Extractive Text Summarisation using Graph Triangle Counting Approach: Proposed Method

Yazan Alaya AL-Khassawneh*, Naomie Salim , Obasa Adekunle Isiaka

Faculty of Computing, University Technology Malaysia, Malaysia

Abstract

Currently, with a growing quantity of automated text data, the necessity for the construction of Summarisation systems turns out to be vital. Summarisation systems confine and condense the mainly vital ideas of the papers and assist the user to find and understand the foremost facts of the text quicker and easier from the dispensation of information. Compelling set of such systems are those that create summaries of extracts. This type of summary, which is called Extractive Summarisation , is created by choosing large significant fragments of the text without making any amendment to the original. One methodology for generating this type of summary is consuming the graph theory. In graph theory there is one field called graph pruning / reduction, which means, to find the best representation of the main graph with a smaller number of nodes and edges. In this paper, a graph reduction technique called the triangle counting approach is presented to choose the most vital sentences of the text. The first phase is to represent a text as a graph, where nodes are the sentences and edges are the similarity between the sentences. The second phase is to construct the triangles, after that bit vector representation and the final phase is to retrieve the sentences based on the values of bit vector.

Keywords. Extractive Summarisation ; Triangle Counting; Graph-Based Summarisation

1 Introduction

With the quick expansion of the Internet, a vast quantity of social contact is offered and has become reachable online. Humans extensively use the internet to find information through proficient Information Retrieval (IR) tools such as Google, Yahoo, AltaVista, and so on. People do not have enough time for reading everything, and so they need to make vital decisions dependent on the information available there. The need for new procedures to assist people achieve and absorb the vital information from these sources has become increasing imperative as the quantity and accessibility of textual information increases. The necessity for computerized summaries is becoming more visible.

*Corresponding author: yakhassawneh@yahoo.com

Improvement in text summarisation and filtration will not only permit the improvement of superior repossession systems, but also support access and analyse information on the basis of the text in multiple ways to help create a separate novelist and permanent access the system. There are diverse explanations for summary. Authors in [1] describes the summary as a purview that is founded and depends on one or more purviews; it keeps the largely vital acquaintance of the original documents and its tenor is not exceeding half of the original documents. Mani [2], defines the summarisation of a text as the procedure to find the major significant content and a procedure for discovering the major foundation of information, and presenting it as an ephemeral text in the predefined prototype.

Text summarising has three main steps [3]. Those steps are identifying the topic, interpretation, and, finally, summary generation. Through identifying the topic, the most important information in the document is recognized. Most systems allocate diverse preference to diverse fragments of the text (words, phrases, and sentences); then the scores of each part are mixed to discover the totality for a part. At the end, the system includes the N highest degree fragments in a concluding summary. Cue Axioms, content counting, and word existence [4] are some examples of the numerous techniques for topic identification which have been studied.

Interpretation step related to Abstract summaries. In this step, associated issues are joined with a view to shape brief common content [5] and extra expressions are ignored. Concluding the topics is complicated; consequently most of the systems produce an extractive summary.

In the step of generating the summary, a text generation procedures are used by the system. This step contains a collection of diverse generation procedures from extremely straightforward word or expression printing to more complicated expression assimilation and sentence generation. In another meaning, this step is to generate the natural language which is easily understood by the users.

Based on the type of generated summary, the summarisation systems are categorized. This work focused on extractive summaries. Extractive summaries are established by elicitation of main text slices (sentences or paragraphs) from the document, dependent on analytical statistics of singular or diverse surface stage lineaments like word/phrase occurrence, position or sign words for finding the sentences that must be extracted. The "mainly vital" tenor is dealt with like the "most common" or the "most positively situated" tenor. This method therefore avoids labour on profound text perceptive. It is straightforward and simple to apply. So far, many different methods proposed for the selection of the most important parts of texts, such as statistical approach, consisting of a compilation scheme similarities [6], Site plan [7] form frequency [8], and the query system based on TF [9] linguistic approach It consists of a split graph theory, WordNet, and Lexical Series. According to [10] and [11], graph is known as a group of nodes and a set of edges joining pair's numbers of nodes. While talking about database, the nodes mean distinct parts, while the edges show the importance of the relationship among these parts. Triangle Counting Approach is a procedure used to prune the graph, and therefore the aim of this study is to use this technique to create a summary.

