IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Hybrid Approach For Automatic Text Summarization

[1]Chandra Sekhar Vinukonda, [2]T. Sharat Kumar, [3]Victor Raj, [4]Rupesh Kumar, [5]Sanjeeta Mistry

[1]Assistant Professor, [2]Under Graduate, [3]Under Graduate, [4]Under Graduate, [5]Under Graduate

[1]Department of Computer Science & Engineering,

[1]Dr. B. R. Ambedkar Institute of Technology, Port Blair, India

Abstract: Automatic text summarization is a method to generate summary of any input text. Commonly, it is classified as extractive summarization or abstractive summarization. Extractive summarization is used for extracting important sentences, whereas, abstractive summarization is used to generate entirely new phrase. The proposed method uses a hybrid approach, in which both extractive and abstractive summarization is performed. So, the generated abstractive summary will only have the important data obtained from the extractive summary. This will help to remove irrelevant information, and will produce more grammatically correct summary of any input text. In our study, we have found our method is performing better than some other summarization techniques. The hybrid approach helped to produce more precise summary in abstractive form. The proposed method produced an average ROUGE score of 0.239.

Index Terms - Automatic Text Summarization, Extractive Summarization, Abstractive Summarization, Sentence Ranking.

I. INTRODUCTION

Internet is holding a huge amount of textual data, and most of them are present in unstructured format. There is a great need to reduce much of this text data to shorter, focused summaries that capture the salient details, so we can read and navigate through it easily. An effective solution to such a problem is automatic text summarization. Automatic text summarization is the task of condensing a piece of text to a shorter version, while preserving the important points and meaning of the content. Automatic text summarization can be classified into various categories, but the most common classification is extractive text summarization and abstractive text summarization. In extractive text summarization, the generated summary will hold the important sentences, which are present in the original content. However, the generated summary may not be grammatically correct, and sometimes meaningless. Abstractive text summarization involves deeper analysis of the text. The generated summary will have a completely new phrase, which will actually be a shorter version of the original content. The generated abstractive summary will be more grammatically correct, preserving the meaning of the original content.

At present, there are various summarization models, which can generate precise summaries of any input text. Some of them can generate extractive summary and some can generate abstractive summary only. In case of extractive summarization models, they only extract important sentences from the original text and present them as the summary. These summaries usually have grammatical mistakes and may not have any meaning, in most of the cases. In case of abstractive summarization models, they produce a completely new phrase as summary of the original text. The generated summaries will have less grammatical errors and are more meaningful than extractive summaries. But, sometimes, they may not be able to remove irrelevant data and include them also in the generated summary. To overcome such issues, we proposed a hybrid approach for text summarization.

In the proposed solution, the summary is generated through a hybrid approach, means both extractive and abstractive summarization techniques are used. The model consists of two phases. On the first phase, the model generates an extractive summary of the input text, with an improved sentence ranking algorithm. In the second phase, the model considers the output of first phase as input, and generates an abstractive summary using LSTM. The final summary will then contains less irrelevant content from the original text. The extractive summarization step acts as a data pre-processing phase for abstractive summarization model, as it helps to remove irrelevant sentences from the original text, and passes only relevant data as input text for generating abstractive summary. Detailed explanation of the proposed method is given in section III.

