2023 International Conference on Artificial Intelligence and its Applications
https://doi.org/10.59200/ICARTI.2023.020

Abstractive Text Summarisation using Recurrent Neural Networks at the Paragraph Level

Israel Christian Tchouya’a Ngoko and Boniface Kabaso
Departement of Information Technology
Cape Peninsula University of Technology
Cape Town, South Africa
tchouya’angokoi@cput.ac.za, kabasob@cput.ac.za

Abstract**—**Continuous production of information has been facilitated by the easy access to new technology. This has made it difficult for many users to find relevant information, which are sometimes buried deeply inside mass-produced content. Without the development of new tools and technology to make this data more accessible, it potential remain unexploited. Abstractive text summarisation aims to extract the key points of the document. Because text generating techniques are still in their early stage, it has received little attention in the past. Recently, the application of recurrent neural network models has made significant progress in abstractive sentence summarisation. Despite the improvement in results, these models still tend to produce grammatical errors. Unfortunately, attempts in abstractive document summarisation are still in their early phases, and evaluation outcomes on benchmark datasets are noticeably inferior to human summarisation. In this study we propose a data-driven for abstractive document. Each word generated in the summary use an attention-based technique depending on the input paragraph. According to experimental findings, our model generates higher-quality summaries, achieving ROUGE-1 score of 44.44, ROUGE-2 score of 22.50, and ROUGE-L score of 45.15 on the document understanding conference 2004 datasets.

Keywords—Abstractive text summarisation, recurrent neural network, DUC, machine learning, ROUGE scores.

I. Introduction

The goal of document summarisation is to produce a concise, coherent summary while preserving essential information. Document summarisation has been extensively studied as a practical solution to reduce information overload. There are two majors approaches of document summarization: extractive and abstractive. Extractive summarization generates the documents by selecting relevant sentences from the input document(s). Although this summarisation frequently results in coherent phrases and preserve the sense of the original document, sometimes there is unnecessary and illogical information present within the sentences.

On the Contrary, abstractive summarisation produces fluent summaries that are salient and accurate to the original document. It uses sophisticated approaches, such as meaning representation and content management, but improvement is still required [1]. Since Natural Language Generation (NLG) techniques are not well developed, fully abstractive approaches cannot always guarantee grammatically correct abstracts.

Recent research in neural networks has introduced a complete framework for NLG. This has been observed in various tasks like abstractive sentence summarization, machine translation and image captioning [2]. Unfortunately, the transition from sentence level abstractive summarization to document abstractive summarization remains a challenging task. The encoding and decoding process for extended amounts of texts still produces poor solutions [1]. Despite recent developments, methods for abstractive paragraph/document(s) summarisation have yet to produce compelling results.

In this research, we investigate how neural summarisation models can extract the essential content from a document. In the encoder-decoder architecture, we include attention mechanism. Furthermore, we study the challenges associated with handling and generating large sequences in sequence-to-sequence models, and propose a beam search technique with a reference mechanism for creating abstractive summaries. Under a unified framework, our proposed method can addresses the limitations related to saliency, redundancy, information accuracy, and fluency. The experiment is conducted on a large-scale corpus using human-generated summaries. The results of our experiment demonstrate that our method beats earlier neural abstractive summarisation models.

The structure of this paper is as follows: Section 2 discusses and overview of related works. Section 3 outlines the methodology of the study. Section 4, details the experiment, present the findings, and provide discussion. Finally, Section 5 offers the conclusion.

II. Related Works

The purpose of abstractive summarisation is to generate a summary from a given input. It uses a word level attention mechanism during each decoding step to identify the significance of words in relation to the target words. This process comes with several challenges such as paraphrase, simplification, and fusion. Previous studies have been limited to one or a few of the issues [3],[4],[5], and [6].

