept at the end. We see the problem of summarization as a problem of selecting important sentences from a set of sentences. As each sentence are semantically connected, we view the collection of sentences as a connected weighted graph, considering sentence as a node. We use minimum vertex cover (MVC) to select the important sentence from the connected weighted graph. A vertex cover is a smallest set of possible number of vertex that covers all the edges of the graph. That is, a graph G =(V,E), $S \subseteq V$, is a vertex cover if $\forall \{u,v\} \in E : u \in S \; \forall v \in S$. An article is a collection of sentences. We consider each sentence of the article as a vertex of the graph. Thus vertex labels are the sentence numbers from the article. For our experiments, we have collected around 1,000 research article from different archive that have explicit assignment of keywords. Based on the keywords, we assign weight to each edge.

This paper is organized as follows: Section-1 includes overview of Text summarization technique. Section-2, describes the prior works on text summarization. Section-3 includes our proposed approaches. In Section-4 report of our experimental results and a discussion thereof has been discussed. Conclusion is included in section-5.

2. RELATED WORK:

Various researchers have been working from different perspective. Takamura et al. [1] discuss some decoding algorithms using greedy, randomized and branch and bound approach. On the basis of the result obtained from comparative experiments, they augmented the summarization model to account the relevance of the document cluster and reported that the augmented model is superior to the best performing method in the literature (DUC'04 on ROUGE-1) without stop words. It has been found that Antiquoueira et al. [2] used the concepts and metrics of complex networks to select sentences form a given text. They used graph, where nodes represented the sentences and two nodes are connected when they share common meaningful nouns. Employing different graph concepts, such as degree of nodes, length of shortest paths, d-rings and k-cores and a voting system on Brazilian-Portuguese texts, they found that the system performed better than summarizers that do not employ deep linguistic knowledge, comparable to state-of-the-art summarizers based on expensive linguistic resources. Gupta et al. [3] has discussed the text summarization by reducing the source text into a shorter description, preserving its information content and the whole meaning. They used linguistic technique to observe and deduce the text and then find new concepts. The new concept generating a new shorter text that conveys the most important information from the original text document. Stern et al. [4] has discussed an open source system for recognizing textual entailment. The RTE system can use for end application of researcher to benefit from generic textual inference and other one is RTE researcher can integrate their novel algorithm and knowledge resources into our system, saving the time and effort of developing a complete RTE system. ALGULIEV et al. [5] has discussed two types of summarization, namely extractive and abstractive. Extractive summarization methods simplify the problem of summarization into the problem of selecting a representative subset of the sentences in the original documents. Abstractive summarization may compose novel sentences, unseen in the original sources. They focused on sentence based extractive document summarization. The extractive summarization systems are typically based on techniques for sentence extraction and aim to cover the set of sentences that are most important for the overall understanding of a given document. They also proposed unsupervised document summarization method that created the summary by clustering and extracting sentences from the original document. For this purpose, new criterion functions for sentence clustering have been proposed.

It has been found that the similarity measures played an important role in document clustering. They also developed a discrete differential evolution algorithm to optimize the criterion functions. Gupta et al. [6] proposed that sentence connectivity is a textual characteristic that may be incorporated intelligently for the selection of sentence of a well meaning summery, which possesses the text summarization task as an optimization problem and attempt to solve it using Weighted Minimum Vertex Cover. They also used an established indicator of semantic relationships between text units used to measure sentence connectivity and construct the graph two operate WMVC. Sarkar [7] proposed that the medical literatures on the web are the important sources to help clinicians in patient-care. Initially, the clinicians go through the author-written abstracts or summaries available with the medical articles to decide whether articles are suitable for in-depth study. Since all medical articles do not come with author written abstracts or summaries, automatic summarization of medical articles will help clinicians or medical students to find the relevant information on the web rapidly. A summarization method has been discussed by them which combines several domain specific features with some other known features such as term frequency, title and position to improve the summarization performance in the medical domain. Their experiments showed that the incorporation of domain specific features improves the summarization performance. Khamayseseh et al. [8] proposed a new algorithm for minimizing the number of node that represents the wireless sensor with high transition power and thus minimized the power consumption of the wireless sensor network and increased the network life time. Balaji et al. [9] proposed the MVC problem as a classic graph optimization NP - complete problem. A competent algorithm, called vertex support algorithm (VSA), is designed to find the smallest vertex cover of a graph. The VSA is tested on a large number of random graphs and DIMACS benchmark graphs. Comparative study of this algorithm with the other existing methods has been carried out. Extensive simulation results showed that the VSA can yield better solutions than other existing algorithms found in the literature for solving the minimum vertex cover problem. **T**he Minimum Vertex Cover of a class of Regular Planar Sub-Graph H(2m+2,3m+3), K(2m+2,4m+4) for m≥2 and J(2m+2,5m+5) for m≥5 obtained from the complete graph K2m+2 had already been discussed by kalita et.al [11]. An algorithm has been developed to find the minimum vertex cover of these type of regular planar sub-graph. Finally the application of minimum vertex cover has been found to reduce the power consumption of sensor network.