II. LITERATURE SURVEY

S. No.	Year	General/ Conference	Authors	Task	Method	Paper
1.	2020	General	Chujie Zheng, Kunpeng Zhang, Harry Jiannan Wang, Ling Fan	Abstractive Text Summarization	It proposes a topic-aware abstractive summarization framework by leveraging the underlying semantic structure of documents represented by their latent topics. The design is able to learn and preserve global semantics of documents and thus makes summarization effective.	*Topic-Aware Abstractive Text Summarization*, arXiv:2010.10323v1 [cs.CL] 20 Oct 2020
2.	2019	General	Darshna Patel, Saurabh Shah, Hitesh Chhinkaniwala	Extractive Text Summarization	The proposed model summarizes multiple documents using fuzzy logic. The fuzzy logic system extracts the most salient sentences from multiple documents, passes it to calculate similarity scores using cosine similarity, and rank the sentences. Then, it picks the top ranked sentences and performs redundancy removal. The output obtained is considered as the extractive summary of multiple documents.	*Fuzzy logic based multi-document summarization with improved sentence scoring and redundancy removal technique*, Elsevier, 2019
3.	2019	Conference	J. N. Madhuri, Ganesh Kumar R.	Extractive Text Summarization	In the proposed approach, the model converts the input text into keywords by removing stop words. Then it calculates the frequency of occurrence of each keyword in the document. Sentence ranking is performed by calculating the maximum frequency of each sentence, and the output is considered as extractive summary.	*Extractive Text Summarization using Sentence Ranking*, IEEE, 2019
4.	2019	General	Sandhya Singh, Kevin Patel, Krishnanjan Bhattacharjee, Hemant Darbari, Seema Verma	Extractive Text Summarization	It proposes a two-stage mechanism to perform single document summarization via multi-document summarization technique. The approach involves the use of popular extractive summarization algorithms to generate summaries which are then further processed as multi-document summarization instance.	*Towards Better Single Document Summarization using Multi-Document Summarization Approach*, IJCSE, Vol.-7, Issue-5, May 2019
5.	2019	General	Aruna Kumara B., Smitha N. S., Yashaswini Patil, Shilpa P., Sufiya	Extractive Text Summarization	The proposed model generates extractive summary using Ranking Algorithm. The model	*Text Summarization Using Ranking Algorithm*, IJCSE,

					ranks the sentence based on their similarity score, which is calculated through cosine similarity of sentence vectors. The top-n sentences are selected to produce the summary.	Vol.-7, Special Issue-14, May 2019
6.	2018	Conference	Mahsa Afsharizadeh, Hossein Ebrahimpour-Komleh, Ayoub Bagheri	Extractive Text Summarization	In this approach, a query-oriented text summarization technique is proposed by extracting the most informative sentences. A number of features are extracted from the sentences, each of which evaluates the importance of the sentences from an aspect. A score is assigned to each sentence based on a linear function of its feature values. The sentences are ranked based on the calculated score, and then top-n sentences are extracted as summary.	*Query-oriented Text Summarization using Sentence Extraction Technique*, IEEE, 2018
7.	2018	General	Chandra Khatri, Gyanit Singh, Nish Parikh	Abstractive Text Summarization and Extractive Text Summarization	It proposes a novel Document-Context based Seq2Seq models using RNNs for abstractive and extractive summarizations. It uses an RNN architecture wherein a document context vector is passed as an input at first time step along with the document sequence in the encoder.	*Abstractive and Extractive Text Summarization using Document Context Vector and Recurrent Neural Networks*, arXiv:1807.08000v2 [cs.CL] 29 Jul 2018
8.	2018	General	M. R. Prathima, H. R. Divakar	Extractive Text Summarization	The proposed model tries to improve the quality as well as performances of the summary generated by the clustering technique by cascading it with Support-Vector-Machine (SVM). Various similarity measures are utilized in order to identify the similarity between the sentences of the document and then they are grouped in cluster on the basis of their term frequency and inverse document frequency (TD-IDF) values of the words.	*Automatic Extractive Text Summarization Using K-Means Clustering*, IJCSE, Vol.-6, Issue-6, June 2018
9.	2017	General	Abigail See, Peter J. Liu, Christopher D. Manning	Abstractive Text Summarization	The proposed model uses a hybrid pointer-generator network that can copy words from the source text via pointing. The pointer-generator network is a hybrid between baseline and a pointer network, as it	*Get To The Point: Summarization with Pointer-Generator Networks*, arXiv:1704.04368v2 [cs.CL] 25 Apr 2017

