2017 International Conference on Big Data, IoT and Data Science (BID)
Vishwakarma Institute of Technology, Pune, Dec 20-22, 2017

Automatic Text Summarization of News Articles

Prakhar Sethi[1], Sameer Sonawane[2], Saumitra Khanwalker[3], R. B. Keskar[4]
Department of Computer Science Engineering, Visvesvaraya National Institute of Technology, India
[1]prakhar.sethi2@gmail.com, [2]sameer9311@gmail.com, [3]theapogee2011@gmail.com, [4]rbkeskar@cse.vnit.ac.in

***Abstract* - Text Summarization has always been an area of active interest in the academia. In recent times, even though several techniques have being developed for automatic text summarization, efficiency is still a concern. Given the increase in size and number of documents available online, an efficient automatic news summarizer is the need of the hour.**

In this paper, we propose a technique of text summarization which focuses on the problem of identifying the most important portions of the text and producing coherent summaries. In our methodology, we donot require full semantic interpretation of the text, instead we create a summary using a model of topic progression in the text derived from lexical chains. We present an optimized and efficient algorithm to generate text summary using lexical chains and using the WordNet thesaurus. Further, we also overcome the limitations of the lexical chain approach to generate a good summary by implementing pronoun resolution and by suggesting new scoring techniques to leverage the structure of news articles.

***Keywords* – Extractive Text Summarization, Lexical Chains, News Summarization, Natural Language Processing, Anaphora Resolution**

I. INTRODUCTION

With the availability of World Wide Web in every corner of the world these days, the amount of information on the internet is growing at an exponential rate. However, given the hectic schedule of people and the immense amount of information available, there is increase in need for information abstraction or summarization. Text summarization presents the user a shorter version of text with only vital information and thus helps him to understand the text in shorter amount of time.The goal of automatic text summarization is to condense the documents or reports into a shorter version and preserve important contents [1].

A. Summarization Definition

Natural Language Processing community has been investigating the domain of summarization for nearly the last half century.Radev et al, 2002 [3] defines summary as "*text that is produced from one or more texts, that conveys important information in the original text(s), and that is no longer than half of the original text(s) and usually significantly less than that.*" Three main aspects of research on automatic summarization are delineated by this definition:

- Summaries may be produced from a single document or multiple documents,
- Summaries should preserve important information,
- Summaries should be short.

B. Need for Automatic Summarization

The main advantage of summarization lies in the fact that it reduces user's time in searching the important details in the document. When humans summarize an article, they first read and understand the article or document and then capture the important points. They then use these important points to generate their own sentences to communicate the gist of the article. Even though the quality of summary generated might be excellent, manual summarization is a time consuming process. Hence, the need for automatic summarizers is quite apparent.

The most important task in extractive text summarization is choosing the important sentences that would appear in the summary. Identifying such sentences is a truly challenging task. Currently, automatic text summarization has applications in several areas such as news articles, emails, research papers and online search engines to receive summary of results found[2].

II. PREVIOUS WORK

The earlier approaches in text summarization focused on deriving text from lexical chains generated during the topic progression of the article. These approaches were preferred since it did not require full semantic interpretation of the article. The approaches also merged several robust knowledge sources like a part-of-speech tagger, shallow parser for the identification of nominal groups, a segmentation algorithm and the WordNet thesaurus.

According to Wikipedia, WordNet is a lexical database for the English language. It groups English words into sets of synonyms called synsets, provides short definitions and usage examples, and records a number of relations among these synonym sets or their members. For example two senses of *"bike"* are represented as: motorcycle, bike and bicycle, bike, wheel, cycle. Words of the same category are linked through semantic relations like synonymy, which is the study of words with the same or similar meaning, or the quality of being similar, and hyponymy, which relates to words of more specific meaning than a general or superordinate term applicable to it. A commonly used example to demonstrate hyponmy is: *daffodil, which is a hyponym of flower*.