The remainder of this paper is structured as follows: Section 2 will debate related work on graph based text summarisation and the triangle counting method. The proposed algorithm is given in section 3. Evaluation measurements will be discussed in section 4. A conclusion will be given in section 5.

2 Related Work

2.1 Graph-Based text Summarisation

Consuming graphs for presenting the construction of texts will assist us in superior recognition of linkages among diverse parts. [12].

Inderjeet Mani et al [13] used graph entry node sets to represent topics in 1997 by corresponding edges with semantic relations between items. In their algorithm, a spreading activation technique was used for discovering nodes that had relation with the core themes by treating the nodes with similar meanings to the topic terms as the graph entry points referred to as activating nodes. Their method was to assign weights to these nodes exponentially in order to get the distance between the nodes as well as the activating nodesŠ weight decaying function. This enabled them to determine output nodes threshold number by calculating the weight of a neighbour node as a function of the activating node and link weights respectively and getting the neighbor of starting nodes to the output.

According to [14], Summarisation of multi-document extractive is dependent on how the most significant entences in a document are recognized through sentence centrality. In their new approach, they made use of the hub-authority framework to unite text content with cues like ¸ Ssentence lengthˇT, ¸Scue phraseˇT, and ¸ Sfirst sentenceˇT in order to examine sub-topics in a multi-document and determine their graph-based sentence ranking algorithm by conveying the sub-topics features into the graph-based algorithms. The summary is then derived based on the sentence ranking score of the whole sentences. This traditional graph-based technique consists of two important facts: firstly, it unites some of the cue characteristics with the content of the text; and secondly, it employs the graph-based sentence ranking algorithm for discovery of sub-topics. This method has been validated using data from DUC 2004 and showed that the combination of interior and exterior features is a superior design within the framework of the Hub/Authority for effectively ranking graph scheme in multi-document standard text abstraction.

Google's PageRank [15] or HITS [16] were initially established as tools to reconnoitre the link-structure to rank Web pages. Subsequently they have been used fruitfully in other fields, e.g. social networks, citation analysis, etc. In graph ranking procedures, the significance of a vertex inside the graph is recursively calculated from the whole graph.

Marina and Mark [17] introduced two new approaches for determining the associated keywords for use in extractive summarisation of text documents, namely supervised and unsupervised techniques. For the graph-based method, syntactic illustration of text improves the conventional vector-space illustration by considering certain basic document landscape. In the supervised method, writers direct the categorisation procedures of a summarized anthology of documents with the function of encouraging archetypal keyword detection. While in the unsupervised method, a

HITS algorithm was used on text graphs based on the postulation that the document keywords are represented by the top-ranked nodes.

Text-Rank, a scheme for unsupervised extractive summarisation that depends on the purpose of repetitive graph based ranking procedures to graph consistent organisation of a text, [8]. An imperative attribute of the scheme is that it is highly transportable to new languages or domains, because it does not rely on some language- particular information property or any physically built preparation data. It is presented by the author that repetitive graph-based ranking procedures work fine on the mission of extractive summarisation because they do not depend only on the narrow environment of a text element (vertex), but yield the info repeatedly gleaned from the complete text (graph) into account.

Ohm Sornil et al. [18] used the Hopfield Network algorithm for joining graph based and content based features to create an automatic summarisation system, where every node is checked for similarity and node weights were joined for every singular node. In the first phase, content-based feature vectors form the slices. After that the slice-feature matrix focussed on a lesser dimensional matrix to discover unseen relationship outlines and decrease little differences in slice features by the use of Singular Value Decomposition (SVD). In the second phase, the graph was constructed and the slices form the nodes, and the edges are the relations between two slices whose match scores are higher than the threshold. The undirected text graph, fabricated from cosine likeness, is considered an arrangement that offers the greatest summarisation presentation.