					allows both copying words via pointing, and generating words from a fixed vocabulary. A coverage model is to solve the problem of repetition.	
10.	2016	General	Ramesh Nallapati, Bowen Zhou, Cicero dos Santos	Abstractive Text Summarization	The proposed model produces abstractive text summarization using Attentional Encoder-Decoder Recurrent Neural Networks. The model uses one embedding vector each for POS, NER tags and discretized TF and IDF values, which are concatenated together with word-based embeddings as input to the encoder.	*Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond*, arXiv:1602.06023v5 [cs.CL] 26 Aug 2016
11.	2015	General	Alexander M. Rush, Sumit Chopra, Jason Weston	Abstractive Text Summarization	It proposes a local attention-based model that generates each word of the summary conditioned on the input sentence. The model consists of attention-based encoder, that trains the model to generate new words for the summary, based on given training data.	*A Neural Attention Model for Abstractive Sentence Summarization*, arXiv:1509.00685v2 [cs.CL], 3 Sep 2015

Table 2.1 – Literature Survey of Automatic Text Summarization.

III. PROPOSED METHODOLOGY

The proposed method performs both extractive and abstractive summarization to produce the final output. First, it takes the input from the user and extracts the text from it. Then, the obtained text is pre-processed which involves splitting sentences, tokenization, lemmatization and stop-word removal. After pre-processing, we get a multidimensional array containing the words of each sentence. Then the pre-processed data is passed to the extractive summarization model, where each sentence is ranked, based on their similarity score, and the top-n sentences are picked to form the extractive summary. The extractive summary is passed as input for the abstractive summarization model, and the generated summary from abstractive summarization model is considered as the final output. For abstractive summarization, we used LSTM, which is an advanced RNN, capable of learning long-term dependencies. A detailed architecture of the proposed method and explanation of steps involved is mentioned below –