In terms of neural network models, success has been achieved in sentence abstractive summarization. [2] use a vast corpus of news documents and headlines to train a neural attention model. The combination of this probabilistic model with a generation method led to the creation of accurate abstractive text summaries. [7] extend their work framework since the grammaticality of summaries continues to be an issue in generating summary at the paragraph level. On the Document Understanding Conference (DUC) competition of producing headlines level summaries for documents, neural abstractive sentence models achieved state-of-the-art results. Recent research looked into neural abstractive models on the document summarisation challenge. Cheng and Lapata [8] create a summary using a word from the incoming document. Nallapati et al. [9] extends the phrase summarisation model by experimenting with tiered attention mechanism and a restricted vocabulary during the decoding phase. However, these models only look at a few aspects of the document summarisation process. Cheng and Lapata [8], firstly explore the uniqueness of summarisation and propose a distraction-based attentional model. However, these neural abstractive summarizing models are not competitive with traditional abstractive summarisation approaches, and many issues remain unresolved.

See et al. [10] described an architecture that improved the traditional sequence-to-sequence attentional model using a hybrid pointer-generator with coverage to keep account of sentence summarisation. It assisted in reducing the replication of incorrect sentences and avoiding redundancy. However, at a higher level of abstraction, the question remains unsolved.

Hou et al. [11] introduced an attention mechanism to prevent information redundancy and employed a sub-word strategy to handle unusual or uncommon. The combined attention mechanism outperformed previous models and produced the most impressive results in single text summarisation. Nevertheless, this model is still dealing with the challenging task of processing various documents.

Rossiello et al. [12] demonstrated the use of prior information to improve neural abstractive text summarisation. The model employed handmade linguistic features to examine the representation of word connections within a document. However, the process of generating abstractive text summaries is still a work in progress.

Numerous studies have shown that the use of abstractive text summarisation using Recurrent Neural Network (RNN) produces good summaries [2, 7, 10, 11]. Moreover, as summaries are produced on a sentence by sentence basic, the total number of sentences in the summary is kept. However, the quality of these text summaries is still much behind human summaries. According to Rush et al. [2], there has been no research on RNN-based abstractive text summarisation at the paragraph level.

III. Methodology

A. *Overview*

This section describes our approach. We employed an encoder-decoder structure, which is commonly used in machine translation [13, 14]. Our model includes the development of relevant hypotheses, data analysis, and result interpretation.

B. *Formulated Hypothesis*

H1. Abstractive text summarization using RNN at the paragraph level does not produce different number sentences and does not improve ROUGE scores compared to sentence level abstractive text summarisation.

H2. Abstractive text summarization using RNN at the paragraph level produces different number sentences and improves ROUGE scores compared to sentence level abstractive text summarisation.

C. *Abstractive text summarisation using RNN at the paragraph level*

This section explores the model (conceptual framework) of abstractive text summarisation using RNN with an attention mechanism based encoder-decoder from the input text document to the output summaries. This summarisation procedure employs unsupervised learning to train a machine to analyze articles from a dataset, after which the final summary and ROUGE scores are generated. This method is repeated until the experiment meets the desired level of satisfaction. The steps of the experiment are illustrated in Fig. 1.

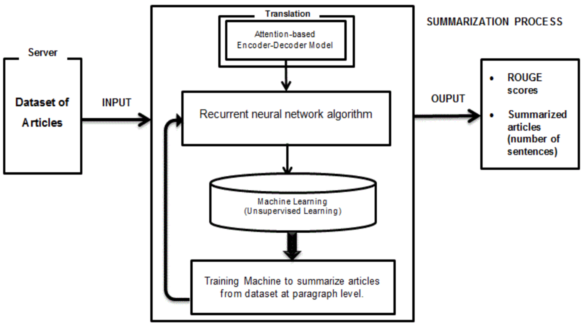
Server
Dataset of Articles
INPUT
Translation
Attention-based Encoder-Decoder Model
Recurrent neural network algorithm
Machine Learning (Unsupervised Learning)
Training Machine to summarize articles from dataset at paragraph level.
SUMMARIZATION PROCESS
OUPUT
ROUGE scores
Summarized articles (number of sentences)

Fig 1: Conceprual framework.

