NLP based Machine Learning Approaches for Text Summarization

Rahul
Department of Computer Science and Engineering
Delhi Technological University
New Delhi, India
Rahul@dtu.ac.in

Surabhi Adhikari
Department of Computer Science and Engineering
Delhi Technological University
New Delhi, India
Surabhi_bt2k18@dtu.ac.in

Monika
Department of Computer Science and Engineering
Delhi Technological University
New Delhi, India
Monika.siwaliya@gmail.com

2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC) 978-1-7281-4889-2/20/$31.00 ©2020 IEEE 10.1109/ICCMC48092.2020.ICCMC-00099

***Abstract*— Due to the plethora of data available today, text summarization has become very essential to gain just the right amount of information from huge texts. We see long articles in news websites, blogs, customers' review websites, and so on. This review paper presents various approaches to generate summary of huge texts. Various papers have been studied for different methods that have been used so far for text summarization. Mostly, the methods described in this paper produce Abstractive (ABS) or Extractive (EXT) summaries of text documents. Query-based summarization techniques are also discussed. The paper mostly discusses about the structured based and semantic based approaches for summarization of the text documents. Various datasets were used to test the summaries produced by these models, such as the CNN corpus, DUC2000, single and multiple text documents etc. We have studied these methods and also the tendencies, achievements, past work and future scope of them in text summarization as well as other fields.**

***Keywords*—** *Natural Language Processing(NLP), Machine Learning(ML), Neural Network(NN), Abstractive(ABS) and Extractive(EXT) method*

I. INTRODUCTION

Today's world is centralized on computers and data. Data are our intangible thoughts and imagination. We are producers and consumers of data at the same time. Every little thing in our mundane lives are either a source or receiver of data. For, example when we drive there's data involved, the speed of the car, mileage, distance traveled, etc. Since 20th century, data has been a significant part of our lives, but these days we infer more from data. We store and access them through electronic and wireless systems.

Since the advent of the internet, there's an enormous amount of data available today. The Internet is a storehouse of data. Information on news, movies, education, medicine, health, nations, weather, geology, etc. is available on the internet. This could be statistical, numerical, mathematical or text data. Text data is more difficult to interpret due to larger amount of characters. Due to this gigantic amount of information, there must be a system in order to get only the essential parts of the information we access. Text summarization is a way of doing this.

Text summarization has been a topic of research and study since decades. Various models have been proposed and tested on different datasets to generate concise summaries. They are compared with different comparison scores. Text summarization can be EXT or ABS, single document or multi-document, and query-based or generic.

EXT text summarization is a way of generating summaries by using the same sentences as in the document. ABS is more general and focuses on key concepts of the document. Similarly, single document summarization techniques give summaries of the text of a single document, and multi-document generates summaries of multiple documents. Moreover, these days, there's a need for summarizing text based on queries. Query-based summarization models give summaries of the text based on a specific area as described by the query given by the user, whereas generic summaries are mostly ABS that focus on the general area of the text input.

Text summarization has been extensively used in various fields like science, medicine, law, engineering, etc. Researchers have focused on generating summaries of doctor's prescriptions, and that has been proved very useful to patients. Similarly, long news articles have been summarized and this way readers can gain a lot of information on several topics within a short span of time[1]. In this paper, we have discussed the various methods used in text summarization for past five years. The most common methods were found to be Machine Learning(ML), NNs, reinforcement learning, sequence to sequence modeling and fuzzy logic. Similarly, various optimization methods have been used to optimize the proposed objective function for the purpose of text summarization. We can see that various methods were tested on the same dataset, and their accuracy scores were found to be different. Moreover, we can also see that some researchers have combined the different methods and found out the summaries are more accurate than when a single method is used. When NLP processing has been used as a technique to summarize text documents, we see that python libraries such as scikit learn, nltk, spacy, and fastai has been used.

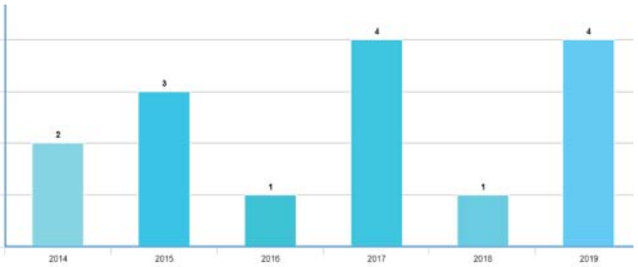

Fig. 1. Distribution of papers studied over the years

II. Related Work

Massimo Mauro, et al. have used a sentence extraction method to generate EXT summaries. In this method, the sentences are checked for their relevance and scored accordingly. Similar sentences were then clustered together to discover the most informative sentences and they were selected on the basis of sentence scores[2].