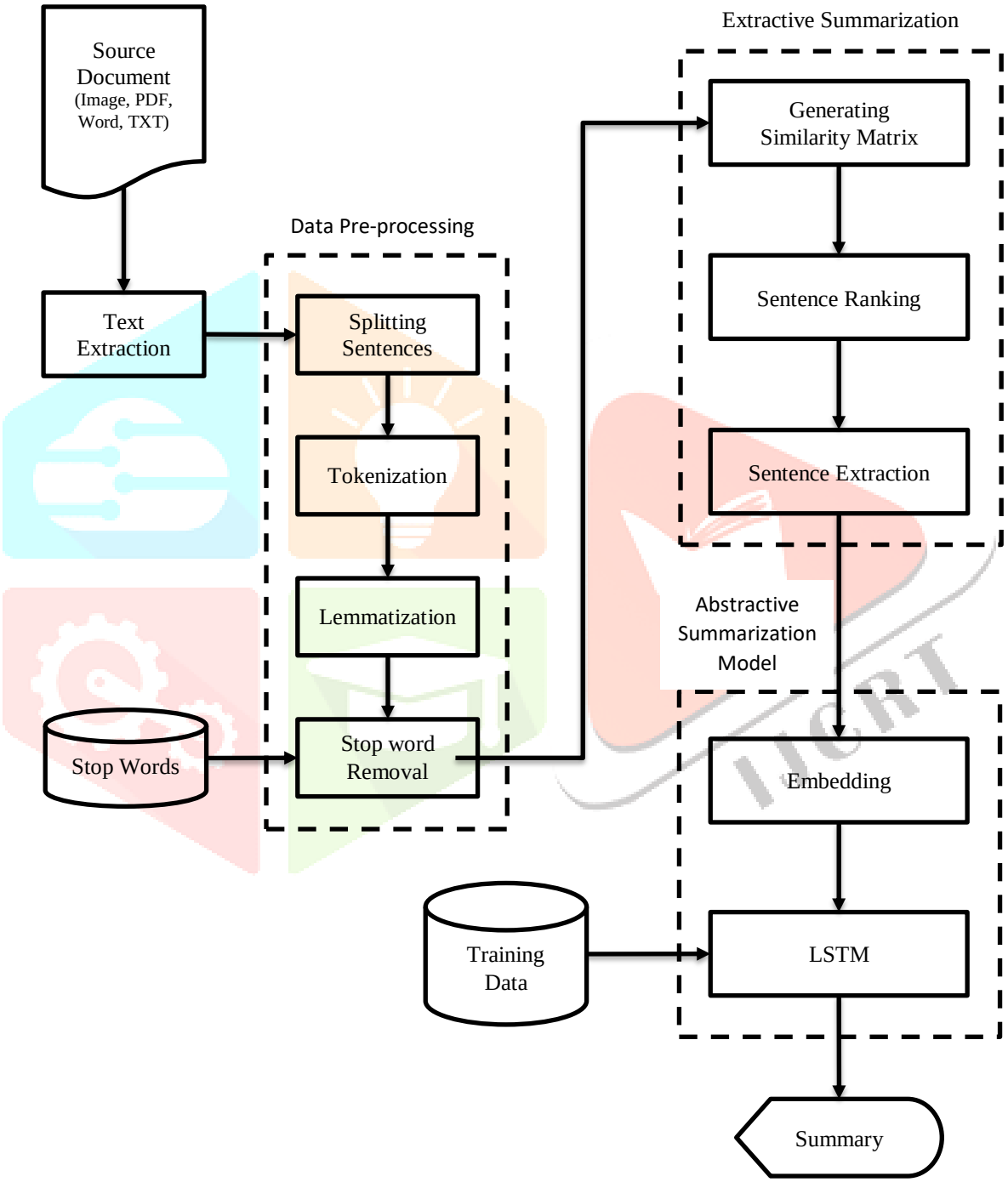

Fig. 3.1 – Architecture of proposed hybrid method for automatic text summarization.

A. Source Document

The source document can be image, word document, pdf document, text file, or simply an input text.

B. Text Extraction

In case, if the input is not a text, but a file from which the text needs to be extracted, this phase will use various algorithms for different kinds of files, and extract texts from the document.

C. Data Pre-processing

Data pre-processing involves various steps –

1. Splitting Lines

In order to generate the similarity matrix of lines present in the original text, first we need to split the lines present in it.

2. Tokenization

Tokenization is generally a process of splitting a sentence into words. Each word present in the sentence will be referred as tokens.

3. Lemmatization

Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. Lemmatization helps in reducing the redundancy of words.

4. Stop word removal

Stop words are a set of commonly used words in any language. They are usually not so important in summary generation. So, we can remove them temporarily from the sentences for summary generation.

D. Generating Similarity Matrix

Similarity matrix will be constructed by calculating the similarity score of a sentence with respect to all other sentences present in the input text. Consider that the input text consists of n-sentences, then the dimensions of the similarity matrix will be marked as S_1, S_2,..., S_n. Now, we will take two sentences S_i and S_j, such that $S_i \neq S_j$. To calculate the similarity score of sentence S_i with sentence S_j, we will count the occurrence of words of sentence S_i in sentence S_j. For example, if two words of sentence S_i is present in sentence S_j, then the similarity score of sentence S_i with sentence S_j, will be 2. Similarly, we will calculate the similarity score of all other sentences and update them in the similarity matrix. In case of same sentences, the similarity score will be zero.

E. Sentence Ranking

After constructing the similarity matrix, we need to rank the sentences based on their actual similarity score. The actual similarity score is the sum of all the similarity scores of a sentence with respect to all other sentences present in the input text. Now, the sentences are arranged in descending order based on their actual similarity score.

F. Sentence Extraction

To construct the extractive summary of the given input text, we will extract the top-n sentences based on their ranking from the previous step. The extracted sentences will then be rearranged in chronological order. Then, it will be considered as the extractive summary of the given input text, and passed to the next phase as input text.

G. Abstractive Text Summarization

In this type of summarization, new sentences are formed which are generally not present in the original text. For generating the abstractive summary, we have used an encoder-decoder sequence to sequence model with LSTM(Long-Short Term Memory), as shown in Fig 3.2.

The model consists of three parts: encoder, intermediate vector and decoder.

- **Encoder**

The encoder processes each token in the input-sequence. It tries to cram all the information about the input-sequence into a vector of fixed length i.e., the ‘context vector’. After going through all the tokens, the encoder passes this vector onto the decoder.

- **Intermediate Vector**

The vector is built in such a way that it's expected to encapsulate the whole meaning of the input-sequence and help the decoder make accurate predictions.