Stochastic graph-based method was suggested in [19] for calculating comparative significance of documentary components for Natural Language Processing (NLP). A novel technique named Lex-Rank, for calculating sentence significance founded on the idea of eigenvector significance in a graph illustration of sentences. An adjacency matrix created from the sentences in the planned method is used for a linking matrix founded on intra-sentence cosine similarity. Also it is shown that degree-based methods (containing Lex-Rank) perform better than centroid-based methods.

Authors in [20] discussed a similarity graph based method to multi-document summarisation. Recommending an incorporated construction and bearing in mind together information affluence and information originality of a sentence founded on sentence similarity graphs. Authors in [21] exhibited how a meta-summariser hooked on covered purpose graph-based approaches for single-document summarisation, may be impressed into a fruitful technique for multi-document summarisation. They represented the graphs as: (a) an undirected graph; (b) a directed weighted graph where the direction of edges are group from sentence to sentence and pursue in the text (directed forward); or (c) a directed weighted graph with the direction of edges group from one sentence to preceding sentences in the text (directed backward). Multi-document summaries for a text group were created using a "meta" summarisation system. For every text in the group of documents, a single document summary was created using one of the graph-based ranking algorithms. A "summary of summaries" was formed using the identical or an altered ranking algorithm.

Kokil Jaidka et al [22] in 2010 produced a pioneering Summarisation method to create literature review of research papers that impressionist the features of human literature reviews. To identify the focus of the human variety of information and review, an exploration was done here. Some key demands of the intriguing technology was high, as: i), where scholars identify this info? ii) What is the info that they have chosen iii) how to fulfil the functions of literature review? Advanced techniques in this scheme are often in the assortment and integration of information to determine the phase information from a variety of semantic levels, and the role of the implementation phase which will be recruiting literature review oratory. In this suggested method, three classes of discourse construction were acknowledged. For sentence S level, an XML representation to explain the legal structure of an XML document used to describe the construction of the literature review, along with predictable characteristics and associations classified is created. A graphic demonstration of the rhetorical relations as a tree structure among the elements that consist of text was characterized for the level of clause and level of intra-clause features. A number of methods were used to determine the projections of this XML tree or graph construction to create a review of the literature on as: A) Improve rates of comparative information for information regarding the association between the immediate area of application and content source, B) semantic similarity measures, C) Connotation amongst the candidate range and source content.

1.2 Triangle Counting Approach

Counting triangles is an imperative area in graph mining. Two commonly used measurements in multifarious network investigation that necessitate the amount of triangles are the transitivity ratio of the graph and clustering coefficients. Moreover, numerous attractive graph mining enforcement depend on calculating the number of triangles in a large-scale graph, such as link commendation in virtual social networks, discovery of spamming commotion and revealing the hidden thematic structure of the web.

Several projects have been implemented to discover or develop fresh procedures for calculating triangles in graph datasets. One approach in [23] was suggested. The authors reduced the problem of triangle counting efficiency by resembling instants for a flow of node augments. They then used an algorithm offered in [27] to continue. For the same reason, [25] suggested a sampling bounding the two – space procedures to approximate the amount of triangles. Again, simple procedures basic sampling. For the edge of the flow illustration, are sampled arbitrarily on the edge and node and then the process of examination if they form a triangle. Monte Carlo procedures are algorithms for the prior art, but that are still in need of the graphics to be quite thick in order to obtain a good approximation.

Schank & Wagner [26] introduced an algorithm known as the node-iterator which has an execution time $O\left(nd_{\max}^{2}\right) \subset O\left(n^{3}\right)$. The algorithm listing-ayz is the record version of the majority proficient counting algorithm [27]. It has a running time of $O\left(m^{3/2}\right)$ and uses the idea of centres. It receipts a node of minimum degree, calculates its triangles in the identical manner as in node-iterator, and then eliminates the node from the graph. The execution time is $O\left(nc_{\max}^{2}\right)$, where c (v) is the core number of node v. Since the node-iterator central is an enhancement over the listing-ayz the execution time of the node-iterator-core is also $O\left(m^{3/2}\right)$.