A. Lexical Chains

Morris, Jane and Hirst[6] first introduced the concept of lexical chains. In any given article, the linkage among related words can be utilized to generate lexical chains. A lexical chain is a logical group of semantically related words which depict an idea in the document. The relation between the words can be in terms of synonyms, identities and hypernyms/hyponyms. For example, we can put words together when:

- Two noun instances are identical, and are used in the same sense. (*The cat in the kitchen is large. The cat likes milk.*)
- Two noun instances need not be identical but are used in the same sense (i.e., are synonyms). (*The bike is red. My motorcycle is blue.*)
- The senses of two noun instances have a hypernym/hyponym relation between them. A hypernym is a word with a broad meaning constituting a category into which words with more specific meanings fall. (*Daniel gave me a flower. It is a Rose.*)
- The senses of two noun instances are siblings in the hypernym/hyponym tree. (*I like the fragrance of Rose. However Sunflower is much better.*)

These relations can be used to group noun instances in a lexical chain given the condition that each noun is assigned to only one chain. The challenging task here is determining the chain to which a particular noun will be assigned since it may have multiple senses or contexts. Also, even though there is a single context for the noun usage, it might be still ambiguous to determine the lexical chain. The reason being, for example, one lexical chain might correspond to hypernym relation of the noun while the other might correspond to its synonym relation. Hence, to be able to resolve such ambiguities, the nouns must be grouped in such a way that it creates longest or strongest lexical chains. If a chain contains several nouns relating to same meaning, then we call that chain as longest chain. Similarly, the lexical chain with highest score will be termed as strongest chain.

Generally, a procedure for constructing lexical chains follows three steps:

1) Select a set of eligible words such as nouns, adjectives, adverbs, etc. In our case, we choose only nouns;
2) For each eligible word, search for an corresponding chain depending on a relatedness criterion among members of the chains;
3) If it is found, insert the word in the chain and update it accordingly.

A similar path was followed by Hirst and St-Onge (H&S) in their approach of summarization. In first step, all words in the document tagged as nouns in WordNet are picked up. In the next step, their relatedness is measured based on the distance between their occurrences and their connection in the WordNet thesaurus. Three kinds of relation are defined extra- strong (between a word and its repetition), strong (between two words connected by a Wordnet relation) and medium- strong when the link between the synsets of the words is longer than one (only paths satisfying certain restrictions are accepted as valid connections).

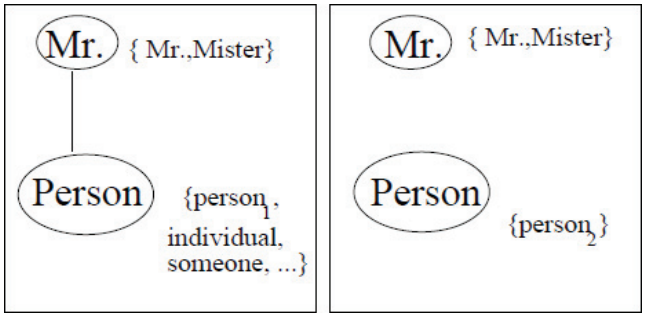
B. Barzilay and Elhadad Approach

Barzilay and Elhadad[7] proposed lexical chains as an intermediate step in the text summarization process. They proposed to develop a chaining model according to all possible alternatives of word senses and then choose the best one among them.

Their approach can be illustrated using the following example -

Mr. Kenny is the person that invented an anesthetic machine which uses micro-computers to control the rate at which an anesthetic is pumped into the blood. Such machines are nothing new. But his device uses two micro-computers to achieve much closer monitoring of the pump leading the anesthetic into the patient.

