Trends in Multi-document Summarization System Methods

Abimbola Soriyan
Dept. of Comp. Sci. & Engineering
Obafemi Awolowo University
Ile-Ife, Nigeria

Theresa Omodunbi
Dept. of Comp. Sci. & Engineering Obafemi Awolowo University
Ile-Ife, Nigeria

ABSTRACT

Information is knowledge if it is rightly applied. Information are stored with different formats in databases but retrieving such from different documents has been a challenge. People want ready-made information for the purpose of decision making in minimal time and thereby crave for summary of information. Automatic summarization helps in mining data and delivering timely and cogent information to users. These systems attempt to address the issue of data mining using different summarization methods. This paper discusses existing methods and state of the art in automatic summarisation system from recent articles. Achievement and challenges involve are also discussed.

General Terms

Automatic summarization system

Keywords

Data mining, summarization, information retrieval, multi-document.

1. INTRODUCTION

Summaries are key important aspects in our day to day activities. One way or the other, people try to listen or watch the glimpse of an event, news, story etc. Summaries have become part of our daily activities. Headlines of news, newsfeed, abstracts of books, summary of result on investigation in the hospital, updates on news etc. More importance is now attached to summaries to the extent that automatic summarisation is considered as a contemporary research in computing. Automatic Text summarization is the process of automatically creating a compressed version of a given text that provides useful information for the user. [1] Also, the fact that information keeps increasing and people are doing away with traditional method (paper based) of storing information, automatic summarization is one of the ways to retrieve information from electronic database especially the internet. Automatic summarization is the act of retrieving cogent information from document(s) and present it as summary [2]. With the new technology of cloud computing [3], [4], [5], data warehousing [6], [7] and big data [8] where different databases and repositories can be accessed, getting correct information has been difficult. Both relevant and irrelevant information are opened to users but to get the exact information needed sometimes takes longer time and this also causes information overload. Too much information is available but getting relevance text from these electronic documents regardless of millions of related documents makes automatic summarization a research to explore and proffer solutions to data mining and information retrieval issue.

Let’s pose a scenario of an individual with full records in different repositories but unknown to him, he absconded from his place of work. Management of the organization pulls out his data form different databases and similar information with additional ones are retrieved. There will be a need to sift such information and get summary of his account to the management. The staff in charge of the case the finds a way to summarize and gives the summary of the report to the management. The time and stress it would take to get the summary manually would have been used for making decision on the person in question. Automatic summarization helps in saving time of processing of information and also helps in digging into information one is less expected to use.

Summarization can be extractive or abstractive [9]. Extractive summaries are summaries in which all the text in the summaries are from the main document(s). Abstractive summaries are summaries produced in which some sentences are paraphrased to represent what the document is saying but the sentence(s) is not exactly as in the document(s). Summarization can also be single document or multiple documents. Single document summarization is summary generated from a document while multiple document summarization [10] is summary generated from two or more related documents. According to type of summary, different approaches are employed. For instance, an extractive summary will not need paraphrasing method or may not even need semantic method in summarization process [11]. To achieve a good summary, approaches such as topic identification, frequency of words, position of sentence, graph based, machine learning, semantic to mention a few are employed. With all these approaches [12], AS is still insufficient as compared to human summaries. Most of the summaries especially multi-document summaries face challenge of redundancy. Similar sentences are present in different documents and these sentences are sometimes rated high [9] due to popularity. Another issue why automatic summaries are still less efficient to manual summary is coherency. Sentences are rearranged for summarization and therefore loses its chronological arrangement. Automatic summarization systems are evaluated for relevancy, precision and call, length, expert evaluation, etc. With new ideas and improvement on existing approaches, summarization system issues.

The next section of the paper will explain related work of automatic summarization (AS) system including techniques employed, challenges in summarization automation will be discussed in section three. Current trends in automatic summarization is discussed in section four and section five concludes the paper.

2. RELATED WORK

The purpose of summary is to identify the sentences that best represent the document. Most summarization systems based their summary on sentences within the documents. Some of the methods used in summarization system include frequency based, graph based, position based, compression/ reduction, machine learning, semantic based, natural language processing etc. Some techniques are combined for better result. In this section, related work of each method focusing on recent research on the methods is explained.