Sarda A.T. et al. have proposed NNs and Rhetorical Structure Theory (RST) to produce EXT summaries of articles. Features are compared and combined to produce a relevant sentence for the summary. The first step is to train the NN so that it learns to select the types of sentences that would be present in the summary[3]. After that, sentences in the document are selected and checked if they fit into the summary. After finding these sentences, they are fed to the rhetorical structure with the help of which linguistic relations are distinguished to form a better summary.

Gabriel Silva et al. have performed experiments on CNN-Corpus to generate EXT summaries of the documents. . . The sentences of the corpus after removing figures, videos, and tables were assigned with feature vectors for scoring. The dimensionality of feature vectors was reduced using the selection algorithms of WEKA viz. CFS Subset Evaluator, Information Gain Evaluator, and SVM Attribute. Naive Bayes was proven to be the best classifier among 5 classifiers tested using WEKA's platform[4].

Taeho Jo has presented an approach to summarize text by considering similarity between attributes and using KNN algorithm. A paragraph is given as input and it is divided into sentences. Words are represented as vectors. Each sentence is then classified as summary or remaining, and based on the similarity score between them and a human generated summary, it's included in the summary[5]. In this paper, feature is considered as an important factor in order to summarize a text paragraph. This approach can be used in multiple areas like medicine, law, engineering[5]. Similarity between only some features are taken into account, and similarity score is calculated using the KNN method, based on which a paragraph is summarized.

Mahsa Afsharizadeh, et al. presented a query-based approach for EXT text summarization. It selects the relevant sentences from the document and includes it in the summary. Eleven query-dependent appropriate features were chosen and used in the paper to find the important sentences[6]. For finding the useful sentences in the document, each sentence is assigned a score by checking on the linear function of its feature values.

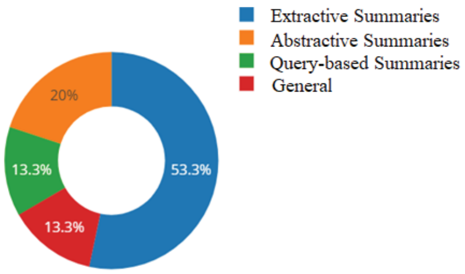

Fig 2. Types of Summaries Studied

For training and testing purpose, The DUC 2007 corpus was used. This method was able to get better average precision, average recall, and average F-score than earlier methods compared in the paper[7].

TABLE I. Summary of the Papers Analyzed

Name of Author	Year of Publication	Methods Used	Dataset used	Remarks
Kamalanathan Kandasamy et. Al[9]	2014	URL analysis, NLP, and Supervised, ML Techniques.	Tweets from users.	Combining three models gives more accuracy than single method used.
Mohamed Abdel Fattah [10]	2014	k—means clustering, differential evolutionary algorithm	data sets from DUC2001 and DUC2002	Results were compared with ROGUE 1 and ROGUE 2 scores. It showed better results than other text summarizers. Graph based algorithms for clustering could improve the result.
Mr. Sarda A.T. et al [3]	2015	NN, ML, RST	Input text documents	Due to inclusion of numerical data features, the output of this summarization method is better than other online text summarizers.
Gabriel Silva et al[4]	2015	STOM, CSF, Information Gain Evaluator and SVM attribute	news texts from CNN corpus	The results confirmed ML techniques improves the text summarization results.
K. Vimal Kumar, et al[15]	2015	Sentence ranking, Sentential semantic analysis and Sentence extraction	Hindi-text Single documents	This method can be applied for multi documents.
Rasim Alguliyev et al[12]	2016	Human Learning Optimization Algorithm	Takes a document as input.	This model tries to find balance between the topic coverage and sentence repetition in a summary, for generating concise summary.
Taeho Jo[5]	2017	k-nearest neighbor algorithm	Input paragraph	Similarity score is taken into account. Can be used in medicine, science and law.