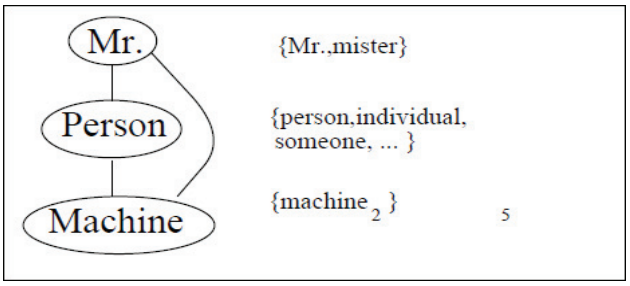
First, a node for the word "*Mr*" is created [lex "Mr.", sense mister, Mr.]. The next candidate word is "*person*" It has two senses: "*human being*" (person-1) and "*grammatical category of pronouns and verb forms*" (person – 2) The choice of sense for "person" splits the chain world to two different interpretations as shown in Figure 1.



(a) Figure 1

They define a component as a list of interpretations that are exclusive of each other. Component words influence each other in the selection of their respective senses. The next candidate word "*anesthetic*" is not related to any word in the first component, so they create a new component for it with a single interpretation.

The word "*machine*" has 5 senses - machine(1) to machine(5). In its first sense, "an efficient person", it is related to the senses "*person*" and "*Mr*". It therefore influences the selection of their senses, thus "machine" has to be in the first component. After its insertion the picture of the first component becomes the one shown in Figure 2.



(b) Figure 2

Under the assumption that the text is cohesive, they define the best interpretation as the one with the most connections (edges in the graph). They define the score of an interpretation as the sum of its chain scores. A chain score is determined by the number and weight of the

relations between chain members. Experimentally, they fixed the weight of reiteration and synonym to 10, of antonym to 7, and of hypernym and holonym to 4. Their algorithm computes all possible interpretations, maintaining each one without self contradiction. When the number of possible interpretations is larger than a certain threshold, they prune the weak interpretations i.e. interpretations having low scores according to this criteria, this is to prevent exponential growth of memory usage. In the end, they select from each component the strongest interpretation.

C. Silber and McCoy Approach

Summarization has been viewed as a two-step process. The first step is the extraction of important concepts from the source text by building an intermediate representation of some sort. The second step uses this intermediate representation to generate a summary.

In the research presented by Silber and McCoy[8], they use a different method to extract important concepts from source and followed Barzilay and Elhadad[7] in employing lexical chains to extract important concepts from a document. They presented a linear-time algorithm for lexical chain computation and offered an evaluation that indicates that such chains are a promising avenue of study as an intermediate representation in the summarization process.

In order to compute lexical chains in linear time, instead of computing every interpretation of a source document as Barzilay and Elhadad[7] did, they create a structure that implicitly stores every interpretation without actually creating them, thus keeping both the space and time usage of the program linear. They then provide a method for finding that interpretation which is best from within this representation.

D. Issues with Silber and McCoy Algorithm

Proper nouns (people, organization, company, etc.) are often used in naturally occurring text, but since we have no information about them, we can only perform frequency counts on them. Anaphora resolution i.e. resolving pronouns in the text, especially in certain domains, is a bigger issue. Silber and McCoy Algorithm fails to address these issues. Much better results are anticipated with the addition of anaphora resolution and proper noun resolution to the system.

Our survey into the work done in the field of summarization discussed above helped us understand the issues mentioned and the challenges in the field. We implemented the basic lexical chain model as discussed by Silber and McCoy[8] and then added our enhancements to resolve the issue of anaphora resolution and time complexity of lexical chain generation

III. Our Approach

The lexical chain generation algorithm proposed by Barzilay and Elhadad[7] described in previous section has exponential run time which was improved by Silber and McCoy algorithm[8] and has linear run time complexity. Hence we adopted Silber and McCoy algorithm[8] to construct the basic lexical chain model. Further, we have also tried to resolve the issues in both algorithms by implementing pronoun resolution and enhanced sentence scoring to leverage the structure of news articles.

The following steps describe our algorithm for text summarization.