2.1 Frequency Based Method

Early work on summarization system was by Luhn [8] based on word frequency. He used IBM processor to analyse scientific articles using the number of times a word occur and the significance of the word in the document. The significant factor of a sentence can be derived as:

$$s.f = \frac{\text{Number of significant words}^2}{\text{Total number of words}} \quad (1)$$

This reflects the number of occurrences of significant words within a sentence and the linear distance between them due to the intervention of insignificant word. All the sentences are ranked and the sentence(s) with highest significant are extracted as auto abstract. Luhn called the summary produced abstract but in the context, it is extract. It is a single document summary. Luhn's work gave birth to ideas to improve summarisation system and up till now, sentences are scored one way of the other in all summarisation system for tropical relevance. Research [13] has shown that binary weights have produce a good result in the world of summarization so that longer sentences will not be rated high especially when it is not a candidate for summary. [9] developed a query based approach summarisation system using sentence and section scorings. The sentences are scored based on Heading, location, term frequency and query methods while the sections are scored based on the measure of its importance using the sum of sentence scores in the section. The system is developed using GATE framework for text engineering as that underlying development environment and the system is applied to web search in Turkish language. Hierarchy identification experiment and task based evaluation were used to evaluate the system and it was found that the system has 20.3% improvement over Google and 23.6% improvement over unstructured summaries in terms of f-measure. Most of the previous summarisation approaches have ignored the structure of a document and have seen the document as a flat sequence of sentences but document structure may be especially helpful in determining the relevancy of a document during information retrieval.

Frequency based approach has been a method combined with other summarization methods for better result.

2.2 Graph-Based Approach

A graph is a pair G= (V, E) of sets satisfying the condition such that E is a subset of V^2. [14] The elements of V are nodes (or vertices) of a graph while the elements of E are its edges or lines. Edges are formed by drawing a line to link two vertices together. In summarization, documents are represented as graphs. Nodes in the graph are sentences in the document while the edges represent weight between the sentences. Different methods use different ways of representing their nodes. Some used only the frequency of words (after removing stop words) in the text to represent the sentences. Some used frequency of stemmed words [stemmer], root words, concepts, preferred word (as the case may be) to represent nodes in the graph. TF and TF*IDF weight of those words can also be used to represent words in the sentence. All sentences in the document must be represented in the graph and there must not be a node with value zero all through. This is needed to determine the edges between the sentences. Weights are popularly calculated using cosine similarity where each sentences are connected to one another to get the weight for each.

[15, 16] used page rank algorithm to rank sentences in order of relevance in the document. Page rank algorithm [17] is a Hidden Markov Model (HMM) which iteratively calculate the rank values web pages based on the in-links and out-links present on the page. The system is iterated until it converges. This implies that the pages with in-links are more popular than pages with few links. In applying this to documents, sentences with words that are popular (apart from stop words) are bound to have higher rank i.e. if there exist a relationship between two sentences such a similarity is a form of recommending the sentence, also, if a word or phrase appears in two or more sentences and a sentence U combines two or more of those words, then other sentences have shared in the words in sentence U which they indirectly gives sentence U a higher rank. Undirectional graph is used in which the number of edges is proportional to the number of vertices. Weight is attached to the graph by calculating the similarity between the sentences as

$$Similarity(S_i, S_j) = \frac{|\{w_k \mid w_k \in S_i \& w_k \in S_j\}|}{\log(|S_i|) + \log(|S_j|)} \quad (2)$$

The values of the weight then shows the connection between the sentences. The graph-based ranking algorithm is then iterated until convergence. To extract sentences, the weight associated with sentence pair gives the strength of connection between the pair. The result when evaluated with other systems, it was at top 5 out of 15 systems.

[1] proposed a method similar to [15] but they applied sentence clustering in calculating the centrality of sentence before applying HMM. The system select array of sentences and calculate the tf-idf score for each word in a sentence. A cosine threshold is defined and any word that falls within the threshold are added to centroid of the cluster. Score of each sentence will then be the sum of centroid values in the sentence. Idf-modified cosine calculation is achieved such that Matrix (i, j) = 1 if it is above threshold and matrix (i, j) = 0 if otherwise. The Lexical rank is calculated using iterative power method. The result of the research was ranked first at the DUC 2004 task evaluation.

[18] developed web page summarisation using pair-wise bipartite method. The study presented a temporary web page summarising the information needed base on query posed by the user. Here the author paired documents and performed bipartite graph on it using singular value decomposition. The result of each pair are paired again to rank sentences until final summary is generated. The system used URL links from google as data. In the first module URL links are compared to select suitable web pages for summarisation by calculating the text difference ratio of the page. Pre-processing stage of the system include tokenization of words, stemming and removal of stop words. Sentences are selected for summarisation by mapping each article as graph. The rows are sentences while columns represent stemmed words in the article. The frequency of a word in the sentence is entered as value of each cell in the graph. Combination theory that pairs the articles and determines the number of pairs that can be gotten given a particular number of pages to be selected. Similarity between sentences are calculated using cosine similarity measurement.