- **Decoder**

The decoder reads the context vector and tries to predict the target-sequence token by token.

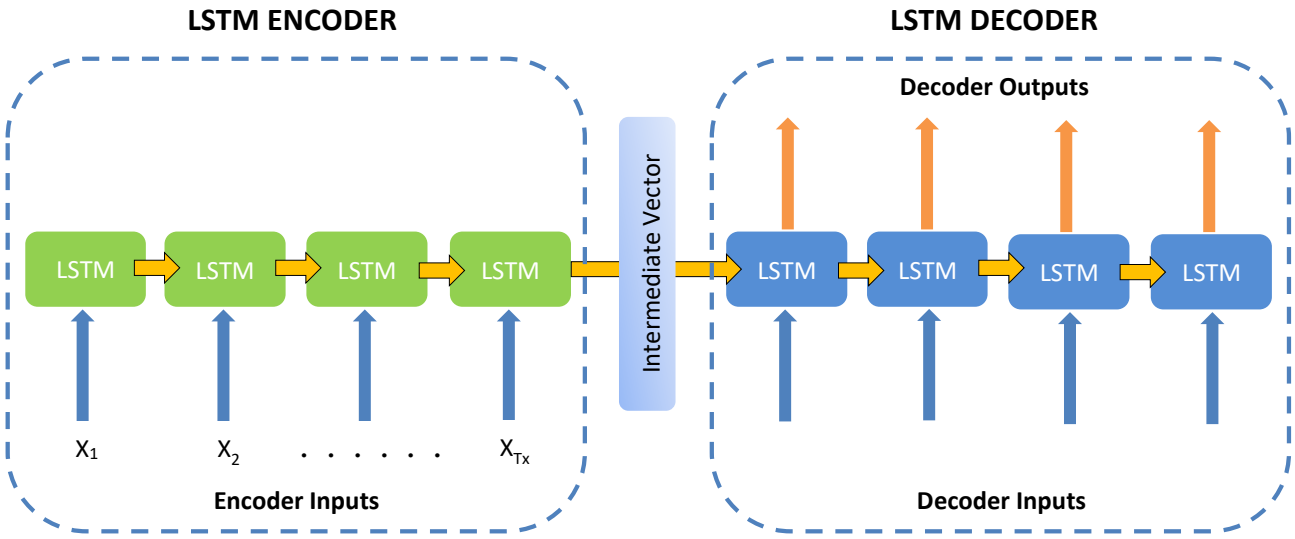

Fig. 3.2 – Seq-2-Seq model using LSTM.

In addition, we have used LSTM with Attention mechanism, which allows the decoder to pay attention to different parts of the source sequence at different decoding steps.

IV. RESULTS & DISCUSSION

In the proposed model, we have used both extractive and abstractive method, so that we can get a more precise summary in abstractive form. We have observed that our method is extracting important sentences from the original text, and producing a topic relevant summary as output. The proposed method improved the quality of summary, because it has used only relevant text as input of abstractive summarization model.

	Recall	Precision	F1-Score
Rouge-1	0.464	0.486	0.475
Rouge-2	0.312	0.286	0.298
Rouge-3	0.384	0.302	0.338
Rouge-4	0.012	0.024	0.016
Rouge-L	0.387	0.328	0.355
Rouge-W-1.2	0.329	0.128	0.184
Rouge-S*	0.186	0.121	0.147
Rouge-SU*	0.112	0.096	0.103
Average	0.273	0.221	0.239

Fig. 4.1 – ROUGE metrics of proposed method.

Fig 4.1 shows the rouge scores produced by our proposed method. The average F1-Score is 0.239, which is better than other methods. In most of the previous methods, either extractive summarization is used or abstractive summarization is used, because of which the summary generated from such methods produces less accurate summaries. Those summaries may have all the important contents, but they may not be grammatically correct, or they may have more irrelevant data. These disadvantages lead to less ROUGE scores of such methods as compared to our method. Because, we have used a combination of both extractive and abstractive summarization, the summary generated from our proposed method will have less grammatical mistakes, and also will contain less irrelevant data.