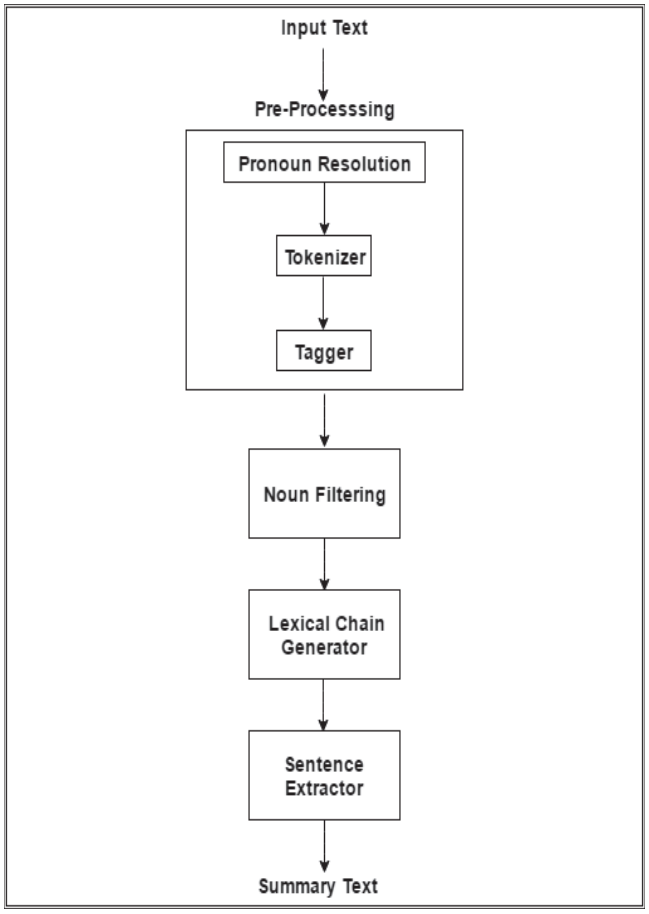
1) After receiving the input, we first perform pronoun resolution on the text.
2) In pronoun resolution, we try to find the best representative noun for a pronoun. After finding them, we replace them in our sentences.
3) To replace the pronouns, we first tokenize the passage into individual sentences.
4) Each of these sentences is further tokenized into words, and after obtaining the word list, we replace the pronoun with the representative noun, and re-construct the sentence.
5) We then find the Part-of-Speech tag for every word to separate the nouns.
6) These nouns are then used for lexical chain construction.
7) Every lexical chain consists of closely related nouns based on their synonymy.
8) We then score the lexical chains based on our scoring criteria, and pick the strong chains whose score is greater than the decided threshold.
9) Using the strong lexical chains, we can then score the individual sentences, and choose those sentences to be in our summary whose score is greater than the decided threshold.
10) We also score the proper nouns in the passage based on their frequency in the passage.
11) We select a subset of these proper nouns whose score is greater than the decided threshold. Subsequently, we pick the sentences which contain the first occurrence of these proper nouns and add them to our summary.
12) Finally, the sentences are ordered according to their occurrence in the passage, and the obtained sets of sentences represent the summary of the news article.

A. Sentence Tokenization

We take the article as input in our system and tokenize it into sentences. We perform this tokenization on the basis of punctuation marks valid for locating sentence termination points as identified by the rules of English Grammar. To tokenize it into valid sentences, we use the NLTK library, which is based on the specified language of the text, which in our case is English.

B. Part of speech tagging for tokenized words

After tokenizing the article into sentences, we drive our focus on every sentence in the article to extract important features related to the article. We further tokenize a sentence further into words. For each word, we identify which POS (Part of Speech) tag it relates to. The Part of Speech tag helps to identify the relation of the word to one of the broad



(c) Flow Diagram of the Algorithm

categories of words defined in the English language, such as Nouns, Pronouns, Verbs, etc. Its significance in our scenario will be justified later in the report. We first tokenize the given sentence to a list of words in the sentence. For Part of Speech tagging, we again refer to the NLTK library. The NLTK library maintains a large corpus of English words, which identifies the word and also stores the Part of Speech tag it relates to. Thus, we generate a new list of items, where each item is a tuple consisting of the word in our sentence, along with its Part of Speech tag.