This is the weight of two articles. Sentences from each article are ranked in relation to article paired using U, S, V= SVD (weight). U ranked article A according to given index while V ranked article B. additional work of [Omodunbi] is the redundancy check function on sentences given the threshold of 0.8. A temporary webpage is produced with links showing where the sentences are extracted.

[19] proposed summarization framework for storytelling. The model used graph based clustering method for mutual effect between clusters, sentences and terms to rank sentences for summary. Documents are mapped into graph and each graph represent sentences in the document. Sentences are clustered into groups in accordance to the distance between two sentences. The distance is determined by calculating the cosine similarity between the sentences. Locally linear embedding formula was used to reduce the effect of unrelated sentences. This is done on set of sentences that are most similar to the sentence. High weight sentences are kept while the low ones are removed after embedding. This allows further processing to focus on promising candidates for summarization. New weight is calculated by getting the edge between a sentence and the cluster group it belongs. Sentences are ranked through mutual reinforcement algorithm to rank sentences based on rank of terms, sentences and clusters. System performance is improved compared to some summarizers.

[20] proposed automatically generated summary from full text articles to stand as step to extracting salient information for medical text indexer (MTI). The system w n the gap between Medical Subject Heading MESH semantically extract sentences from biomedical literature. Using graph based approach, concepts are generated from the medical dictionary UMLS by ascertaining the concept in the sentence graph. The documents are clustered based on the concepts in the graph using degree clustering. The clusters are grouped into sets depending on how strongly connected the concepts are. Sentences are selected by calculating the similarity between the sentences in the cluster using a non-democratic vote mechanism where votes are added to a cluster depending on how the sentence is related to the vote. Sentence with highest scores are selected for summary. Summaries generated here provides more information than abstracts and title and therefore improves automatic indexing approach and reduce the number of false notion given by MEDLINE citation.

Graph based has been an approach with good result and it is not limited to language. It can be applied to any form of language as long as it can be represented in a graph of vertices and edges.

2.3 Sentence Compression Approach

Sentence compression supports extractive multi-document summarization by reducing the length of summary candidates while preserving their relevant content, thus allowing space for the inclusion of additional material [21]. It was discovered that some words such as adjectives, adverbs, conjuncted verbs and so on can actually be removed from some sentences without losing the meaning of the sentences [22].

Extraction algorithms have a strong tendency to select long sentences from the text (since word frequency and distribution are often crucial, and are higher in long sentences even when sentence length is factored in). Standard summary length (250 words) by evaluating conferences [23, 24, 25] has allowed more application of sentence reduction method in summarization. Shortening the extracted sentences can be a way to further reduce the resulting summary, provided that the (essential) meaning of the sentence is preserved. Such summaries can presumably allow for shorter reading time [26].This method can actually be used as summarisation process and after the compression, there should be another method to re-arrange the sentences for coherency and relevancy to query posed.

[21] applied sentence compression of single-document to solve the problem of multi-document summarization. Topiary model, Hidden Markov Model HMM Hedge generator and Multi-candidate reduction MCR were used to extract the sentences. The sentences are first parsed and then trimmed before applying HMM hedge. The trimmer and HMM were applied to compress the sentences while the MCR was used to extract sentences for summary. The sentences are then parsed to be trimmed. The purpose of the trimmer is to omit determiners, auxiliary verbs and other adjuncts that their absence will not affect the remainder of the sentence.

Trimmer applies syntactic compression rules to a parsed tree are removal temporal expressions; preprocessed adjuncts; conjunctions; modal verbs; prepositional phrases that do not contain named entities etc.

Even with this algorithm, it is constrained to build a headline from a single sentence. However, it is often the case that no single sentence contains all the important information in a story. Relevant information can be spread over multiple sentences, linked by anaphora or ellipsis. When a candidate is chosen for summary, all other compressed variants of that sentence are eliminated.

[27] used syntactic approach to compressed sentences for summarisation. The parse trees method used was parts of speech tagger of Stanford parser data used are biomedical articles to implement the method. Cases considered are subtitles denoted with colons and dashes; determinants like the or a; serial prepositional phrases (PP) except the first; any PP embedded in three or more levels deep; Conjoined Noun Phrases (NP) except the first and NP with conjoined adjectives etc. are removed from the sentence. This method removes unnecessary words for short summary and allow more words to be present in the summary.