Charalampos, E [28] proposed a new, highly reliable, fast, and parallelisable algorithm to count triangles. The parallelising was vital because it provided an opportunity to mine a large number of graphs using corresponding architectures such as e map/reduce (Hadoop). [29].The proposed method uses two algorithms as follows:

Theorem 1: (Eigen Triangle) The total number of trigonometric functions in the graph commensurate with the cube of the total value of a self-neighbourliness matrix, as represented below:

$$\Delta(G)=\frac{1}{6}\sum_{i=1}^{n}\lambda_i^3 \qquad (1)$$

Theorem 2: (Eigen Triangle Local) The number of triangulations that participate in the node can be counted from the cube of the eigenvalues from the matrix generated.

$$\Delta_i=\frac{\sum_j \lambda_j^3 u_{i,j}^2}{2} \qquad (2)$$

Where $u_{i,j}$ is the $j-th$ entry of the $i-th$ Eigen vector.

Ilaria Bordino et al, [30] conducted research on the counting of triangle algorithms and proposed a collection of methods based on random sampling for approximating with high accuracy the number of triangles in 3-noded and 4-noded minors of directed and undirected graphs. The algorithm was tested on various networks from 10 different fields. Based on the rate of reoccurrence of all the minors, they have also proposed an effective network clustering algorithm.

Charalampos, E et al [31] conducted research on triangle counting algorithms and recommended a new approach. They devised a method for the sparsification of a graph by changing it into an alternative weighted graph. The newly converted graph will have a smaller number of edges. The triangles would be calculated with the EIGENTRIANGLE method. Their method combines the concepts of the EIGENTRIANGLE with the Achlioptas-McSherr algorithms.

Charalampos, E et al [32] examined a new sampling algorithm for counting triangles. They executed the method on large networks, and proved speed-ups that are 70,000 times faster in counting triangles. The performance of the algorithm is more precise when the densities of triangles are mild.

Haim Avron, [33] presented a high level of parallel development and a fresh indiscriminate procedure for estimating the amount of triangulations in an undirected graph. The procedure uses the popular Monte-Carlo simulation to count the number of triangles. Each sample needs $O(|E|)$ time and

$$O(\varepsilon^{-2}\log(1/\delta)\rho(G)^2) \qquad (3)$$

samples are needed to ensure a (ε,δ) rough calculation, where $\rho(G)$ is a quantifier of the triangle scatter of G. The ($\rho(G)$ is not essentially small. The algorithm needs only $O(|V|)$ space to work professionally.

The author provided experiences to prove that in this pursuit usually only $O(\log^2|V|)$ samples are critical to the accurate estimation of graphs. This algorithm is more efficient than other relevant state-of-the-art algorithms for counting triangles, especially in speed and accuracy. Unfortunately this algorithm is parallel only when the critical path of $O(|E|)$ is attainable on as little as $O(\log^2|V|)$ processors.

The Eigen Triangle and Eigen Triangle Local algorithms were proposed in [34] to calculate the full quantity of triangles and the quantity of triangles were every node shares with an undirected graph. Those procedures are effective for all types of graphs, except when the graphs have certain spectral properties. The authors confirmed this pragmatically by conducting 160 tests on diverse kinds of actual networks. They observed important speedups between 34× and 1,075 × faster performances with 95% precision compared to an uncomplicated counting algorithm.

Based on the work in [35], the authors projected an approach that can adjust a graph by switching it to an alternative graph with fewer edges and nodes. The major aspiration of the study was to utilise the approach of counting the triangle for graph-based association ruled mining. A triangle counting technique for graph-based association rules mining was suggested to reduce the graph in the process of searching for common items. A combination between the triangle counting with one of the graph-based ARM methods was done. It involves four imperative steps; data demonstration, triangle production, bit vector demonstration, and triangle combination with the graph-based ARM technique. The rendering of the suggested technique was compared with the main graph-based ARM. Experimental outcome showed that the suggested technique shortened the accomplishment time of rules production and created fewer of rules with greater assurance.