C. Pronoun Resolution

English passages use a lot of pronouns to continuously refer some nouns in an article, to replace their over usage. Thus, if we want to identify important nouns in a passage, we should resolve every pronoun in it to relate to their respective noun occurrence. The problem of pronoun resolution has been identified as a very hard problem because it requires a syntactic as well as semantic understanding of the passage. There exists numerous algorithms for the same, some solely on the syntactical features, and others which use machine learning techniques to train their system to identify and understand the semantic relations in the text. For our problem, we refer to an existing solution implemented by the Stanford NLP Group, Stanford CoreNLP, which is a suite providing numerous natural language analytics tools, including pronoun resolution. We have implemented their local library as a local server on a machine, and perform API calls to it. Thus, we perform an API call to its local server from our program, passing our passage along with the necessary options to perform pronoun resolution and we receive an output describing the relations of various pronouns and the noun it would be referring to. With this information, we replace the pronouns in the passage with the referenced noun.

D. Lexical Chain formation

We had identified every word's part of speech tag and also resolved the pronoun occurrences with the respective nouns. Our next step towards summarization is to identify the main concept the passage focuses on. We try to find the main concept on the basis of the nouns in the passage. The intuition behind this is since we are referring to news articles, they contain a lot of nouns and generally direct their focus on a particular set of nouns, whether the news article belongs to the category of World News, Political News, Sports News, Technology News, etc. Thus, if we are able to identify a set of nouns which form the core of the news article, extracting sentences more focused on them generates a concise and relevant summary. To identify the important nouns in the passage, we implement the technique of lexical chain formation, presented by Morris, Jane and Hirst[6], and implemented for text summarization by Barzilay and Elhadad[7]. Using lexical chains, we try to group together similar nouns into chains and then identify strong chains on the basis of a scoring criteria. After identifying the strong chains, various extraction techniques can be used to extract a subset of sentences from the news article. In our implementation of lexical chain formation, we first try to find all the possible meanings or senses a noun is used. This was achieved by using WordNet. WordNet is a large lexical database of English. WordNet superficially resembles a thesaurus, in that it groups words together based on their meanings. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. To generate a lexical chain in our case, we use the data structure Dictionary, where each found meaning will represent a list of those nouns in the article having this as one of their meanings. In this way, we are able to capture every noun and their possible senses in our lexical chain structure. Using this structure, we find the important noun sets which will be used for sentence extraction based on our scoring and sentence extraction techniques.

E. Scoring Mechanisms

The steps discussed till now are used for extracting important information from the news article, which will help us to generate a summary from the article. In this step, we discuss the various aspects of the text which will be scored, and will be used for summary extraction.

1) **Lexical Chain Scoring:** We have formed the lexical chains using all the nouns in the article, except for proper nouns. The problem with proper nouns is that they generally dont mean anything. Thus, they cannot be added to any lexical chain. One can try to assign genders to proper nouns and then find a chain accordingly. One method to find the gender of proper nouns could be to train a system through real world examples so that it returns the gender that it finds the most probable. However, this method also would not ensure high success rate, as multiple lexical chains could have the same gender elements. We had to now figure out a heuristic function which would help us score the lexical chains which were formed above. For scoring these chains, there can be multiple heuristics possible. The heuristic we implemented makes use of the important criteria identified by Barzilay and Elhadad[7], i.e. chain length, distribution in the text and the text span covered by the text. The following parameters are good predictors of the strength of a chain:

   Length: The number of occurrences of the members of the chain.
   Homogeneity Index:

$$\frac{Length - Number\ of\ distinct\ occurences}{Length} \tag{1}$$

   We evaluated the score of a chain as:

$$Score(Chain) = Length * Homogeneity \tag{2}$$

   On the basis of these scores, we later describe the criteria on which strong chains are identified.