D. *Machine learning*

Through Machine learning (ML) algorithms, computers can learn and accomplish complete tasks via intelligent software [15]. The goal of ML is to develop methods and models that enable computers to learn without being explicitly programmed. In general ML is divided into four types: supervised, semi-supervised, unsupervised, and reinforcement learning [16]. Unsupervised learning enables computers in the model represented in Fig. 1 to generate output by referencing original input [17].

E. *Recurrent neural network*

RNN is an artificial neural network that connects nodes in a straight graph. It is one of the most often used unsupervised learning algorithms [18]. It assesses the likelihood of sentences occurring in input paragraphs to assign scores, considering both grammatical and semantic accuracy. RNN also works with sequential data to make calculations. For NLP tasks, RNN has become the industry standard. To prevent large RNNs overfitting, researchers often use early stopping, small models, and one that are not fully specified [19]. RNN performs well in summarisation, data modeling, and statistical analytic jobs.

For tasks such as article compression and sequence prediction, RNN uses encoder-decoder models with variable inputs and outputs. The convolutional, bag-of-words, attention-based encoder-decoder models are the most used [2]. Using attention-based RNN encoder-decoder, this work explores how phrases are encoded and decoded in continuous space while maintaining semantics and syntactic details. An illustrative example of RNN encoder-decoder is shown in Fig. 2.

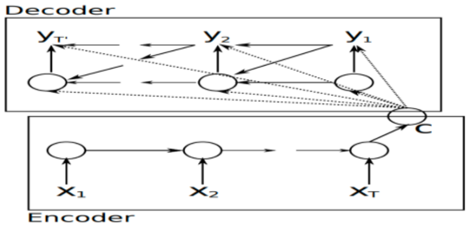

Fig. 2: RNN encoder-decoder

F. *Attention-based model*

With the help of the attention-based model, the machine can scan through and hold onto information from the original sentence. It then predicts the target word considering all previous generated target words along with context vectors at different source positions. This process converts the complete source sentence into a fixed-length vector connected to specific source positions. The attention mechanism model predicts the translated words by considering relevant information from the source sentence and previously generated target words. Moreover, the attention model connected the encoder and decoder allowing the decoder to receive information from each hidden state of the encoder [20]. The translation issue, which necessitates reading whole phrases and condenses all information into a fixed-length vector, has been solved by attention mechanism [13, 20]. A sentence with multiple words representing hundreds of words will result in an inadequate translation or loss of information. The fundamental idea behind attention is that each step of the decoding procedure is closely tied to specific encoder components. This method was developed to enhance the encoder-decoder RNN's performance in summarization and machine translation.

Recently, attention-based methods [13] has been linked to several articles on machine translation models [20, 21]. These models primarily use RNNs to generate text documents. An example of attention is shown in Fig.3 below:

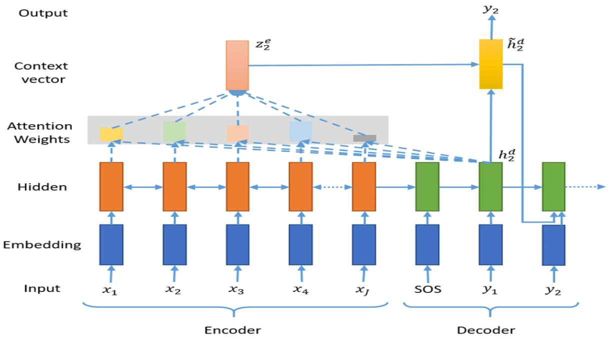

Fig. 3:Attention-based mechanism.