[22] assumed output of a summarizer and compress as many sentences the system can compress without deleting a single one. Dependent tree pruning method was used to remove some words from sentences thereby summarising documents with short sentences. Their first goal was to get a syntactic tree based on the grammatical importance, where for each node; a daughter node is an incident constituent which may be removed under certain conditions. X-bar theory was used which focused on placing adjuncts, complements and specifiers. Adjuncts are systematically removed but complements and specifiers applied other case by case rules to remove ones that are less relevant.

Apart from the syntactic function classification, the linguistic properties were used to preserve important part of the sentences (noun and clause heads) for coherency. These properties include lexical functions, fixed expressions, type of the article (definite or indefinite), parenthetical phrases, detached noun modifiers, the dependent constituent position in the sentence, negation and interrogation.

[28] summarized documents by compressing sentences using Integer Linear Programming (ILP). To compress the sentences, three different methods were employed namely: Compression of sentences before summarization, after summarization and combined.

Bi-gram language model was used to utilize predicate argument relations of a sentence and define constraints based on semantic roles to improve the weakness of lexical and syntactical constraints. Here, words in parenthesis are removed including personal pronouns and possessive words.

The bigram model first select the words that can be present in the sentence and then define indicator variable to make decision based on the sequence of words in the sentence. This combination allows words that should be pruned if they stand alone to remain in the sentence depending on the context that is used in the sentence. Predicates are also prevented from pruning through semantic role labelling. There is a constraint that says if a word is a predicate, it is included in the compression and if a predicate is in compression, then its argument is also kept in the compression. Compression before summarisation is very important as it allows guide in selecting sentences without considering adjuncts in the sentences.

The work of [29] focused on evaluating automatic summary based on syntactic pruning of sentences. Mead and Blogsum [1] summarizers were used to test the method. Their main purpose was to remove redundant and irrelevant information from sentences to allow more space for more relevant content. The sentences are parsed into Stanford parser [30] for analysis. As each sentence is parsed for pruning, three things are done on the sentence namely: Syntax-driven pruning, Syntax and relevancy based pruning and Relevance driven syntactic pruning.

Syntax-driven pruning used the syntactic simplification method of pruning where Relative clauses, adjective and adverbial phrases, conjunct clauses as well as specific types of PPs are pruned but PPs that modify verb phrases are pruned with caution as they may be part of the verb's frame and required to understand the verb phrase. Syntax and relevancy based pruning is employed to know the relevancy of the phrase according to query. To do this, cosine similarity measure was employed to get the tf-idf values between the phrase and the query and this phrase can only be removed if it's below certain threshold.

Relevance driven syntactic pruning focused more on preserving relevant information. The extracted sentences are parsed using Stanford parser while the cosine similarity is calculated to show the relevancy of the phrase or sentence to its dependant. Figure one shows the dissemination of sentence on Stanford parse tree

Sentence: *Turkey had been asking for three decades to join the European Union but its demand was turned away by the European Union in December 1997 that led to a deterioration of bilateral relations.*

The highlighted color are phrase candidates for reduction.

The system was evaluated using DUC and TAC documents on MEAD and Blogsum summarizers. It was discovered that the compression rate is high and this reduces the pruning by about 8 to 11%. But while evaluating the content of information generated using ROUGE, it was discovered that there is no much improvement to the summary previously generated by the summarisers.

From the papers reviewed on sentence reduction, it can be regarded as abstractive summary since the sentences present are not exactly what is in the document. For example, sentence in [29] review will be

Turkey had been asking to join the European Union but its demand was turned away by the European Union

Some phrases are missing within and after the new sentence and therefore cannot be called extractive.

Also, it is deduced that syntactic based compression does not improve a generic summarization system [27, 28, 29, 31] when evaluated by state-of-the-art standard but can achieve better performance if semantic role information can be incorporated into the model.

2.4 Biased Approach

This section explains different ways summarizers are developed without a particular approach. The commonest way is to measure the importance of sentence based on position of sentence. [32] worked on discovering the best locations for picking out abstract-worthy sentences. From the study, it was deduced that extracting important sentences depend on the genre of the text. (Scientific, news, business etc.) [28] said the first and last three sentences of the document are considered as good candidate good for summary. Cue words such as finally, conclusion, furthermore etc. makes a sentence relevant in a document. Similarity with topic title which matches the word in the query with words in the sentence are also considered as candidate for summary. The length of a sentence can also be considered.