2) **Sentence Scoring:** Using the scores computed for lexical chains, now we wish to find the scores for the sentences. After our sentences are scored, we can extract a particular subset of the sentences which would have a score above a decided threshold, and would form a part of the summary.
   In our approach for scoring the sentences, we choose the previously computed scores for lexical chains. After scoring the chains, we have to identify a set of strong chains, which would help us score the sentences. To identify the strong chains, we use the following criteria to rank the chains.

$$Score(Chain) > Average(Scores) + 2 * Standard\ Deviation(Scores) \tag{3}$$

   All the chains whose score satisfy these criteria are identified as strong chains. To score the sentences, our implemented heuristic makes use of the following criteria, noun frequency and sentence length. The noun frequency refers to the count of the nouns in a sentence that occur in strong chains. The sentence length was kept into consideration to avoid unfair scoring. If only the noun frequency is taken into consideration, long sentences would naturally have a higher probability of having large number of nouns. Thus, sentence length can be used to normalize the score.
   Thus, the formula for sentence scoring implemented is

$$Score(Sentence) = \frac{Count\ (Nouns\ in\ strong\ chains)}{Sentence\ length} \tag{4}$$

   On the basis of these scores, we later describe the criteria on which sentences are extracted.

3) **Proper Noun Scoring:** News articles contain a large number of proper nouns. These proper nouns cannot be added to our lexical chain structure, as there is no acceptable approach to identify the usage sense of these nouns. But, proper nouns are an integral part of news articles. Thus, their occurrence cannot be completely ignored. Our basis for scoring proper nouns in an article is its frequency in the article. We could not find any other appreciable characteristic for them. The major reason we could argue for using only the frequency was due to non-existence of features they could have relating to the language. Proper nouns are independent of the language of the article and thus, language specific criteria doesn't exist.
   Our scoring formula for proper nouns is

$$Score(Proper\ Noun) = Frequency(Proper\ Noun) \tag{5}$$

   On the basis of these scores, we later describe our criteria for sentence extraction based on these scores.

F. Summary Extraction

After scoring various aspects of our news article, we now present various methods which we implemented together to generate a relevant summary.

1) **Extraction based on our Article Category:** The articles that our summarizer focuses on are news articles. We tried to identify some relevant feature specific to a news article, which we added to our summarizer for summary generation. News articles are well-structured and organized. The writers tend to maintain a proper flow of information in them. The first couple of sentences usually contain most of the relevant information on what the article would later elaborate and discuss on. Thus, these sentences should have a higher importance over the later sentences in the article.
   We then parsed various news articles from websites like BBC, Times of India, the Hindu, etc to try to identify how many sentences should be given the most relevance. After evaluating some articles from each of these websites, we came to the conclusion that only the first sentence should be given a higher priority over other sentences. The first sentence in

these articles is generally long and covers the main gist of the news article.
Thus, for our summary generation, the inclusion of this sentence is necessary. We added this feature in our summarizer to necessarily extract this sentence and add it to our summary.

2) **Using Sentence Scoring:** In the earlier section, we described our scoring technique where we constructed a lexical chain data structure for similar nouns and used it to score our sentences. Our threshold for selecting a sentence is if its score is greater than the average of the scores of all the sentences. These sentences have a higher concentration of important nouns compared to other sentences, and thus should be part of the summary.
Thus, using sentence scoring, only those sentences would be part of the summary where

$$Score(Sentence) > Average(Sentence\ Scores) \quad (6)$$

3) **Using strong Lexical Chains:** We described our method of sentence extraction where we used the sentence scores to extract relevant sentences. The lexical chains played a major role here as they were used to score the sentences.
To increase the importance of the strong lexical chains, we applied another heuristic here described by Barzilay and Elhadad[7] in his text summarization techniques.
For each chain in the summary representation, choose the first sentence that contains the first appearance of a representative chain member in the text.
We implemented this technique for text extraction on our strong chains. The main reason behind this is that if sentence extraction is based on every chain, it would result in large summary size as well as increase the probability of adding irrelevant sentences to the summary because of the low scored lexical chains. Thus, for sentence extraction, we find the first sentence which contains one of the chain members for every strong chain.