G. *Dataset*

The DUC 2004 huge corpus, which has been used in neural document summarization tasks, incorporates the conventional paragraph summarization evaluation [22, 23]. For this experiment, the dataset comprised fifty articles from the Press Wire services and the New York Times each coupled with a set of four human reference summaries extracted from the stories on the DUC website. An attention-based model fed the tokens of the article into the encoder (RNN layer). Attentional is calculated as a weight sum of a set of encoder hidden states that is dependent of the current decoder hidden state [13]. On each time interval, the decoder (RNN layer) obtains the embedding (Doc2vec) of the preceding word (paragraph). Subsequently, it guided on where to focus based on the attention distribution to produce the next word. To manage out of vocabulary terms and accurate mistakes, a pointer generator is used. At every time interval, a probability is computed from the context vector, the decoder input, and the state. To ensure that only recall evaluation is unaffected by length, the output length is limited to 100 characters, and no advantage is offered for short summaries. NLP provides metrics to assess the quality of machine summaries such ROUGE, BLUE, and GLUE. However, ROUGE is the most popular and used metric to evaluate generated summaries [24]. The evaluation of this model using the pyrouge package provides precision, recall, and an F-score for metrics such as ROUGE-1, ROUGE-2, and ROUGE-L.

H. *Research process and hypotheses*

The experimental diagram in Fig. 4 outlined the modification of input factors (article and its length) that produce output variables (summarised articles and ROUGE scores).

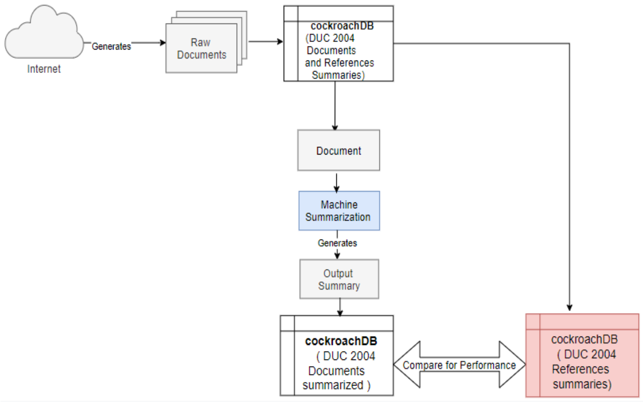
Internet
Generates
Raw Documents
cockroachDB (DUC 2004 Documents and References Summaries)
Document
Machine Summarization
Generates
Output Summary
cockroachDB (DUC 2004 Documents summarized)
Compare for Performance
cockroachDB (DUC 2004 References summaries)

Fig. 4: Experimental diagram

Fig. 4 shows the input documents from the online DUC-2004 database. From each document (article) stored in a Cockroach database, a machine generated summary is produced. Subsequently, their performance is evaluated by comparing them to the alternative set of four reference summaries provided by the DUC 2004 online database using ROUGE scores.

I. *Implementation*

Our implementation is based on the Torch numerical framework (http://torch.ch/), and we will provide public access public to the code and data pipeline. The fact that training is done on a Graphic Process Unit (GPU) is noteworthy. The parameters are modifiable settings for the algorithm. The 256-dimensional RNN model has pre-trained and trained word embedding that is driven by data. 32128 vocabulary tokens totaled the used vocabulary size. The pre-trained word embedding are generated using T5Tokenizer. The validation set loss was used to account for premature stopping. The size five beam search approach is used to generate the summaries while messaging. The total number of tokens is divided by the loss of the sequence. Using the final hidden state of the encoder as input, a further layer determines the initial hidden state of the decoder.

IV. Experiments and Results

The experiment was conducted using the DUC-2004 evaluation dataset. We employed 50 papers for each summary, with documents ranging from a maximum of 655 sentences to a minimum of 153 sentences, each accompanied by four references. The algorithm's execution time for each document shown in Fig. 5 ranges from 46.27 to 87.71 seconds. This has resulted in both the worst and the optimal time for the time executing of the algorithm during computation.