3 Methodology

In this work, a triangle counting algorithm for graph-based representation of text is suggested in order to reduce the graph in the seeking for the best representation of a sub- graph. Three important stages are involved, i) representing text graphically, where modes in the graph represent sentences in the text and edges represent relationships between the sentences. In this paper, the relationship is the similarity between sentences, ii) discover the number of triangles in the graph by using an adjacency matrix and De-Morgan's laws, and represent them as a sub-graph of the main graph, iii) by using bit vector values for each edge in (ii) we can decide which nodes are the most important and then represent every node as sentence in the text to form the summary.

To represent the text as graph, a matrix was applied to indicate an adjacency construction. For example, if G was a directed graph then $A_{ij} = 1$ if have a relation

pointed from node i to node j occurred; if not $A_{ij} = 0$ (no relation occurred). If the nodes in the graph are numbered 1,2,…m, then, the adjacency matrix will be of the type m×m. If A×A×…×A (p terms, p←m) was calculated, the non-zero records specified those vertices which were combined by a route of length p; definitely the worth of the (i,j)th record of Ap donated the amount of routes of size p from the vertex i to vertex j.
For example if we have a simple graph represented in Fig.1.

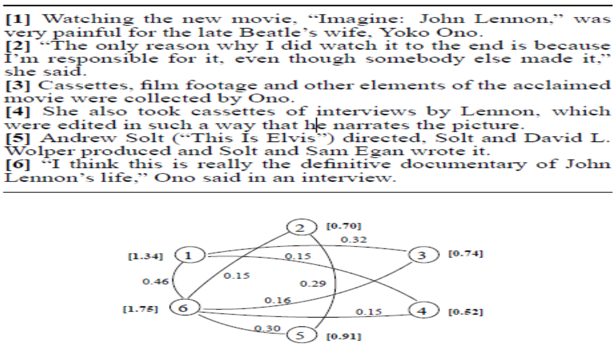
[1] Watching the new movie, "Imagine: John Lennon," was very painful for the late Beatle's wife, Yoko Ono.
[2] "The only reason why I did watch it to the end is because I'm responsible for it, even though somebody else made it," she said.
[3] Cassettes, film footage and other elements of the acclaimed movie were collected by Ono.
[4] She also took cassettes of interviews by Lennon, which were edited in such a way that he narrates the picture.
[5] Andrew Solt ("This Is Elvis") directed, Solt and David L. Wolper produced and Solt and Sam Egan wrote it.
[6] "I think this is really the definitive documentary of John Lennon's life," Ono said in an interview.

Fig.1. Text Graph representation

Then the adjacency matrix for this data will be as in Table 1.

	1	2	3	4	5	6
1						
2						
3	1					
4	1					
5		1				
6	1	1	1	1	1	

TABLE I Adjacency Matrix representation

The triangle counting step was accompanied based on the represented data where the nominated triangles were achieved. In this work, the De – Morgan's Laws were applied to produce the triangles where given two conjugation edges XY and YZ unite with an aggregation XY^YZ then we get XZ.

By the use of De-Morgan's laws, we could know the number of triangles in addition to the nodes and edges form these triangles:

13^36 ⟶ 16, 14^46 ⟶ 16, 25^56 ⟶ 26

Here, we can see that the triangles in our example are formed from these links. To find most vital sentences the Bit–vector exemplification was adapted in this work to symbolise the pruned graph / graph with triangles, gotten from the preceding section.

Bit vector	Bit vector value
BV1	10110000
BV2	00000101
BV3	01100000
BV4	00011000
BV5	00000110
BV6	01101011

TABLE 2 Bit – Vector Values

From the above table we can note that the importance of the sentences depends on the value of the bit vector, the highest value is the most important.