3. NEW METHOD ADOPTED:

To perform Text generation we need to first construct a sentence graph. Consider each sentence is a node in the collection document. Use all sentences for building the sentence graph. First, find out the Keyword from the document or article. Read every sentence and find the keyword. When we find a keyword, count it. There after count same keyword in a sentence .Thus considers each sentence is a node. Every adjacent sentence is connected by a node. Each edge has weight according to their adjacent vertices keyword. When adjacent sentences are connected and found that if there is no keyword then weight should be considered as zero otherwise the weight is counted according the number of keywords and degree is counted according the edges incidents in the node which will be discussed later.

3.1 Preprocessing Phase: At first we read all the sentences from the documents or article and provide sentence number. When sentences are ended with dot (.) or question mark (?) or exclamatory mark (!) then provide a sentence number. Sentence number is started with ‘0’ and incremented in ascending order until and unless the article or documents is completed. After providing sentence number, we find out the Keyword from the article. We read every sentence and find out the keyword and count it. We do not count same keyword in a sentence and consider each sentence is a node. Every adjacent sentences connected by an edge. Each edge has weight according their sentence keyword and adjacent sentences keyword. We consider one paper and construct Table 1 in section-4 according to above details. After finding the number of keywords and same keywords presents in the article or papers, two graph have been constructed by the following two rules.

- Rule-1: Just connect the sentences (nodes) one by one up to last nodes or sentence and for construct a graph we connect last and first nodes otherwise it will be a tree.
- Rule-2: Connect all the adjacent sentences (nodes) and also connect the nodes which contain similar keywords.

3.2 Calculating the Weight: First calculate the weights according their keywords. When adjacent sentences(vertices) are connected for the graph to calculate the weight of the edge,the keywords plays an important rule. We put the weight of the edge is 0 when there is no keywords, otherwise the weight is counted according sentence keyword and adjacent sentences keywords. [weight = sentence keyword + adjacent sentence keyword]. The following example of figure-1 is considered to calculating the weight of sentence graph. The weight of the edge connecting the vertices a and b is 4 when each of them contains two keywords. If **c** is another vertex which contains 3 keywords and which is adjacent to **b** then the weight of edge between **b** and **c** is 5 and total weight of **b** is changed to 9.The same procedure can be applied after introduction of new vertex with keywords. Again if **d** is another nodes which contains 2 keywords and adjacent to **a** and **b** nodes then total weight of **b** and **a** is updated to 13 and 8 respectively and total weight of **d** is now 8. The sentences which contain same keywords for the vertices, these vertices are connected by different edges.

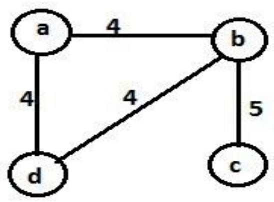

Figure -1(showing calculating weight)

3.3 Calculating the Degree: The degree of the sentence graph is calculated according the rules of graph theory. When first rule is applied as stated in 3.1 to construct a sentence graph the graph will be always two regular planar sentence graph. But when the rule two is applied , as stated in 3.1 to construct a sentence graph the graph may not be regular but it is always a simple graph(it will be discussed later with example).

The following algorithm has been developed to find out the minimum vertex cover of the sentence graph.

3.4 - Algorithm:

INPUT: Sentence Graph.

OUTPUT: To find out the minimum vertex cover of Sentence Graph.