.

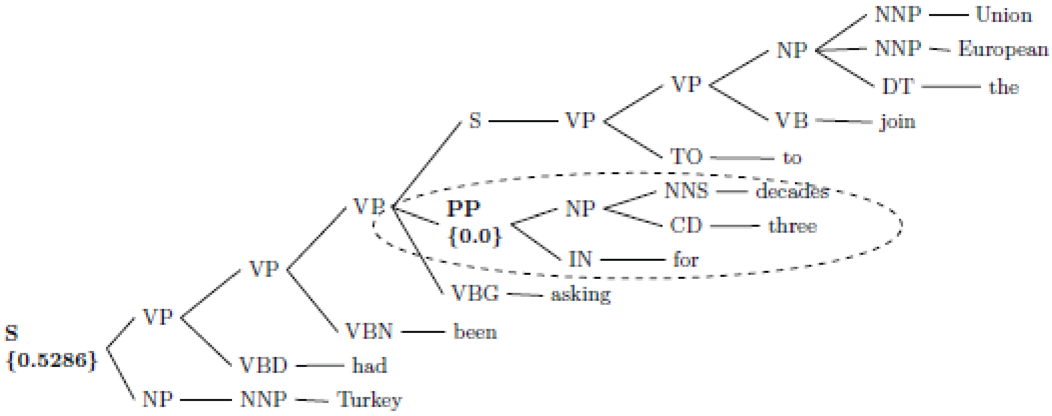

Fig 1: Dependency Phrase Structure of Sample Sentence [29]

From this methods good as combine two or more approaches to produce a good summary, other approaches include machine learning approach, semantic approach etc.

3. ISSUES IN AUTOMATIC SUMMARIZATION

Although TS has been in existence, its result compared to human summary is very low in terms of coherency. There is still need for human intervention to edit the extracted sentences to smoothen the language [33]. Since most of the summarizers are based on extraction, the systems do not have the ability to understand the chronological structure of the text.

Ambiguity of words is a huge issue that is yet to be resolved in TS. Words with different meanings (such as bank beside a river or where money is kept) may appear in the same text but this can be misinterpreted for other meanings. Also anaphors such as pronoun (e.g. he) that refers to an entity in previous sentence(s) is a great concern especially when the sentence is ranked higher the reference sentences. A more reliable solution would require linguistic analysis which is beyond the scope of a pure IR approach [34].

In a multi-document summarization, there is an evidence of duplication or redundancy in the summary. This is because information in document A will also appear in document B and this even make such information good candidate for summary. Only one of such sentence should remain in the summary. [18] used text difference to remove duplicated sentences by setting the threshold of the sentence to 0.8 but this has not perfectly removed redundancy because of its lack in semantic training.

When developing automatic summaries to mimic human summaries, a huge gap is yet to be filled. This is because most of the summarization systems are extract summaries and it is very rare that human summaries will have exactly the same sentences in the main text. The definition of summary means ability to understand and comprehend text and be able to give the short version of the story [35]. No summarizer has been able to attain this. Abstract summary is difficult because the system will have to construct sentence by itself and it must be able to construct correct English.

Another issue is the evaluation methods used in testing summarization system. Common evaluation methods of automatic text summarization are Human-generated Summary and ROUGE (Recall Oriented Understudy for Gisting Evaluation). ROUGE sets standard evaluation using human produced summary and compare the result with other results of the systems run on ROUGE. Precision, Recall F-score is calculated for two or more systems. Other parameters such as coherency and informativeness of the summary are not tested automatically.

Conferences on text analysis [36] and document understanding are set up to deliberate on summarization systems as the provide dataset for systems and evaluate the system based on manual summarization of the dataset. These conferences keep track of state-of-the-art in Information retrieval field and produce tasks for such field. For instance TAC of 2014 has incorporated summarization task to summarize biomedical literature. The reason for choosing biomedical literature is the rapid increase in biomedical articles. It was discovered that 1.5 articles are added to PubMed per minute. [23] summarizes the dataset manually and run it on ROUGE to compare the result with automatic systems registered for the task. Even with this, it is still not efficient to say the system has performed well.

4. APPLICATIONS OF AUTOMATIC SUMMARIZATION

AS are used and applied on daily basis. Due to this, researchers have engaged in developing AS that will fit at least the field it is targeting. Such fields include news, web pages, biomedical information, update text, email, user focused, scientific articles and so on. For this review, some of the fields that has been proposed by DUC [23, 24, 25] for evaluation are discussed.

