International Journal of Computer Trends and Technology
ISSN: 2231 – 2803 / https://doi.org/10.14445/22312803/IJCTT-V69I5P111
Volume 69 Issue 5, 77-81, May 2021

Original Article

Text Summarization of News Events using Semantic Triples

Shikha Singh[1], Garima Srivastava[2]

[1,2] Assistant Professor, Department of Computer Science & Engineering, Amity School of Engineering and Technology, Amity University (Lucknow Campus), Uttar Pradesh, India.

Received Date: 29 April 2021
Revised Date: 02 June 2021
Accepted Date: 07 June 2021

Abstract *- Text summarization is basically used to generate a compact version of the original document for the article. The summarization task can be challenging due to the same report is generated by different people of diverse opinions. But here, major issues are to rectify redundant information or relevant information for text summarization. Currently, many techniques are available in the market, and in which modeling events as semantic triples is one of them. In semantic triple, triples are weighted Based on frequencies and then to form a summary. Generally, triples are extracted from the statement of the report which may sometime lose important information. This paper focuses on lossless summarisation with the help of graph structure. Summary sentences are generated by picking the top-rated path from a complete structured graph with the maximum number of triples and grammatical correctness. Here we have also done several improvements to rectify the limitations of the model. We have included entity linking and verb linking to eliminate the limitation of coverage, correctness, and grammaticality.*

Keywords *- Extractive summary, Abstractive summary, PSB, PGF.*

I. INTRODUCTION

A huge number of articles are published daily to cover different news events. Reading such huge number of articles daily to get overall news is very much challenging for readers. Readers want a concise summary to get what is happening around them. To solve this, multiple techniques are developed to get quick access to essential information. Here, the main challenges are to cover different perspectives, different views, and different levels of the same news event. So here we have used abstractive way of summarisation to get more lumen like summary which is a more compact version. In abstractive summarization, there are how-to approaches one is Phrase-selection based (PSB) and another is Pattern-graph Fusion (PGF) [5]. In this paper authors are using PGF abstractive approach to generate a summary for the report. With the progress of the model certain limitations in the approach was encountered. To overcome the limitations, authors added some features like entity linking and verb linking. These features will improve the model by adding more coverage and correctness. After developing a model, we need to find out the standard of the generated summary by evaluating the model. Generate models can be evaluated by ROUGE techniques as well asby humans also. Evaluation is done by comparing the generated summary to the reference summary.

II. RELATED WORK

There are two ways of doing summarization, the first one is Extractive and the second one is Abstractive summarization. In extractive summarization, it simply try to extract the most relevant and important pieces of information from the document and combine them to generate a summary without doing any sort of modification. On the other hand in abstractive summarization, after extracting important and relevant information it checks for similar sentences. Similar sentences are combined to generate novel sentences and words for the summary. For example,
"Hurricane" "Nate" "slammed" "Louisiana".
"Nate" "struck" "the" "State" "of" "Louisiana".
"The" "hurricane" "killed" "2" "people".

A. Extractive Summary

"Hurricane Nate slammed Louisiana."
"The hurricane killed 2 people."

B. Abstractive Summary, with Conjoined Facts

"Hurricane Nate slammed Louisiana, killing 2 people."

Here, it can be seen that abstractive summarization is producing a more human-like and concise summary because of conjoined facts. In extractive summarization, it is just copying the different sentences. So, in this paper abstractive summarization is preferred over extractive summarization due to mentioned reasons [5].

III. ABSTRACTIVE NEWS SUMMARIZATION APPROACHES

There are two approaches for the Abstractive News Summarization [6].

- Phrase-selection based (PSB)
- Pattern-graph Fusion (PGF)

In PSB, it tentatively pairs subject and verb phrases from different sentences and checks for the compatibility. So there is always a chance of losing information or being redundant. Whereas in PGF, it looks for similar tokens in different sentences and fuses them to form a connected graph. There is a strong possibility of losing some information or being redundant in PSB. So, for this paper, PGF is selected over PSB.