Algorithm 1: Sentence selection using minimum vertex cover

1. while $e \in E = \varphi$ do
2. select v_i with *max*(*degree*(v_i)), $\forall$ i,j = 1,2,3 .... n
3. if degree(v_i) = degree(v_j) then
4. select vi with *max*(*degree*(v_i)) and *max*(*weight*(v_i))
5. else if (*degree*(v_i) and *weight*(v_i)) = (*degree*(v_j) and *weight*(v_j)) then
6. select either v_i or v_j
7. degree($v_{i,}v_j$) = degree($v_{i,}$ v_j) - no of connected adjacent E(v_i,v_j)
8. weight(v_i) = weight($v_{i,}v_j$) – no of connected adjacent Weight(v_i,v_j)
9. end if
10. end while

4. EXPERIMENTAL RESULT:

The following example is taken from on **article** published by Khamis et al. [8], there were 41 sentences but we consider only 20 sentences having keywords. The letters A ,B, C and D appears in the four columns as shown in table-1 indicates sentence number, number of different keywords, name of the keywords and keywords related with sentence number and with keywords name respectively.

Table 1: Keyword Abstraction Table

A	B	C	D
0	1	sensor network	
1	1	Sensor	
2	1	Sensor	sensor (1)
3	2	sensor network, sensor	sensor network (0), sensor (1,2)
4	1	network	
5	1	sensor network	sensor network (0,3)
6	2	sensor network, network	sensor network (0,3,5), network (4)
7	2	cover, algorithm	
8	1	vertex cover	
9	1	vertex cover	vertex cover (8)
10	2	vertex cover, cover	vertex cover (8,9), cover (7)
11	3	vertex cover problem, cover, vertex cover	vertex cover (8,9,10), cover (7,10),
12	1	cover	cover(7,10,11)
13	1	problem	
14	2	vertex cover, algorithm	vertex cover (8,9,10,11),Algorithm(7)

15	2	approximation algorithm, vertex cover	vertex cover (8,9,10,11,14)
16	2	algorithm, vertex cover	algorithm(7,14),VC(8,9,10,11,14,15)
17	2	algorithm, vertex cover, vertex cover	Algorithm(7,14,16), C(8,9,10,11,14,15,16)
18	2	algorithm, vertex cover,	Algorithm(7,14,16,17),VC(8,9,10,11,14,15,16,17)
19	2	sensor network, sensor	SN(0,3,5,6),Sensor(1,2,3)

We have now construct two graphs [figure-2 and figure-3] from keyword abstraction table-1 according to rule-1 and rule-2 as discussed in section 3.1.

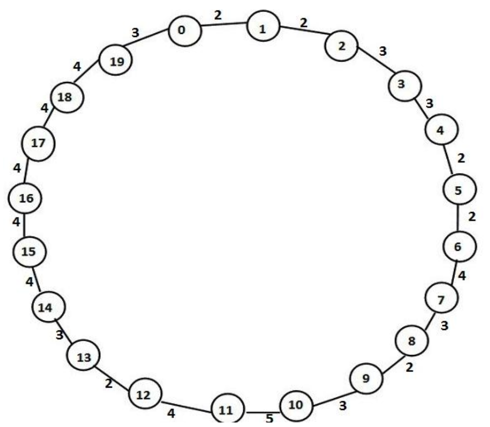

Figure -2

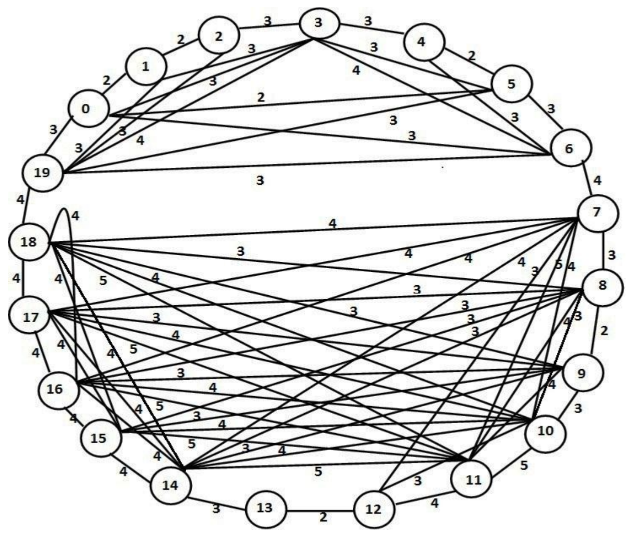

Figure- 3

The graph of figure -2 is a regular graph and figure-3 is a non regular graph which are constructed using rule-1 and rule-2 of section 3.1

Now applying the algorithm for minimum vertex cover in the sentence graph of figure-2 it gives only 10 sentences as output(Table-3) from the 20 input sentences as shown in Table-2.