4.1 Web Page Summaries

The growth of web is alarming. Everybody wants to be known worldwide. Information on the web increase per second. The Indexed Web contains at least 1.26 billion pages [37]. With this vast information, there is a need to find a way of retrieving information need from the billions of web pages. Searching information from a huge amount of webpages would be impossible without the help of search engine. AS provide a way to do this. For example, Google [17] search engine result is a form of summary. It brings the URL link of pages about the query posed by the user. [38] developed a temporal web summary where sentences are ranked not only according to its term frequency but also according to the content source of the webpage. Temporal summarizer extracted sentences from web pages base on the statistical parameter of text feature. [18] made use of URL link result from google to summarise web pages based on query posed. It is assumed that the first 10 links from search engine are the web pages needed by the user to retrieve information. Instead for the user to search each page, summary of those pages are given as a temporary webpage.

4.2 Update Summaries

These are summaries that are given to people who already have the knowledge of the information. It give the summary of current topic in question. The purpose is to generate a relevant summary for text documents at a particular time taking into consideration the users have read the earlier documents. This is used for news update, update of football match, update of an election, hospital patient's update, web page update, mail update, etc. this form of summary is dynamic and it has time attached to every event of the new. One of the DUC's 2008 task is update summary where documents are given with timeline. Updates are produced dynamically without producing previous information that has been produced in the previous summary.

[39] developed a system to update document using reinforcement process where previous sentences in the summary will be used as constraints to extract new summary from new sentences released. None of the old sentences will be present in the new summary. Quadratic programming is used to formulate summarization problem as quadratic problem so that it can be polynomial time solvable. [40] built an evolutionary manifold ranking model using iterative feedback mechanism to integrate update of information in a temporarily evolving data in order to produce summary based on query posed by the user. Spectral clustering is used to improve the coverage of summary content by the partition of sub-topics with less informed sentences and more informed one.

4.3 Biomedical Summarisation

Biomedical summarization involves application of summarization on biomedical information basically biomedical journals. Information regarding biomedical journals are so enormous that MEDLINE (biomedical bibliography text) has over 22 million articles and MESH manually index the abstract and keywords of these articles. To write abstract for average of 3000 articles per day is a great task. To retrieve desired literature on a particular topic in free text, summarization is applied as a solution to information overload. [20, 41, 42] proposed summarisation of such articles to serve as automatic indexing of summaries of such journal. They used semantic graph based approach to generate summaries of literature. Instead of using terms in the sentence, the systems run the text on UMLS to generate the concepts in the sentences. These concepts replace terms in the sentences as a step to summarisation methods. Graphed based methods combine with frequency and clustering of concepts are used to rank the sentences. TAC 2014 [36] has added evaluation of biomedical summaries to its task where Reference Paper (RP) is presented with the papers cited in the RP. AS system should be able to see the citance, pick the sentences that talks about the citance in RF (not more than 5 sentences before or after the cite) and check the journal cited to know the exact paragraph or sentence that explained what RP has cited. This is an interesting task and we hope that AS will be able to perform this expectation.

Another biomedical summarization is the application of AS to clinical documents. Clinical documents are represented in Electronic Health Records (EHR) in a standardized format or codes to be able to effectively retrieve information stored in the database. As a result, patient information may not be completely represented in the database. Also exchange of information between EHR of different standard may be difficult. [43]. Automatic summaries can be used to retrieve information from EHR and complete information can be stored there if summarization can be used to retrieve the information. [44] used knowledge based and extractive text summarization method to generate ICD codes from patient’s EMR by computing the c-value of noun phrases extracted by TS. The noun phrases are run on name Entity Recognition system MetaMap for recognition of biomedical concepts in EMRs and mapped them to ICD codes using UMLS Meta thesaurus. [45] used natural language rule to automatically generate summary from patient’s history in the database. The system is structured in a way that information from database from each module determines the content of sentences in each paragraph as human medical summary does. Although the authors did not consider the length of summary nor the format (sentence, list or phrases), summarization in medical record will bridge the gap of IR problem.

5. CONCLUSION

In this paper, we have attempted to give the trends in Automatic summarization methods with the issues facing the study. Current applications of AS were explained among others. As people cannot do without summaries everyday, AS is improved and the research is applied in different field. It should be noted that AS is important to tackle the problem of information overload and other problems of retrieving information especially field with large amount of information like world wide web, hospital information and biomedical articles. With different AS available, their evaluations compared to human summaries are still not good. Most AS are extractive while human summaries are mostly abstractive. Abstractive AS will require NLP to construct sentences and this is one of the issues in AS that will be tackled in the future work. With large amount of data and confidentiality issues in hospital information system, automatic summarization system can be used as solution to interoperability and IR in health care domain.