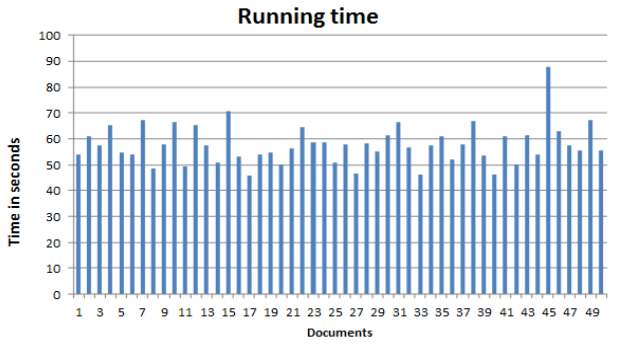

Fig. 5: Execution time of documents in seconds.

The performance metrics for ROUGE-1, ROUGE-2, and ROUGE-L of the proposed approach are shown Figs 6,7 and 8 below. In Fig. 6 below, the proposed algorithm produces summaries with a ROUGE-1 F-score of 44.

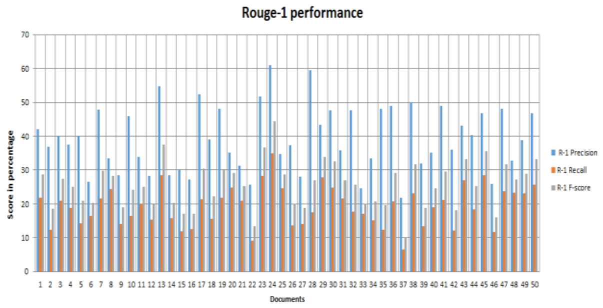

Fig. 6: ROUGE-1 metrics performance.

The algorithm produces summaries on Fig. 7 below reaching a ROUGE-2 F-score 22.50.

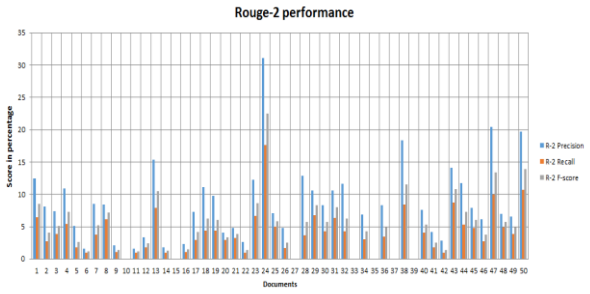

Fig. 7: ROUGE-2 metrics performance.

The algorithm produce summaries on Fig. 7 below reaching a ROUGE-L F-score of 44.15.

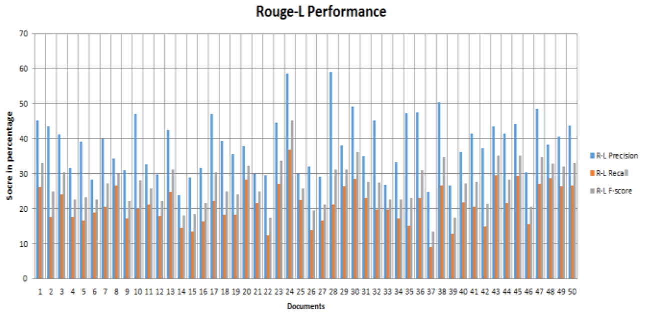

Fig. 8: ROUGE-L metrics performance.

The summaries generated in this experiment typically consist of two, three, or four sentences, with a maximum of seventy-four words. In contrast, the alternative reference summaries provide by the DUC 2004 dataset consist of at least one sentence more than the experiment's results. [2] uses Gigaword to demonstrate that both reference summaries and abstractive text summaries produced using RNN at the sentence level have the same number of sentences. Fig 9 compares the number of sentences created per document at the paragraph level (experiment) with alternative reference summaries that are provided at the sentence level.