Input Sentences: (20 sentences) :

Table 2 (Input Sentences)

Sentence No	Selected Sentences
0	Wireless Sensor Networks (WSN) are an emerging communication technology that offers a rich interaction model with the environment
1	Sensors are equipped with data processing and communication capabilities
2	Sensors gather and send data to a base-station either directly or through another sensor node [15]
3	WSN supports nodes mobility and sensors are have limited capabilities
4	Such limitations enforce the need for power-efficient resource management protocols to extend the network lifetime
5	WSNs offer a wide range of possible applications both military and civil [13]
6	Maximizing the network life time is an important issue in sensor networks due to its scarce resources
7	Several schemes were proposed to prolong the life time, one such schemes is using minimal cover set algorithm
8	A vertex cover is a set of vertices *V′*; such that *V′* is a subset of *V*, where *V* is a set of vertices in an undirected graph *G = (V, E),* such that for each edges with two vertices (u, v), either u or v or both must be a member of V’
9	The number of the vertices in V’ represent the vertex cover size
10	The number of the vertices in V’ represent the vertex cover size. For example if we have a graph G with vertices *V = {A,B,C,D,E,F}* and edges *E = {(A,B), (A,C), (A,E), (A,D), (C,D), (C,E), (D,F), (E,F)}*, the graph G has a vertex cover *V’ = {A,D,E}* of size 3 that covers all the edges of the graph, as shown in Figure 1
11	A vertex cover problem is a problem to determine the minimum (optimal) number of vertices that cover all the edges in the graph G; in other words, we want to get the minimum vertex cover size [9]
12	A vertex cover problem is a problem to determine the minimum (optimal) number of vertices that cover all the edges in the graph G; in other words, we want to get the minimum vertex cover size [9]
13	The problem of determining the minimum number of verticesis classified as NP-complete problem [1, 2, 5, 7]
14	Therefore, we can't find an optimal vertex cover size in polynomial algorithm
15	For this reason an approximation algorithm is used to find an approximate solution for the vertex cover

16	This section presents some proposed algorithms to find vertex cover for graph G, with polynomial time complexity
17	The algorithm in [7] finds the vertex cover for graph with n elements and maximum degree Δ, so that the vertex cover size is no more than (n - ceiling (n/ Δ+1)), which is the best possible solution for n and Δ
18	The algorithm firstly defines vertex cover *Ci* for vertex *i* as all vertices except *vi*, then it is search for removable vertices in vertex cover *Ci* to decrease vertex cover size
19	In [4], the authors focus on the communication issues by assuming the wireless sensor network consist of two types of sensor devices: coverage sensors and communicating sensors

First Experimental Results:

First Experiment Output for Figure-2

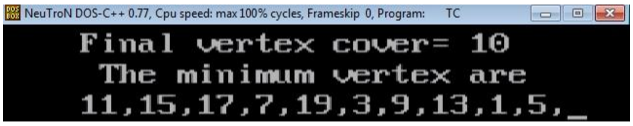

Figure-4

From the above experiments (Figure-4) we may analysis that the graph(Figure-2) contains 20 vertices and the Minimum Vertex Cover is 10 and the vertices are 11,15,17,7,19,3,9,13,1,5. We arrange them in ascending order using bubble sort and then nodes are found 1,3,5,7,9,11,13,15,17,19 . Therefore, after implementation of the Minimum Vertex Cover algorithm which summaries the article is as follows in Table-3.

Output Sentences:- Output Sentences of 1st Experiment of Figure-2 (Summarizes Article of 10 Sentences)

Table-3

Sentence No	Selected Sentences
1	Sensors are equipped with data processing and communication capabilities
3	WSN supports nodes mobility and sensors are have limited capabilities
5	WSNs offer a wide range of possible applications both military and civil [13]
7	Several schemes were proposed to prolong the life time, one such schemes is using minimal cover set algorithm
9	The number of the vertices in V' represent the vertex cover size
11	A vertex cover problem is a problem to determine the minimum (optimal) number of vertices that cover all the edges in the graph G; in other words, we want to get the

	minimum vertex cover size [9]
13	The problem of determining the minimum number of vertices is classified as NP-complete problem [1, 2, 5, 7]
15	For this reason an approximation algorithm is used to find an approximate solution for the vertex cover
17	The algorithm in [7] finds the vertex cover for graph with n elements and maximum degree Δ, so that the vertex cover size is no more than (n - ceiling (n/ Δ+1)), which is the best possible solution for n and Δ
19	In [4], the authors focus on the communication issues by assuming the wireless sensor network consist of two types of sensor devices: coverage sensors and communicating sensors

Second Experimental Results:

Second Experiment Output for graph of Figure-3.