Massimo Mauro et al.[2]	2017	sentence extraction, sentence scoring and sentence ordering using Python libraries such as scikit learn, nltk.	DUC2001 dataset	It was mostly useful for multi-text summaries when compared with ROUGE score
Aditya Jain, Divij Bhatia et. Al[14]	2017	Word vector embedding, NN	100 news articles from CNN news with their ABS summaries.	the proposed model outperformed other online text summarizers such as SPLITBRAIN, AutoSummarizer when the ROUGE scores were compared.
Nicholas Giamblanco et al[11]	2017	Newtonian method	SemEval from the University of Waikato	The method outperformed RAKE and Tf-IDF scores.
Mahsa Afsharizadeh, et al[7]	2018	Query based EXT summarization using TF-IDF, fuzzy logic and LSA.	DUC 2007 corpus	A fluent 250-word summary is generated and compared to ROGUE score.
J.N.Madhuri et al[8]	2019	Sentence ranking method using Python 3.6 and NLTK[8]	Takes text document as input	Those sentences whose rank is greater than 8 are included in the summary. The summarized text is then converted to audio form.
Milad Moradi et al[13]	2019	BERT model	Test on preliminary experiments of the development corpus	The size of summary would be good if it is 30% of the original text. This method worked best on biomedical texts.
Begum Mutlu et al[6]	2019	ML, fuzzy network	Input text documents and their features.	The resulting rule sets provide more coverage and generalization
Jesus M. Sanchez-Gomez et al[1]	2019	Multi Objective Artificial Bee Colony.	DUC 2002 datasets.	Efficiency problems were seen in this algorithm and they were solved using the asynchronous parallel MOABC.

III. Overview of Machine Learning Methods for Text Summarization

We have seen various researchers presenting ML methods for text summarization. Various supervised ML models such as Naïve Bayes, Random Forest, SVD have been used to generate EXT text summaries. Kamalanathan Kandasamy, et al. have used ML technique to classify spams on twitter. The data sets were also tested using SVM, but Naïve Bayes was found to be more accurate. For this experiment, 100 users were tested. Out of them, 60 were legitimate and 40 were spam. 98 users were categorized correctly[9].

We have also seen many researchers use deep learning techniques for EXT as well as ABS text summarization. Deep learning is an area of ML. Various NN techniques have been used. Similarly, reinforcement learning, Convolutional NN(CNN), RNN have also been applied to generate text summaries[10]. There's also a study of sequence-to-sequence models for text summarization these days. These methods are extension of ML. We describe some of the papers that use the above mentioned methods to generate summaries of text.

Nicholas Giamblanco et al presented a Newtonian method to generate key phrases. The author has used four steps for keyword and key phrase extraction. First the stop words are removed which is known as noise filtering, then each words is assigned a mass, and the relations between words is calculated which is also known as word attraction, and finally a key phrase is generated[11]. To test the accuracy of this algorithm, it was tested against RAKE and TF-IDF scores. The data used was from SemEval from the University of Waikato[12]. The author has also shown how Key-LUG has outperformed RAKE and TF-IDF. The metrics used for this purpose was recall, precision and F1 score[13].

Aditya Jain et. al paper proposed a binary approach for EXT text summarization. The document is broken into sentences and checked if it's relevant or irrelevant to the main theme of the document. A NN is then used and tested if the sentence is to be included or not. 100 news articles from CNN news along with their corresponding ABS summaries have been used as data sets for this model. Sentences in the news article and summary are compared and those with higher similarity score between them are chosen for EXT summary[14]. The author has used vector embedding to represent each word in a sentence as 100 dimensional vectors. For testing the performance of the proposed model, first 284 documents of the DUC 2002 dataset was used. It was seen that the proposed model was more accurate other online text summarizers[15].

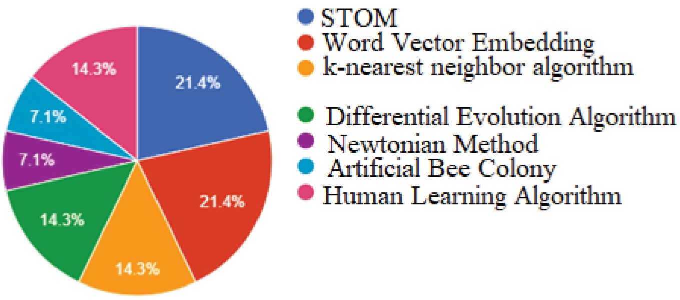

Fig 3. Different algorithms used

IV. CONCLUSION AND FUTURE WORK

We have seen that due to abundant availability of data, text summarization has a very vital role in saving user's time, as well as resources. Text summarization is indeed an important tool for today. We have seen the use of various algorithms and methods for this purpose. These methods, in individual and together give different types of summaries. Their accuracy score can be compared to find the better and more concise summaries. For this purpose, ROGUE score has been used more frequently. Similarly, in some cases TF_IDF scores have been used too.

The summaries generated using these methods are not always up to the mark. Sometimes, it's also irrelevant to the original document. Therefore, this topic is ongoing and people have done various works on this. There isn't any specific model that generates best summaries. So, for future, the models discussed can be modified for more accurate summaries. For e.g., we could use GAN's and transfer learning. For future, this can give a way to develop and enhance further ideas for text summarization.