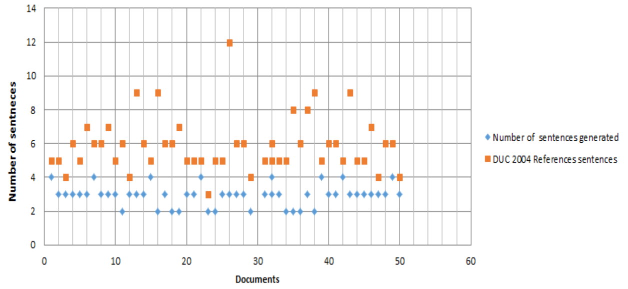

Fig. 9: Compare number of sentences produced between the algorithm and the alternative reference summaries provided by DUC 2004.

The ROUGE scores of the algorithms are illustrated in Fig. 10. The highest ROUGE-1, ROUGE-2, and ROUGE-L scores were obtained using RNN at the paragraph level for DUC 2004 datasets, with scores of 44.44, 22.50, and 45.15, respectively.

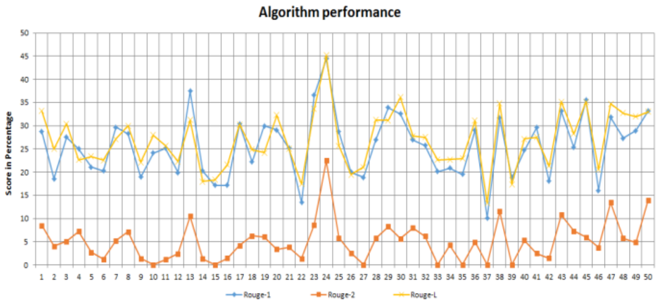

Figure 10: Algorithm performance.

Table 1 provides a comparison of the encoder-decoder and training procedures used in the abstractive summarisers described in the related studies (Section 2) with the experimental approach details in 3.H.

TABLE 1: Comparison of the techniques used in the abstractive summarisers.

Model (M)		Encoder	Decoder	Training
M1	Abstractive sentence summarisation [2]	Attention-based and Bag-of-word encoder.	Neural network-based language model	Stochastic gradient descent (SGD)
M2	RAS-LSTM and RASElman [20].	convolutional neural networks + attention	Elman RNN or Long short-term memory (LSTM)	SGD
M3	Hierarchical Attentive RNNs [9]	feature-rich bidirectional-GRU + hierarchical attention	GRU (Gate Recurrent unit) + The large vocabulary trick + pointer switch	SGD
M4	Pointer-Generator Networks [25]	bidirectional LSTM + attention	LSTM + pointer switch + coverage mechanism	SGD
M5	Neural Intra-attention Model [26]	bidirectional LSTM + intra-attention	LSTM + pointer switch + intra-attention	SGD + REINFORCE
M6	Abstractive paragraph summarisation (Experiment)	Attention-based encoder	Neural network-based language model	SGD

Based on the research findings presented in table 1 above, model M6 has reduced complexity in terms of encoder-decoder and data training. The outcomes from the abstractive text summarisation experiment employing RNN at the paragraph level support the hypothesis outlined in 3.B, as illustrated in table 2 below.

TABLE 2: Rouge (R) F-scores of the abstractive summarizers on DUC 2004, Gigaworld and CNN/DailyMail Datasets.

Model	DUC 2004			Gigaworld			CNN/DailyMail		
	R1	*R2*	*R3*	*R1*	*R2*	*R3*	*R1*	*R2*	*R3*
M1	28.2	8.4	23.8	31.0	12.7	28.3	-	-	-
M2	-	-	-	33.8	15.9	31.2	-	-	-
M3	-	-	-	35.3	16.7	32.6	-	-	-
M4	-	-	-	-	-	-	39.5	17.3	36.4
M5	-	-	-	-	-	-	39.8	15.8	36.9
M6	44.4	22.5	45.2	-	-	-	-	-	-

The experiment's findings show that abstractive text summarisation using RNN at the paragraph level (M6) achieves higher scores on ROUGE-1, ROUGE-2, and ROUGE-L when utilizing different datasets compared to abstractive text summarisation using RNN at the sentence level (M1) and other summarizers. The experiment outperforms M1 in terms of sentences count using datasets from DUC 2004 (refer to table 2 and Fig. 9). The quality of the paragraph improves since there is a reduced number of sentences in the summary, leading to decreased redundancy.