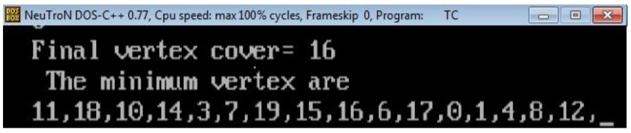

Figure-5

From the above experiments (Figure-5) we may analysis that the graph(Figure-3) contains 20 vertices and the minimum vertex cover as found as 16 and the vertices are 11,18,10,14,3,7,19,15,16,6,17,0,1,4,8,12. Again using bubble sort we arrange the sentences in ascending order as 0,1,3,4,6,7,8,10,11,12,14,15,16,17,18,19. So after implementation of the Minimum Vertex Cover algorithm which summaries the article is as follows in Table-4.

Output Sentences: Output Sentences of 2nd Experiment of Figure-3 (Summarizes Article of 16 Sentences)

Table-4

Sentence No	Selected Sentences
0	Wireless Sensor Networks (WSN) are an emerging communication technology that offers a rich interaction model with the environment
1	Sensors are equipped with data processing and communication capabilities
3	WSN supports nodes mobility and sensors are have limited capabilities
4	Such limitations enforce the need for power-efficient resource management

	protocols to extend the network lifetime[15]
6	Maximizing the network life time is an important issue in sensor networks due to its scarce resources
7	Several schemes were proposed to prolong the life time, one such schemes is using minimal cover set algorithm
8	A vertex cover is a set of vertices *V'*; such that *V'* is a subset of *V*, where *V* is a set of vertices in an undirected graph *G = (V, E),* such that for each edges with two vertices (u, v), either u or v or both must be a member of V'
10	The number of the vertices in V' represent the vertex cover size. For example if we have a graph G with vertices *V = {A,B,C,D,E,F}* and edges *E = {(A,B), (A,C), (A,E), (A,D), (C,D), (C,E), (D,F), (E,F)}*, the graph G has a vertex cover *V' = {A,D,E}* of size 3 that covers all the edges of the graph, as shown in Figure 1
11	A vertex cover problem is a problem to determine the minimum (optimal) number of vertices that cover all the edges in the graph G; in other words, we want to get the minimum vertex cover size [9]
12	For example,the minimum cover is 2, as shown in Figure 1
14	Therefore, we can't find an optimal vertex cover size in polynomial algorithm
15	For this reason an approximation algorithm is used to find an approximate solution for the vertex cover
16	This section presents some proposed algorithms to find vertex cover for graph G, with polynomial time complexity.
17	The algorithm in [7] finds the vertex cover for graph with n elements and maximum degree Δ, so that the vertex cover size is no more than (n - ceiling (n/Δ+1)), which is the best possible solution for n and Δ
18	The algorithm firstly defines vertex cover *Ci* for vertex *i* as all vertices except *vi*, then it is search for removable vertices in vertex cover *Ci* to decrease vertex cover size
19	In [4], the authors focus on the communication issues by assuming the wireless sensor network consist of two types of sensor devices: coverage sensors and communicating sensors

Implementation Process of Algorithm 3.4:

In sentence selection we use Minimum Vertex Cover and which is a NP complete problem. First we select a node which contains maximum degree. If two or more nodes contains more than same degree than select the nodes which contains maximum degree and maximum weight and if degree and weight is same then select any one of them. After selecting the nodes we remove all the incident edges. When we remove the edges then decrease the degree and weight of adjacent nodes. This process is continues until and unless all the edges are not selected.

ANALYSIS AND RESULTS

A) In our first experiments (Minimum Vertex Cover Algorithm) if we enter the adjacency matrix of the first graph then we find that only 10 sentences instead of 20 sentences and covers all the sentences. So we can summarize the article using only 10 sentences instead of 20 sentences.