4 Performance Metrics

To estimate a text summarisation scheme, two extensively used measurements are Recall and Precision [18]. Those two measurements were used for assessing extractive summaries. Recall is defined as the division of sentences selected by somebody that at the same time were also acceptably recognized by the scheme. A person is requested to choose clauses that appear to greatest relocate the denotation of the text to be summarised. Then sentences mechanically nominated by the scheme are assessed in contradiction of the human choice.

$$Recall = \frac{|system\ summaries| \cap |human\ summaries|}{|human\ summaries|} \quad (4)$$

And Precision is the division of scheme sentences that were truthful.

$$Precision = \frac{|system\ summaries| \cap |human\ summaries|}{|\ system\ summaries} \quad (5)$$

$$F = 2* \frac{Precision * Recall}{Precision + Recall} \quad (6)$$

ROUGE-N [36] is an alternative standard which is extensively used in assessing summaries. ROUGE-N is calculated as follows:

$$\mathrm{ROUGE-N} = \frac{\sum_{S\in\{ReferenceSummarization\}n\_gram\in S} \sum Count_{match}(n\_gram)}{\sum_{S\in\{ReferenceSummarization\}n\_gram\in S} \sum Count(n\_gram)} (7)$$

Where n represents the length of the n-gram, n_gram, and $Count_{match}$ (n_gram) is the supreme amount of n-grams co-occurring in an applicant summary and a group of reference summaries. Clearly that ROUGE-N is a recall-related metric because the primarily of the equation is the overall sum of the amount of n-grams happening on the reference summary side.

5 Concluding Remark

Automatic text summarisation is an outline of a foundation text by a machine to show the most significant information in a shorter version of the main text while still conserving its major semantic content and helps the user to rapidly recognise huge amounts of information. The majority of existing automatic text summarisation approaches extracts the most significant information from source documents. Conventionally, automatic text summarisation systems mine the sentences from the source documents depending on their importance to the documents. The summarisation systems evaluate the weights of diverse extraction features for every text element then the weights of sentence are joined as the general weight of the text element. Finally, the sentences with the highest weight will be extracted. This study has shown that one graph can be converted into another graph, with significantly smaller number of edges by counting the number of triangles. The use of Adjacency Matrix Representation is simple and this feature has contributed towards shorter execution times.

ACKNOWLEDGMENT

This work is supported by Ministry of Higher Education (MOHE) and Research Management Centre (RMC) at the Universiti Teknologi Malaysia (UTM) under Research University Grant Category (VOT: Q.J130000.2528.07H89).

References

1. Frankel, David S. “Model Driven Architecture Applying Mda”. John Wiley & Sons, 2003.
2. Mani, Inderjeet. Automatic summarization. Vol. 3. John Benjamins Publishing, 2001.
3. Lin CY, Hovy EH “Identify topic by position”. In: Proceedings of 5th conference on applied natural language processing, March 1997
4. Mazdak N A Persian text summarizer, master thesis, department of linguistics, Stockholm University, Jan 2004
5. Wills, Rebecca S. "Google’s pagerank." The Mathematical Intelligencer 28.4 (2006): 6-11.
6. Kyoomarsi, F., Khosravi, H., Eslami, E., Dehkordy, P. K., & Tajoddin, A. “Optimizing Text Summarization Based on Fuzzy Logic”. In ACIS-ICIS (pp. 347-352), 2008.
7. Kupiec, Julian M., and Hinrich Schuetze. "System for genre-specific summarization of documents." U.S. Patent No. 6,766,287. 20 Jul. 2004.
8. Rada M. “Graph-based ranking algorithms for sentence extraction, applied to text Summarisation” , annual meeting of the ACL 2004, pp 170–173
9. Patil, K., & Brazdil, P. SumGraph: text summarization using centrality in the pathfinder network. International Journal on Computer Science and Information Systems, 2(1), 18-32. 2007.
10. McKee, Terry A., and Fred R. McMorris. “Topics in intersection graph theory”. Vol. 2. Siam, 1999.