Additionally, it surpassed the performance of all papers presented at DUC 2004, Gigaword, and CNN/DailyMail, and establishing a new state-of-art-art with the least computation time (see table 2). By converting lengthy texts words into a more manageable format for the summariser to analyze, this experiment takes use of the drawbacks of existing abstractive summarisation tools. This leads to accelerated data training due to the reduced model complexity and the exclusion of unnecessary sentence(s)/paragraph(s) from the summary.

This study aimed to evaluate the efficiency of abstractive text summary at the paragraph to improve the quality and enhance the quality of abstractive summarization. To assess whether our approach outperforms abstractive text summarization at the sentence level, the t-test results are displayed in table 3.

TABLE 3: T-test of abstractive text summarisation with DUC 2004 ROUGE scores

Abstractive text summarisation ROUGE Scores		
	Sentence level (M1)	***Paragraph level (M6)***
Mean(M)	20.12333333	37.36333333
Sample size(n)	3	3
Standard deviation (StDev)	10.4021	12.87692
Variance(S)	108.2036333	165.8150333
Degree of freedom (df)	2	2
Critical value of t, df=4	2.776	
sample Mean difference (M1-M2)	-17.24	
Hypothesized diffence (μ1- μ2)	0	
Pooled sample variance	137.0093333	
Standard error for difference (Sm1-m2)	57.50233174	
Obtained value of t	-0.299813929	
p-value or t-test or P	0.145579619	

Our method significantly demonstrate an improvement over sentence level abstractive summarisation, as shown by the statistical t-test results (M = 37.36, S = 165.81), t(4) = 2.776, p-value = 0.15. P-value>0.05 indicates that, when compared to abstractive text summarization at the sentence level, RNN at the paragraph level produces a different number of sentences and improves ROUGE scores.

V. CONCLUSION

In this study, we presented abstractive text summarization using RNN at the paragraph level. Integrating our model with a tailored algorithm resulted in accurate abstractive summarization. We then use ROUGE scores to compare the effectiveness of abstractive text summarization using both paragraph level and sentence level. The experimental results revealed that paragraph-level abstractive text summarisation using RNN produces higher ROUGE scores and fewer sentences compared to sentence-level abstractive text summarisation using RNNs. This study addresses challenges associated with abstractive document summarisation for a single document, as well as the need for salient content from the original text. To generate a multi-sentence summary, we use the beam search technique. The experiment uses the DUC 2004 dataset, and we also plan to expand the scope of the research. Additionally, we intend to gather more data to investigate abstractive multi-document summarisation using RNN.

REFERENCES

[1] X. Chen, M. Li, X. Gao, and X. Zhang, “Towards improving faithfulness in abstractive summarization,” Advances in Neural Information Processing Systems, 35, pp.24516-24528, arXiv [cs.CL], 2022.

[2] A. M. Rush, S. Chopra, and J. Weston, “A neural attention model for abstractive sentence summarization,” arXiv (Cornell University), Sep. 2015.

[3] K. Woodsend and M. Lapata, “Multiple aspect summarization using integer linear programming,” Empirical Methods in Natural Language Processing, pp. 233–243, Jul. 2012.

[4] K. Thadani and K. McKeown, “Supervised Sentence Fusion with Single-Stage Inference,” International Joint Conference on Natural Language Processing, pp. 1410–1418, Oct. 2013.

[5] D. Pighin, M. Cornolti, E. Alfonseca, and K. Filippova, “Modelling Events through Memory-based, Open-IE Patterns for Abstractive Summarization,” Jan. 2014.