B) In our experiments (Minimum Vertex Cover Algorithm) if we enter the adjacency matrix of the second graph then we find that only 16 sentences instead of 20 sentences and covers all the sentences. So we can summarize the article using only 16 sentences instead of 20 sentences.

CONCLUSIONS:

From the above analysis (A&B) it has been found that one can find the minimum number of sentences which summarize the article and this will give the actual meaning of original article. It is observed that the graph (experiment A) discussed in figure-3 , is more efficient, which gives the concise and more details summary of the text than the experiment (A) discussed in figure-2 . But the second experiment(B) is time consuming than the experiment(A). So one can use our algorithm and summarizes any article any paper or any documents without changing the original meaning.

REFERENCES

[1] Hiroya Takamura andManabu Okumura "Text Summarization Model based on Maximum Coverage Problem and its Variant", Proceedings of the 12th Conference of the EACL, pages 781–789,Athens, Greece, 2009.

[2] Lucas Antiqueira, Osvaldo N. Oliveira Jr, Luciano da Fontoura Costa and Maria das Graças Volpe Nunes," A complex network approach to text summarization", Information Sciences, 584–599, 2009

[3] Vishal Gupta and Gurpreet Singh Lehal "A Survey of Text Summarization Extractive Techniques", Journal Of Emerging Technologies in Web Intelligence, VOL. 2, NO. 3, AUGUST 2010.

[4] Asher Stern and Ido Dagan" A Modular Open-Source System for Recognizing Textual Entailment", Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 73-78, Jeju, Republic of Korea, 2012.

[5] Rasim ALGULIEV and Ramiz ALIGULIYEV, "Evolutionary Algorithm for Extractive Text Summarization", Intelligent Information Management, 2009

[6] Anand Gupta, Manpreet Kaur, Adarsh Singh, Aseem Goel and Shachar Mirkin, "Text Summarization through Entailment-based Minimum Vertex

Cover", Proceedings on the Third Joint Conference on Lexical and Computational Semantics, pages 75-80, Dublin, Ireland, 2014.

[7] Kamal Sarkar," Using Domain Knowledge for Text Summarization in Medical Domain", International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009

[8] Yaser Khamayseseh, Wail Mardini , Anas Shantnawi, "An Approximation Algorithm for Vertex Cover Problem" International Conference on Computer Networks and communication Systems, 2012, Singapore.

[9] S. Balaji, V. Swaminathan and K. Kannan, "Optimization of Unweighted MinimumVertex Cover", World Academy of Science, Engineering and Technology, 43 2010.

[10] Chandan Kalita, Navanath Saharia, and Utpal Sharma, "An extractive approach of Assamese text summarization", in proceedings of Sixth Global WordNet Conference, Matsu, Japan, 2012.

[11] Atowar ul Islam, Bichitra Kalita and AnupamDutta ," Minimum Vertex Cover of Different Regular Planar Graph and Its Application" International Journal of Mathematical Archive-5(10), 2014175-184 Available online through www.ijma.info ISSN 2229 – 5046.www.ijma.info ISSN 2229 – 5046.

[12] Sparck Jones, K. (1995). Discourse modelling for automatic summarising. In Travaux du Cercle Linguistique de Prague (Prague Linguistic Circle Papers) 1995 (Vol. 1, pp. 201–227).

[13] Sparck Jones, K. (1999). Automatic summarising: Factors and directions. In I. Mani & M. T. Maybury (Eds.), Advances in automatic text summarisation (pp. 1–14). Cambridge, MA: MIT Press.

[14] Sparck Jones, K. (2001). Factorial summary evaluation. In Proceedings of the DUC 2001, 2001.

[15] Sparck Jones, K. (2007). Automatic summarising: A review and discussion of the state of the art. Technical Report 679, Computer Laboratory, University of Cambridge, 2007.

[16] Sparck Jones, K., & Endres-Niggemeyer, B. (Eds.).,(1995). Information Processing and Management, 31(5), 625–784 (Special issue).

[17] Sparck Jones, K., & Galliers, J. R. (1996). Evaluating natural language processing systems. Berlin Springer.

[18] Hahn U, Mani I (2000) The challenges of automatic summarization,IEEE Comput 3(11):29–36.

[19] Kupiec J, Pedersen J, Chen F (1995) A trainable document summarizer, In: Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'95), Seattle, WA, USA, pp 68–73.