11. Thomas, J. A., & Thomas, J. A. (2006). Elements of information theory. New York: Wiley.
12. Saeedeh G, Mohsen AS, Bahareh G "A comprehensive survey on text Summarisation systems''. CSA 2:462–467. 2009.
13. Mani, I., & Bloedorn, E. (1997). "Multi-document summarization by graph search and matching". arXiv preprint cmp-lg/9712004.
14. Zhang, J., Sun, L., & Zhou, Q. "A cue-based hub-authority approach for multi-document text summarization". In Natural Language Processing and Knowledge Engineering, IEEE NLP-KE'05. Proceedings of IEEE International Conference on (pp. 642-645). IEEE. 2005
15. Brin, S., & Page, L. "The anatomy of a large-scale hypertextual Web search engine". Computer networks and ISDN systems, 30(1), 107-117. 1997
16. Kleinberg, J. M. "Authoritative sources in a hyper-linked environment". In Journal of the ACM, 46(5). 604-632. 1999
17. Litvak, M., & Last, M. "Graph-based keyword extraction for single-document summarization". In Proceedings of the workshop on Multi-source Multilingual Information Extraction and Summarization (pp. 17-24). 2008
18. Sornil, O., & Gree-Ut, K. "An automatic text summarization approach using content-based and graph-based characteristics". In Cybernetics and Intelligent Systems, 2006 IEEE Conference on (pp. 1-6). IEEE. 2006
19. Erkan, G., & Radev, D. R. "LexRank: Graph-based lexical centrality as salience in text summarization". J. Artif. Intell. Res.(JAIR), 22(1), 457-479. 2004
20. Wan, X., & Yang, J. "Improved affinity graph based multi-document summarization". In Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers (pp. 181-184). 2006
21. Mihalcea, R., & Tarau, P. "A language independent algorithm for single and multiple document summarization". 2005
22. Kokil. J, "Multidocument Summarisation of Information Science Research Papers," Bulletin of IEEE Technical Committee on Digital Libraries: JCDL/ICADL Doctoral Consortium Issue, vol. 6, issue 2, 2010.
23. Bar-Yossef, Z., Kumar, R., & Sivakumar, D. "Reductions in streaming algorithms, with an application to counting triangles in graphs". InProceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms (pp. 623-632).2002
24. Alon, N., Matias, Y., & Szegedy, M. "The space complexity of approximating the frequency moments". In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (pp. 20-29). 1996
25. Buriol, L. S., Frahling, G., Leonardi, S., Marchetti-Spaccamela, A., & Sohler, C. "Counting triangles in data streams". In Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (pp. 253-262). 2006
26. Schank, T., & Wagner, D. "Finding, counting and listing all triangles in large graphs, an experimental study". In Experimental and Efficient Algorithms(pp. 606-609). Springer Berlin Heidelberg. 2005
27. Alon, N., Matias, Y., & Szegedy, M. "The space complexity of approximating the frequency moments". In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (pp. 20-29). 1996

28. Tsourakakis, C. E. “Fast counting of triangles in large real networks without counting: Algorithms and laws”. In Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on (pp. 608-617). 2008
29. Dean, J., & Ghemawat, S. “MapReduce: simplified data processing on large clusters”. Communications of the ACM, 51(1), 107-113. 2008
30. Bordino, I., Donato, D., Gionis, A., & Leonardi, S. “Mining large networks with subgraph counting”. In Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on (pp. 737-742). 2008
31. Tsourakakis, C. E., Kang, U., Miller, G. L., & Faloutsos, C. “Doulion: counting triangles in massive graphs with a coin”. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 837-846). ACM. 2009
32. Tsourakakis, C. E., Kang, U., Miller, G. L., & Faloutsos, C. “Doulion: counting triangles in massive graphs with a coin”. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 837-846). ACM. 2009
33. Avron, H. “Counting triangles in large graphs using randomized matrix trace estimation”. In Workshop on Large-scale Data Mining: Theory and Applications. 2010
34. Tsourakakis, C. E. “MACH: Fast Randomized Tensor Decompositions”. In SDM (pp. 689-700). 2010
35. Al-Khassawneh, Y. A. J., Bakar, A. A., & Zainudin, S. “Triangle Counting Approach for graph-based Association Rules Mining”. In Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on (pp. 661-665). IEEE. 2012
36. Lin, C. Y. “Rouge: A package for automatic evaluation of summaries”. In Text Summarization Branches Out: Proceedings of the ACL-04 Workshop (pp. 74-81). 2004