IV. PATTERN GRAPH FUSION (PGF)

PGF is diagrammatically shown in figure 1. PGF consist of three actions:

A. Extracting

In this phase, the collection of documents to extract subject, predicate, and object from the sentences using Ollie (Open Information extraction technique) [1].

B. Typing

In this phase, try to annotate the subject or the object with the typing [8]. In other words, try to label the subject or the object like "people" as "PERSON", "Louisiana" as "LOCATION". The annotation is done using Stand ford NER [14] and SEMAFOR [15].

C. Graph Fusion

In this phase, try to combine all these pipelines results in one connected graph.

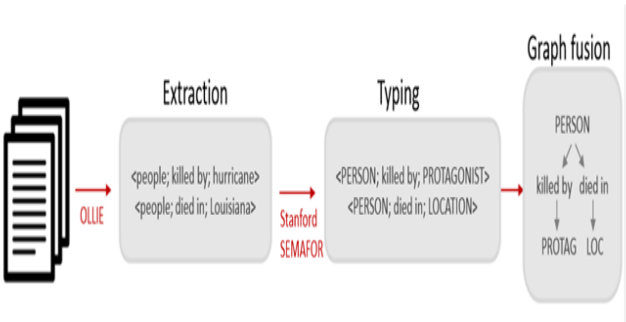

Fig 1. Pattern Graph Fusion (PGF)

Once the connecting graph is created, select the path which covers the maximum number of nodes in the graph. This select path will generate the required summary for that particular document. When two statements "Hurricane Nate slammed Louisiana with 85 mph winds" and "Paul sent his prayer for the Louisiana" are summarized. In the typing phase, "Nate" and "Paul" is going to be annotated by "PERSON". While generating summary, it may create ambiguity in statements like "Nate sent his prayer for Louisiana with 85 mph winds". Typing might lead to an incorrect result.

Sometimes subject or object may be denoted with the same label which leads to an incorrect result. For example, Paul and Nate both are the subjects which means they are going to be denoted as "PERSON" in the Typing phase which leads to in correct results.

V. PATTERN GRAPH FUSION: LIMITATIONS

A. Coverage

It is used to have control over and under production of target words. Some words may not get enough attention which may result in poor summarization. Low Coverage may also be due to merging misses semantically similar verbs. For example, “withdraw” vs “pull out” both means the same but treated differently by the algorithm, may leads to lowering of coverage.

B. Grammaticality

While merging of different statements by the extract-based summarization systems leads to ungrammatical sentence. For example, “The US pulled out from the Paris Agreement caused disappointment among environmentalists”.

VI. PROPOSED SYSTEM

To overcome the mentioned limitations we need to improve our model by including features like Correctness and coverage by Entity Linking, Coverage by Verb Linking, and Grammaticality by Grammatical Fixing. Figure 2 reflects the summarization pipeline. Now our updated Summarization pipeline includes:

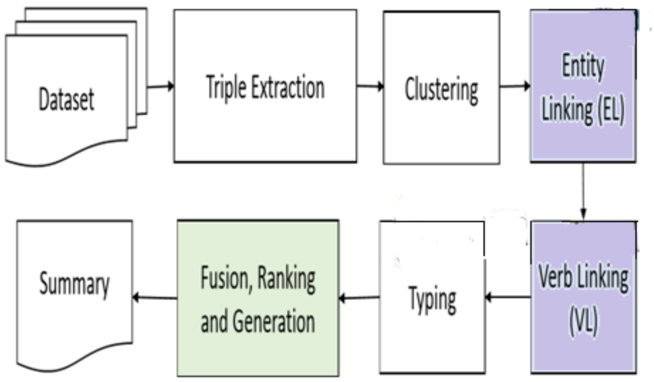

Fig. 2 Summarization Pipeline

A. Dataset

In the model, we have chosen DUC 2004 dataset for text summarization. It is the collection of newspaper articles or documents from TREC. It has the original summaries. Additionally, the datasets are public. Therefore, many researchers select DUC 2004 to study text summarization. For each document, humans wrote two summaries: one is 200 words and the other is 400 words. The DUC 2004 comprises of 50 news topics, each topic has 10 news article and 4 human summaries [1].