4) **Using Proper Noun Scoring:** Since we are summarizing news articles, some extraction on the basis of proper nouns is necessary. We earlier described our scoring technique for proper nouns. Using these scores, we tried to explore a proper heuristic which would help generate relevant and concise summary.
After testing for multiple heuristics, the final accepted heuristic is to extract the first sentence for all those proper nouns whose score is greater than one-third of the number of sentences in the article.

$$Score(Proper\ Noun) > \frac{1}{3} * Count(Sentences) \quad (7)$$

The main idea behind keeping the heuristic to compare the number of sentences to the proper noun score, i.e. the frequency of the proper noun in the text was that, if a proper noun occurs large number of times in a news article, it has to be relevant to the subject of the article. If we had compared the score of a proper noun with the average of the scores of proper nouns, it would have chosen proper nouns relative to other proper nouns in the news article. But, our main aim here is to find some proper noun which dominates the news article in itself. Thus, we used the count of the sentences. After testing for multiple values, one-third was found to generate the most acceptable summary.
Thus, after choosing the important proper nouns, we extract the first sentence that occurs in the article.
All of these techniques would extract a subset of the sentences from the article. Our final summary would comprise of union of all the unique sentences extracted by each of the above described techniques.

The summary generated by our approach consists of the important sentences identified by lexical chains as well as the sentences containing vital information about the subjects (proper noun occurrences) in the article. We have mentioned the results of our experiments on one such news article in the next section.

IV. Experimental Results

We tested our algorithm at all stages on various inputs of various lengths to understand its strength and weaknesses. Following is one such article we tested our algorithm on along with the generated summary.

Article

Islamic State (ISIS) hackers have published a ”hit list” of over 70 US military personnel who have been involved in drone strikes against terror targets in Syria and asked their followers to ”kill them wherever they are”. According to ’The Sunday Times’, the hackers have links with Britain and call themselves ’Islamic State Hacking Division’ and circulated online the names, home addresses and photographs of more than 70 US staff, including women and urged supporters: ”Kill them wherever they are, knock on their doors and behead them, stab them, shoot them in the face or bomb them.” The group also claimed that it might have a mole in the UK’s ministry of defense and threatened to publish ”secret intelligence” in the future that could identify Britain’s Royal Air Force (RAF) drone operators. The new hit list features the ISIS flag above the heading: ’Target – United States Military’ and the document, circulated via Twitter and posted on the JustPaste website, states: ”You crusaders that can only attack the soldiers of the Islamic State with joysticks and consoles, die in your rage! ”Your military has no courage, neither has your president as he still refuses to send troops. So instead you press buttons thousands of miles away in your feeble attempt to fight us.

"A nation of cowards that holds no bravery as you resort to sending your remote-controlled unmanned Reaper and Predator drones to attack us from the skies. So this is for you, America. "These 75 crusaders are posted as targets for our brothers and sisters in America and worldwide to hunt down and kill." The group also warned: "In our next leak we may even disclose secret intelligence the Islamic State has just received from a source the brothers in the UK have spent some time acquiring from the ministry of defense in London as we slowly and secretly infiltrate England and the USA online and off."At the bottom of the ISIS document is an image of the Statue of Liberty with its head cut off. The ISIS hacking division was previously led by Junaid Hussain, a former British Muslim computer hacker from Birmingham who was killed in a US drone strike in Syria last August. His wife, Sally Jones, a Muslim convert from Kent in the UK, is still believed to be involved in the organization, which in the past has urged "lone wolf" attacks against RAF bases in the UK. Inquiries made by 'The Sunday Times' found that the names on the American hit list are genuine. However, the information published by ISIS does not appear to be the result of a leak or genuine hack. Instead, the group seems to have painstakingly gleaned the names of Reaper and Predator drone operators from news articles and military newsletters, before matching them to addresses, photos and other personal details from publicly available sources on the internet. Some of the information appears to have been taken from social media sites, including Facebook and LinkedIn.