[6] R. Sun, Y. Zhang, M. Zhang, and D. Ji, “Event-Driven Headline Generation,” . in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp. 462–472, Jan. 2015.

[7] S. Chopra, M. Auli, and G. Rushton, “Abstractive Sentence Summarization with Attentive Recurrent Neural Networks,” In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 93–98, Jan. 2016.

[8] J. Cheng and M. Lapata, “Neural Summarization by Extracting Sentences and Words,” Jan. 2016, doi: 10.18653/v1/p16-1046.

[9] R. Nallapati, B. Zhou, C. N. D. Santos, C. Gulcehre, and B. Xiang, “Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond,” Jan. 2016.

[10] A. See, P. J. Liu, and C. D. Manning, “Get To The Point: Summarization with Pointer-Generator Networks,” T, Jan. 2017.

[11] L. Hou, P. Hu, and C. Bei, “Abstractive Document Summarization via Neural Model with Joint Attention,” in Lecture Notes in Computer Science, 2018, pp. 329–338.

[12] G. Rossiello, P. Basile, G. Semeraro, M.D. Ciano, and G. Grasso, “Improving neural abstractive text summarization with prior knowledge”, URANIA, 16, 2016.

[13] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” arXiv [cs.CL], 2014.

[14] Y. Gao, C. Herold, Z. Yang, and H. Ney, “Is encoder-decoder redundant for neural machine translation?,” arXiv [cs.CL], 2022

[15] M. S. Ibrahim, W. Dong, and Q. Yang, “Machine learning driven smart electric power systems: Current trends and new perspectives,” Appl. Energy, vol. 272, no. 115237, p. 115237, 2020.

[16] K.. Aggarwal, Mijwil, M.M., A.H. Al-Mistarehi, S. Alomari, M. Gök, Alaabdin, A.M.Z. and S.H. Abdulrhman, “Has the future started? The current growth of artificial intelligence, machine learning, and deep learning,” Iraqi Journal for Computer Science and Mathematics , 3(1), pp.115-123, 2022.

[17] M. Gambhir and V. Gupta, “Recent automatic text summarization techniques: a survey,” Artif. Intell. Rev., vol. 47, no. 1, pp. 1–66, 2017.

[18] Y. Su and C.-C. J. Kuo, “Recurrent neural networks and their memory behavior: A survey,” APSIPA Trans. Signal Inf. Process., vol. 11, no. 1, 2022.

[19] W. Zaremba, I. Sutskever, and O. Vinyals, “Recurrent neural network regularization,” 2014.

[20] M. Melese, “Attention-based Neural Machine Translation from English-Wolaytta (Doctoral dissertation, St. Mary’s University),” 2023.

[21] A. V. Potnis, R. C. Shinde, and S. S. Durbha, “Towards natural language question answering over earth observation linked data using attention-based neural machine translation,” in IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium, 2020.

[22] A. Alomari, N. Idris, A. Q. M. Sabri, and I. Alsmadi, “Deep reinforcement and transfer learning for abstractive text summarization: A review,” Comput. Speech Lang., vol. 71, no. 101276, p. 101276, 2022.

[23] D. Morozovskii and S. Ramanna, “Rare words in text summarization,” Natural Language Processing Journal, no. 100014, p. 100014, 2023.

[24] P. J. A. Colombo, C. Clavel, and P. Piantanida, “InfoLM: A New Metric to Evaluate Summarization & Data2Text Generation,” Proc. Conf. AAAI Artif. Intell., vol. 36, no. 10, pp. 10554–10562, 2022.

[25] I. Sutskever, O. Vinyals, and Q.V. Le, “Sequence to sequence learning with neural networks,” Advances in neural information processing systems, 27, pp.3104-3112, 2014.

[26] R. Paulus, C. Xiong, and R. Socher, “A deep reinforced model for abstractive summarization,” arXiv [cs.CL], 2017.