B. Triple Extraction

In this stage, each sentence gets converted into triples (set of 3 entities made up of statement in the form of "Subject-Predicate-Object"). Extraction is done by the OLLIE. Open information extraction (OLLIE) indicates to the extraction of relational triples, mainly binary relations, from plain text, such as *(Paul; founded; champ)*. The primary difference from other information extraction is that the *schema* for these relations does not need to be specified in advance; the relation is just the text linking two arguments. For example, *Anurag was born in India* will create a triple as *(Anurag; was born in; India)*, where the relation between subject and object is "was-born-in".

C. Clustering

It is a task of breaking the sentences or triples into a similar group such that sentences or triples are in the same groups are more similar to other sentences or triples in the same group and dissimilar to the sentences or triples in other groups [4].

D. Entity Linking

While finding similar triples, entities are important. The first entity of triples can be a PERSON, an ORGANIZATION, a LOCATION, or any well-defined concept [5]. There are two issues with entity recognition, first is an existing tool that doesn't know what entity is. The second is entities are not always used with full names, sometimes used with abbreviations or last name of the people, which we call alias. So, there is no way to detect identical entities with the same meaning.

To overcome the above problem, we will perform entity linking [6]. We will do entity recognition with the help of DBpedia [10]. It is a huge-scale extraction system using Wikipedia. It's a graph database that uses the RDF (Resource Description Framework) format. It uses the name normalization technique to convert all alias to normalize name facilitate entity extraction. In short, it does the task of assigning a unique identity to the entities mentioned in triples. At last, we will apply Stanford Deterministic Co reference Resolution [10] to map Co references of different entities.

E. Verb Linking or Predicate Similarity

Some predicates, which are represented as verbs have the same meaning. For example, the two triples, <Paul; fired; James> and <Paul; dismissed; James> are basically the same. So, this can't be detected if both the predicates are treated as different words. To solve this problem, we will use WordNet, which has a similarity [13] as the parameter to detect similar verbs or predicates and uses only one representative word or a verb for them. Similarity parameter return score for each pair of predicates, if the similarity score is more than 90% then only, we will fuse them and can say both have very close meanings.

F. Fusion Graph

For this we generally follow the baseline approach to make a fusion graph of each group of similar triples. Firstly, we will follow the baseline approach strictly. Construct a graph by iteratively adding patterns to it. A node is going to be added to the constructed graph for each word(token) in the pattern, where consecutive words are linked with directed edges. When we add a new pattern, a token of the pattern is merged with the existing node of the graph, providing that they have the same Part Of Speech (POS) tag [5]. Here, we have to keep a watch on some words like "he" and "his" because they have the same POS tag but they should not be merged. Also, stop words like "the", "to" and "of" should not be merged in order to avoid noise in the summary. Without the annotation, each pattern will be the sentence of the original text only. Triples are formed to identify the predicates and arguments and to perform annotation and triples similarity checks. We will do the strict merging, where merging is done only for matching entities and predicates not with the other nodes. The basic idea behind this is to avoid concatenating of triples that are not compatible. For example, the fuse of the two triples <Herry; told; Ron to stop his investigation about India> and <Herry; fired; Ron because of his investigation about Clinton> may lead to a summary as "Herry told Ron to stop his investigation about Clinton" which is incorrect.

G. Summary Sentence Selection

Sentences to generate summary are selected from the fusion graph. One directed edge of the graph represents a single sentence for a summary. The rank of path depends upon two factors. First is grammaticality and second is triples coverage. So, the highly ranked path must have covered many paths which means it summarizes several facets of the same facts. Furthermore, they must be grammatically correct [11].