SUMMARY GENERATED

Islamic State (ISIS) hackers have published a "hit list" of over 70 US military personnel who have been involved in drone strikes against terror targets in Syria and asked their followers to "kill them wherever they are". The group also claimed that it might have a mole in the UK's ministry of defense and threatened to publish "secret intelligence" in the future that could identify Britain's Royal Air Force (RAF) drone operators. "A nation of cowards that holds no bravery as you resort to sending your remote-controlled unmanned Reaper and Predator drones to attack us from the skies. The group also warned: "In our next leak we may even disclose secret intelligence the Islamic State has just received from a source the brothers in the UK have spent some time acquiring from the ministry of defense in London as we slowly and secretly infiltrate England and the USA online and off." The ISIS hacking division was previously led by Junaid Hussain, a former British Muslim computer hacker from Birmingham who was killed in a US drone strike in Syria last August. Inquiries made by "The Sunday Times" found that the names on the American hit list are genuine. Instead, the group seems to have painstakingly gleaned the names of Reaper and Predator drone operators from news articles and military newsletters, before matching them to addresses, photos and other personal details from publicly available sources on the internet.

V. CONCLUSIONS

We were able to auto-summarize news articles and compare summaries generated by them to analyze what scoring parameters would lead to better results. In the process, we tweaked methods we had researched on to leverage the fact that we were dealing with news articles only. We found that journalists follow a fixed pattern to write a news article. They start with what happened and when it happened in the first paragraph and continue with an elaboration of what happened and why it happened in the following paragraphs. We wanted to use this knowledge while scoring the sentences by giving the nouns appearing in the first sentence a higher score. But after reviewing the preliminary results of our scoring method as described in Barzilay and Elhadad[7], we realized that the first sentence always got a high score since it had nouns that were repeated several times in the article. This is intuitively consistent since the first sentence of the article always has nouns that the article talks about i.e. the topic of the article. In [7], lexical chains were getting created in exponential time. We implemented a linear time algorithm as described in Silber and McCoy[8].

FUTURE WORK

Currently we are only adding nouns in lexical chains. Adjectives too play a major role in defining the important sentences. The future work includes adding adjectives also to the lexical chains along with nouns and then observe the effect on the summary generated. Also, we would like to explore graph based algorithms for sentence scoring and extraction like those mentioned in [9] and [10].

REFERENCES

[1] H. Saggion and T. Poibeau, "Automatic text summarization: Past, present and future", *Multi-source, Multilingual Information Extraction and Summarization*, ed: Springer, pp. 3- 21., 2013

[2] M. Haque, et al., "Literature Review of Automatic Multiple Documents Text Summarization", *International Journal of Innovation and Applied Studies*, vol. 3, pp. 121-129, 2013.

[3] D. R. Radev, et al., "Introduction to the special issue on summarization", *Computational Linguistics*, vol. 28, pp. 399-408, 2002.

[4] C. Fellbaum, "WordNet: An Electronic Lexical Database", *Cambridge, MA: MIT Press.*, 1998

[5] G. A. Miller, "WordNet: A Lexical Database for English", *Communications of the ACM*, Vol. 38, No. 11: 39-41., 1995

[6] Morris, Jane and Graeme Hirst. "Lexical cohesion computed by thesaural relations as an indicator of the structure of text", *Computational linguistics* 17.1, 21-48, 1991

[7] R. Barzilay, & M. Elhadad, "Using lexical chains for text summarization", *Advances in automatic text summarization*, 111-121, 1999

[8] H. G. Silber, & K. F. McCoy, "Efficiently computed lexical chains as an intermediate representation for automatic text summarization", *Computational Linguistics*, 28(4), 487-496. , 2002

[9] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing, Discourse segmentation of multi-party conversation, in *Annual Meeting-Association for Computational Linguistics*, vol. 1. Association for Computational Linguistics, 2003

[10] S. Brin and L. Page, The anatomy of a large-scale hyper textual Web search engine, *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 107117, 1998