6. ACKNOWLEDGMENTS

Our thanks to the University of Kentucky that provided us with current journals to enrich the work.

7. REFERENCES

[1] Erkan, G., and Radev, D. R. “LexRank: Graph-based lexical centrality as salience in text summarization”, Journal of Artificial Intelligent Research. (JAIR), .22(1), (2004), 457-479.

[2] Luhn, H.P., “The Automatic Creation of Literature Abstracts”, IBM Journal of Research and Development, 2(2), (1958), 159-165.

[3] Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee, G., Pattersom, D.A., Rabkin, A., Stoica, I., and Zaharia, M. 2009 Above the Clouds: A Berkeley View of Cloud Computing.

[4] Malathy, G., Somasundaram, Rm and Duraiswamy K., “Performance Improvement in Cloud Computing Using Resource Clustering”, Journal of Computer Science 9 (6) (2013). 671-677, ISSN: 1549-3636

[5] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., and Brandic, I. “Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility”. Future Generation computer systems, *25*(6), (2009), 599-616.

[6] Watson, H.J., Goodhue, D.L. and Wixom, B.H., “The benefits of data warehousing: Why some organizations realize exceptional payoffs”. Information and Management, 39(6), (2002), .491-502.

[7] Awoyelu, I., Omodunbi, T. and Udo, J. “Bridging the Gap in Modern Computing Infrastructures: Issues and Challenges of Data Warehousing and Cloud Computing”. Computer and Information Science, 7(1) (2013), 33.

[8] Krishnan, K., 2013. *Data Warehousing in the Age of Big Data*, Elsevier.

[9] Pembe, F., and Gungor, T. 2008. Towards a new summarization approach for search engine results: An application for Turkish. Proceedings of the 23rd International Symposium on Computer and Information Sciences, Istanbul, pp.1-6

[10] Hahn, U. & Mani, I., 2000. The challenges of automatic summarization. Computer, 33(11).

[11] Goldstein, J., Mittal, V., Carbonell, J., and Kantrowitz, M. (2000, April). Multi-document summarization by sentence extraction. In Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization-Volume 4 (pp. 40-48). Association for Computational Linguistics.

[12] Barzilay, R., and McKeown, K.R. Sentence fusion for multi-document news summarization. Computational Linguistics, 31, (2005), 297–328.

[13] Nenkova, A. and McKeown, K., A Survey of Text Summarization Techniques. In C. C. Aggarwal & C. Zhai, eds. Mining Text Data. Springer US, (2012) 43-76.

[14] West, D. B. 2001. Introduction to graph theory (Vol. 2). Upper Saddle River: Prentice hall.

[15] Mihalcea, R. 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In Proceedings of the ACL 2004 on Interactive poster

and demonstration sessions (p. 20). Association for Computational Linguistics.

[16] Mihalcea, R. and Tarau, P., 2004. TextRank: Bringing order into texts. .Proceedings of EMNLP, 4(4), .404–411.

[17] Brin, S. and Page, L. 1998. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 30(1–7).

[18] Theresa Omodunbi and Abimbola Soriyan (2012): Multi-Web page Summarization and Presentation using Pair-wise Bipartite Graph. Proceedings of 4th Annual International Conference on ICT for Africa, Kampala, March 21st-24th, pp. 241 - 242, Uganda.

[19] Zhang, Z., Ge, S. S., and He, H. (2012). Mutual-reinforcement document summarization using embedded graph based sentence clustering for storytelling. Information Processing & Management, 48(4), 767-778.

[20] Jimeno-Yepes, A. J., Plaza, L., Mork, J. G., Aronson, A. R., and Díaz, A. MeSH indexing based on automatically generated summaries. BMC bioinformatics, 14(1), (2013), 208

[21] Zajic, D., Dorr, B.J., Lin, J. and Schwartz, R. Multi-candidate reduction: Sentence compression as a tool for document summarization tasks .Information Processing and Management 43, (2007), 1549-1570.

[22] Yousfi-Monod, M. and Prince, V. 2008. Sentence Compression as a Step in Summarization or an Alternative Path in Text Shortening. CoLing: Companion volume: Posters 139-142.