A sentence can either be grammatically correct or grammatically incorrect. So, we have to focus on “how suitable or acceptable are the sentence structures to be a part of a summary? We have performed a grammatical fix, which is done by transforming the verb phrase into a well-formed clause using a relative pronoun or a participle [7]. This is generally done by analyzing grammatical dependency to detect the dangling verbs i.e. verbs that are not correctly attached to the subjects and entity typing to determine the correct pronoun. Finally, sentences without verbs are dropped.

To enhance path ranking exploiting, authors have used node degree with pattern coverage and grammaticality. A node degree defined as the total number of incoming and outgoing edges of the node. The main idea is to select the path which has the most important node, which has a node with the highest degrees. Path ranking is a multi-step pair-wise comparison in the following order: 1) Pattern Coverage, 2) Node Degree, and 3) Grammaticality. For Node Degree, first, we compare the average number of degrees then the total number of degrees of two paths. Finally, to support our grammatical checker and fixer model, we will set precedence order in the following sequence: originally grammatical path, grammatically fixable paths, and ungrammatical, non-fixable path [12].

VII. EVALUATION

Summary evaluation is a challenging job because there is no ideal summary for a document or a collection of documents [3]. The definition of a valid summary is an open question to a large extent. It has been observed that human summarizers have a low satisfaction for evaluating and producing summaries. Furthermore, the common use of many metrics, as well as the lack of a standard assessment metric, has made summary evaluation complex and challenging.

A. Evaluation of Automatically Produced Summaries

Since the late 1990s, there have been many appraisal campaigns in the United States [3]. These conferences serve an important role in the creation of assessment criteria and the evaluation of summaries based on both human and automated scoring. To conduct an automated summary assessment, we must resolve three major challenges: I The most relevant sections of the original text that must be retained must be decided and defined. ii) Since this information can be presented in a variety of ways, automatically recognise these pieces of important information in the provided description. iii) The readability of the summary must be assessed in terms of grammaticality and coherence.

B. Human Evaluation

The most straightforward way to measure the accuracy of a summary is to have a person do so. The judges, for example, will assess the summary's coverage, or how well the produced summary covered the original input. The judges then assess how well a description responds to the given question. Non-redundancy, grammaticality, incorporation of most important pieces of knowledge, structure, and coherence are all factors that human experts must consider when scoring each produced description. For more information [3].

C. Automatic Evaluation Methods

Since the early 2000s, several sets of metrics have been available to automatically test summaries. The most commonly used metric for automated summary assessment is ROUGE.

a) ROUGE

ROUGE (Recall Oriented Understudy for Gisting Evaluation) compares a summary to human (reference) summaries to automatically assess its consistency. ROUGE [4] comes in a variety of flavours, and we'll only go over the most common ones here.

b) ROUGE-n

This metric is based on a comparison of n-grams and is a recall-based calculation. From the reference summaries and the produced description, a series of n-grams (mostly two and three, but rarely four) are extracted [9]. Assume p is "the number of common n-grams between the reference and produced summary," and q is "the number of n-grams extracted solely from the reference summary". The score is calculated as follows:

i. ROUGE-n =p/q

c) ROUGE-L

Between the two text sequences, this metric uses the definition of the longest common subsequence (LCS). The idea is that the more similar two summary sentences are, the longer the LCS between them is. This metric is more versatile than the previous one, but it has the downside of requiring all n-grams to be consecutive [4].

d) ROUGE-SU

This metric, known as skip bi-gram and uni-gram ROUGE, takes into account both uni-grams and bi-grams. It allows for the inclusion of terms between the first and last words of the bi-grams, allowing them to be non-consecutive word sequences.