ACKNOWLEDGMENT

We are extremely grateful to the Department of Computer Science and Engineering, Delhi Technological University for providing us the platform and all the necessary materials required to carry out this study.

REFERENCES

[1] C. Ordonez, Y. Zhang, and S. L. Johnsson, "Scalable machine learning computing a data summarization matrix with a parallel array DBMS," *Distrib. Parallel Databases*, vol. 37, no. 3, pp. 329–350, 2019, doi: 10.1007/s10619-018-7229-1.

[2] M. Mauro, L. Canini, S. Benini, N. Adami, A. Signoroni, and R. Leonardi, "A freeWeb API for single and multi-document summarization," *ACM Int. Conf. Proceeding Ser.*, vol. Part F1301, 2017, doi: 10.1145/3095713.3095738.

[3] A. T. Sarda and M. Kulkarni, "Text Summarization using Neural Networks and Rhetorical Structure Theory," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 6, pp. 49–52, 2015, doi: 10.17148/IJARCCE.2015.4612.

[4] G. Silva, R. Ferreira, S. J. Simske, L. Rafael Lins, M. Riss, and H. O. Cabral, "Automatic text document summarization based on machine learning," *DocEng 2015 - Proc. 2015 ACM Symp. Doc. Eng.*, pp. 191–194, 2015, doi: 10.1145/2682571.2797099.

[5] T. Jo, "K nearest neighbor for text summarization using feature similarity," *Proc. - 2017 Int. Conf. Commun. Control. Comput. Electron. Eng. ICCCCEE 2017*, pp. 1–5, 2017, doi: 10.1109/ICCCCEE.2017.7866705.

[6] B. Mutlu, E. A. Sezer, and M. A. Akcayol, "Multi-document extractive text summarization: A comparative assessment on features," *Knowledge-Based Syst.*, vol. 183, p. 104848, 2019, doi: 10.1016/j.knosys.2019.07.019.

[7] M. Afsharizadeh, H. Ebrahimpour-Komleh, and A. Bagheri, "Query-oriented text summarization using sentence extraction technique," *2018 4th Int. Conf. Web Res. ICWR 2018*, pp. 128–132, 2018, doi: 10.1109/ICWR.2018.8387248.

[8] J. N. Madhuri and R. Ganesh Kumar, "Extractive Text Summarization Using Sentence Ranking," *2019 Int. Conf. Data Sci. Commun. IconDSC 2019*, pp. 1–3, 2019, doi: 10.1109/IconDSC.2019.8817040.

[9] K. Kandasamy and P. Koroth, "An integrated approach to spam classification on Twitter using URL analysis, natural language processing and machine learning techniques," *2014 IEEE Students' Conf. Electr. Electron. Comput. Sci. SCEECS 2014*, pp. 1–5, 2014, doi: 10.1109/SCEECS.2014.6804508.

[10] M. A. Fattah, "A hybrid machine learning model for multi-document summarization," *Appl. Intell.*, vol. 40, no. 4, pp. 592–600, 2014, doi: 10.1007/s10489-013-0490-0.

[11] N. Giamblanco and P. Siddavaatam, "Keyword and Keyphrase Extraction using Newton's Law of Universal Gravitation," *Can. Conf. Electr. Comput. Eng.*, pp. 1–4, 2017, doi: 10.1109/CCECE.2017.7946724.

[12] R. Alguliyev, R. Aliguliyev, and N. Isazade, "A sentence selection model and HLO algorithm for extractive text summarization," *Appl. Inf. Commun. Technol. AICT 2016 - Conf. Proc.*, pp. 1–4, 2017, doi: 10.1109/ICAICT.2016.7991686.

[13] M. Moradi, G. Dorffner, and M. Samwald, "Deep contextualized embeddings for quantifying the informative content in biomedical text summarization," *Comput. Methods Programs Biomed.*, vol. 184, p. 105117, 2020, doi: 10.1016/j.cmpb.2019.105117.

[14] A. Jain, D. Bhatia, and M. K. Thakur, "Extractive Text Summarization Using Word Vector Embedding," *Proc. - 2017 Int. Conf. Mach. Learn. Data Sci. MLDS 2017*, vol. 2018-Janua, pp. 51–55, 2018, doi: 10.1109/MLDS.2017.12.

[15] J. K. Mandal, S. C. Satapathy, M. K. Sanyal, P. P. Sarkar, and A. Mukhopadhyay, "Information systems design and intelligent applications: Proceedings of second international conference India 2015, volume 1," *Adv. Intell. Syst. Comput.*, vol. 339, pp. 301–310, 2015, doi: 10.1007/978-81-322-2250-7.