[23] Dang, H. and Owczarzak, K. 2008. Overview of the TAC 2008 Update Summarization Task. In: Proceedings of the Text Analysis Conference, TAC 2008, Gaithersburg

[24] Dang, H.T.: Overview of DUC 2006. In: Proceedings of the HLT-NAACL 2006 Document Understanding Workshop.

[25] Salakoski, T., Ginter, F., Pyysalo, S. and Pahikkala T. (Eds.): Advances in Natural Language Processing, 5th International Conference on NLP, FinTAL 2006, Turku, Finland, August 23-25, 2006.

[26] Gagnon, M. and Da Sylva, L., 2006. Text compression by syntactic pruning. In ADVANCES IN ARTIFICIAL INTELLIGENCE, PROCEEDINGS. pp. 312-323.

[27] Lin, J., & Wilbur, W. J. Syntactic sentence compression in the biomedical domain: facilitating access to related articles. Information Retrieval, 10(4-5), (2007), 393-414.

[28] Chali, Y. & Sadid, H. 2012. On the Effectiveness of Using Sentence Compression Models for Query-Focused Multi-Document Summarization. Proceedings of COLING 2012, (December 2012), p.457-474.

[29] Perera, P. and Kosseim, L. 2013. Evaluating Syntactic Sentence Compression for Text Summarisation. In Natural Language Processing and Information Systems (pp. 126-139). Springer Berlin Heidelberg.

[30] Marneffe, M.C.D. and Manning, C.D. 2008. The Stanford typed dependencies representation. In: Proceedings of the Workshop on Cross-Framework and Cross-Domain Parser Evaluation, CrossParser 2008, Manchester, pp. 1–8

[31] Jing, H., 2000. Sentence reduction for automatic text summarization. *ANLP*, p.310-315.

[32] Lin, C. Y., and Hovy, E. 2000. The automated acquisition of topic signatures for text summarization. In Proceedings of the 18th conference on Computational linguistics-Volume 1 (pp. 495-501). Association for Computational Linguistics.

[33] Verma, R., Chen, P., and Lu, W. 2007. A Semantic Free-Text Summarization Systems Using Ontology Knowledge, Document Understanding Conference DUC 2007, pp. 1-5.

[34] Schuemie, M.J., Kors, J.A. and Mons, B. 'Word Sense Disambiguation in the Biomedical domain: An Overview', Journal of Computational Biology, Vol. 12, No. 5, (2005), 554-565.

[35] Lyman, R. L. "Summary of Investigations Relating to Grammar, Language, and Composition (January, 1929, to January, 1931). II'. The Elementary School Journal, (1932), 352-363.

[36] Dang, (2014) CFP: NIST Biomedical Summarization shared task http://www.nist.gov/tac/2014/BiomedSumm/ Accessed 02/05/2014

[37] Daily Estimated size of World Wide Web. Retrieved from http://www.worldwidewebsize.com/ Accessed 14 June, 2014

[38] Jatowt, A., and Ishizuka, M. 2004. Temporal web page summarization. In *Web Information Systems–WISE 2004* (pp. 303-312). Springer Berlin

[39] Li, X., Du, L., & Shen, Y. D. (2013). Update summarization via graph-based sentence ranking. *Knowledge and Data Engineering, IEEE*

[40] He, R., Qin, B., and Liu, T. "A novel approach to update summarization using evolutionary manifold-ranking and spectral clustering. "Expert Systems with Applications 39.3 (2012): 2375-2384

[41] Plaza, L., Díaz, A., and Gervás, P. A semantic graph-based approach to biomedical summarisation. *Artificial intelligence in medicine*, *53*(1), (2011), 1-14.

[42] Yoo, I., Hu, X., and Song, I. Y. A coherent graph-based semantic clustering and summarization approach for biomedical literature and a new summarization evaluation method. *BMC bioinformatics*, *8*(Suppl 9), (2007), S4.

[43] Iroju, O., Soriyan A., Gambo I., and Olaleke J. "Interoperability in Healthcare: Benefits, Challenges and Resolutions." *International Journal of Innovation and Applied Studies* 3.1 (2013), 262-270.

[44] Kavuluru R., Han S., and Harris D. (2013). Unsupervised Extraction of Diagnosis Codes from EMRs Using Knowledge-Based and Extractive Text Summarization Techniques, NDLB 2013

[45] Scott, D., Hallett, C., & Fettiplace, R. Data-to-text summarisation of patient records: Using computer-generated summaries to access patient histories. *Patient education and counseling*, *92*(2), (2013), 153-159.