VIII. CONCLUSION

As the Internet has grown in popularity, a vast amount of knowledge has become accessible. It's difficult for readers to summarise a vast volume of information. As a result, in this era of information overload, automated summarization tools are in high demand. We discussed different abstractive methods for multi-document summarization in this paper. While it is impossible to cover all of the different approaches in this paper, we believe it offers useful insight into recent developments and advancements in automated summarization methods. With this much detail, we can claim that PGF is best suited for abstractive summarization.

IX. FUTURE SCOPE

In this paper, authors have tried to automate text document into text summary using semantic triples ("subject", "predicate", and "object"). Following will be future scope:

A. Add More Fine-Grained Representation of Facts

Here we have simply conjoined the subject, predicate, and object of one statement to another statement. In the future, we will try to add more fine-grained representation of facts like "On Saturday", "After leaving", "With winds of" in automated summary.

B. Fluency

We will try to add more fluency to machine automated summary to make it more human-like and more readable.

C. Audio, Video, Image

Try to generate text summary from audio, video clips, and images using available algorithms and techniques.

REFERENCES

[1] Amato, F., d'Acierno, A., Colace, F., Moscato, V., Penta, A., Picariello, A.: Semantic summarization of news from heterogeneous sources. Advances on P2P, Parallel, Grid, Cloud and Internet Computing. LNDECT, 1, 305–314.Springer, Cham (2017).

[2] Arora, S., Liang, Y., Ma, T.: A simple but tough-to-beat baseline for sentence embeddings (2016)

[3] Horacio Saggion and Thierry Poibeau. 2013. Automatic text summarization: Past, present and future. In Multi-source, Multilingual Information Extraction and Summarization. Springer, 3–21

[4] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In Text Summarization Branches Out: Proceedings of the ACL-04 Workshop. 74– 81.
[5] Genest, P.-E., Lapalme, G.: Framework for abstractive summarization using text-to-text generation. In: Proceedings of the Workshop on Monolingual Text-To-Text Generation, pp. 64–73. Association for Computational Linguistics (2011).
[6] Khan, A., Salim, N., Kumar, Y.J.: A framework for multi-document abstractive summarization based on semantic role labelling. Appl. Soft Comput. 30, 737–747 (2015).
[7] Del Corro, L., Gemulla, R.: ClausIE: Clause-based open information extraction. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 355–366. ACM (2013).
[8] Kshirsagar, M., Thomson, S., Schneider, N., Carbonell, J., Smith, N.A., Dyer, C.: Frame-semantic role labeling withheterogeneous annotations. In: ACL, 2(2015) 218–224.
[9] Lin, C.-Y., Hovy, E.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of the 2003 NAACL, vol. 1, pp. 71–78. Association for Computational Linguistics (2003).
[10] Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: DBpedia spotlight: shedding light on the web of documents. In: Proceedings of the 7th International Conference on Semantic Systems, 1–8. ACM (2011).
[11] Nenkova, A., Passonneau, R.J.: Evaluating content selection in summarization: The Pyramid method. In: HLT-NAACL, 4 (2004) 145–152.
[12] Oya, T., Mehdad, Y., Carenini, G., Ng, R.: A template-based abstractive meeting summarization: leveraging summary and source text relationships. In: Proceedings of the 8th International Natural Language Generation Conference (INLG), pp. 45–53. Association for Computational Linguistics, Philadelphia, June 2014.
[13] Pedersen, T., Patwardhan, S., Michelizzi, J.: WordNet::Similarity: Measuring the relatedness of concepts. In: Demonstration Papers at HLT-NAACL 2004, pp. 38–41. Association for Computational Linguistics (2004).
[14] Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J.R., Bethard, S., McClosky, D.:. The Stanford coreNLP natural language processing toolkit. In: ACL (System Demonstrations), (2014) 55–60.
[15] Kshirsagar, M., Thomson, S., Schneider, N., Carbonell, J., Smith, N.A., Dyer, C.: Frame-semantic role labeling with heterogeneous annotations. In: ACL, 2 (2015) 218–224.