A comparison of some previous automatic text summarization methods and our proposed method is mentioned in Table 4.1.

	FB	KLD	LR	LSA	MTR	TAAS	Proposed Method
Rouge-1	0.293	0.313	0.323	0.308	0.306	0.386	0.475
Rouge-2	0.147	0.123	0.146	0.151	0.152	0.156	0.298
Rouge-3	0.091	0.071	0.087	0.093	0.094	0.089	0.338
Rouge-4	0.063	0.048	0.059	0.064	0.064	0.059	0.016
Rouge-L	0.276	0.290	0.303	0.290	0.288	0.354	0.355
Rouge-W-1.2	0.151	0.147	0.158	0.156	0.155	0.176	0.184
Rouge-S*	0.078	0.085	0.093	0.084	0.085	0.124	0.147
Rouge-SU*	0.079	0.088	0.095	0.085	0.087	0.127	0.103
Average	0.147	0.146	0.158	0.154	0.154	0.184	0.239

Table 4.1 – Comparison of different other methods with our proposed method.

The generated summary from the proposed document is obtained in abstractive form, making it more grammatically correct, as compared to the summary generated from extractive summarization models. This makes the summary to be easily understandable, and precise, as it also doesn't contain any irrelevant data from the original text. This method can be extended by using deep learning-based sentence ranking approach.

V. CONCLUSION & FUTURE SCOPE

In this paper, we have discussed the implementation of a hybrid approach of automatic text summarization. The hybrid approach involves both extractive and abstractive summarization techniques. Improved TextRank algorithm is used to generate extractive summary, and LSTM is used to generate the abstractive summary. We have observed that the proposed method performs better than some other text summarization techniques. In future work, deep learning approach can be implemented to produce extractive summary. Also, we can use larger dataset to enhance the abstractive summarization model. Topic aware summarization can also be implemented which may help to generate summaries, more focused on the topic. Bi-Directional LSTM can also be used, which is capable of capturing context from both the directions. Still a lot of improvements can be done in automatic text summarization techniques, to produce more precise and meaningful summary.

VI. REFERENCES

[1] Chujie Zheng et. al., *"Topic-Aware Abstractive Text Summarization"*, arXiv:2010.10323v1 [cs.CL], 20 Oct, 2020.

[2] Darshna Patel et. al., *"Fuzzy logic based multi-document summarization with improved sentence scoring and redundancy removal technique"*, Elsevier, 2019.

[3] J. N. Madhuri, Ganesh Kumar R., *"Extractive Text Summarization using Sentence Ranking"*, IEEE, 2019.

[4] Sandhya Singh et. al., *"Towards Better Single Document Summarization using Multi-Document Summarization Approach"*, IJCSE, Vol.-7, Issue-5, May 2019.

[5] Aruna Kumara B. et. al., *"Text Summarization Using Ranking Algorithm"*, IJCSE, Vol.-7, Special Issue-14, May 2019.

[6] Mahsa Afsharizadeh et. al., *"Query-oriented Text Summarization using Sentence Extraction Technique"*, IEEE, 2018.

[7] Chandra Khatri et. al., *"Abstractive and Extractive Text Summarization using Document Context Vector and Recurrent Neural Networks"*, arXiv:1807.08000v2 [cs.CL] 29 Jul, 2018.

[8] M. R. Prathima, H. R. Divakar, "*Automatic Extractive Text Summarization Using K-Means Clustering"*, IJCSE, Vol.-6, Issue-6, June 2018.

[9] Abigail See et. al., *"Get To The Point: Summarization with Pointer-Generator Networks"*, arXiv:1704.04368v2 [cs.CL] 25 Apr, 2017.

[10] Ramesh Nallapati et. al., *"Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond"*, arXiv:1602.06023v5 [cs.CL] 26 Aug, 2016.

[11] Alexander M. Rush et. al., *"A Neural Attention Model for Abstractive Sentence Summarization"*, arXiv:1509.00685v2 [cs.CL], 3 Sep, 2015.