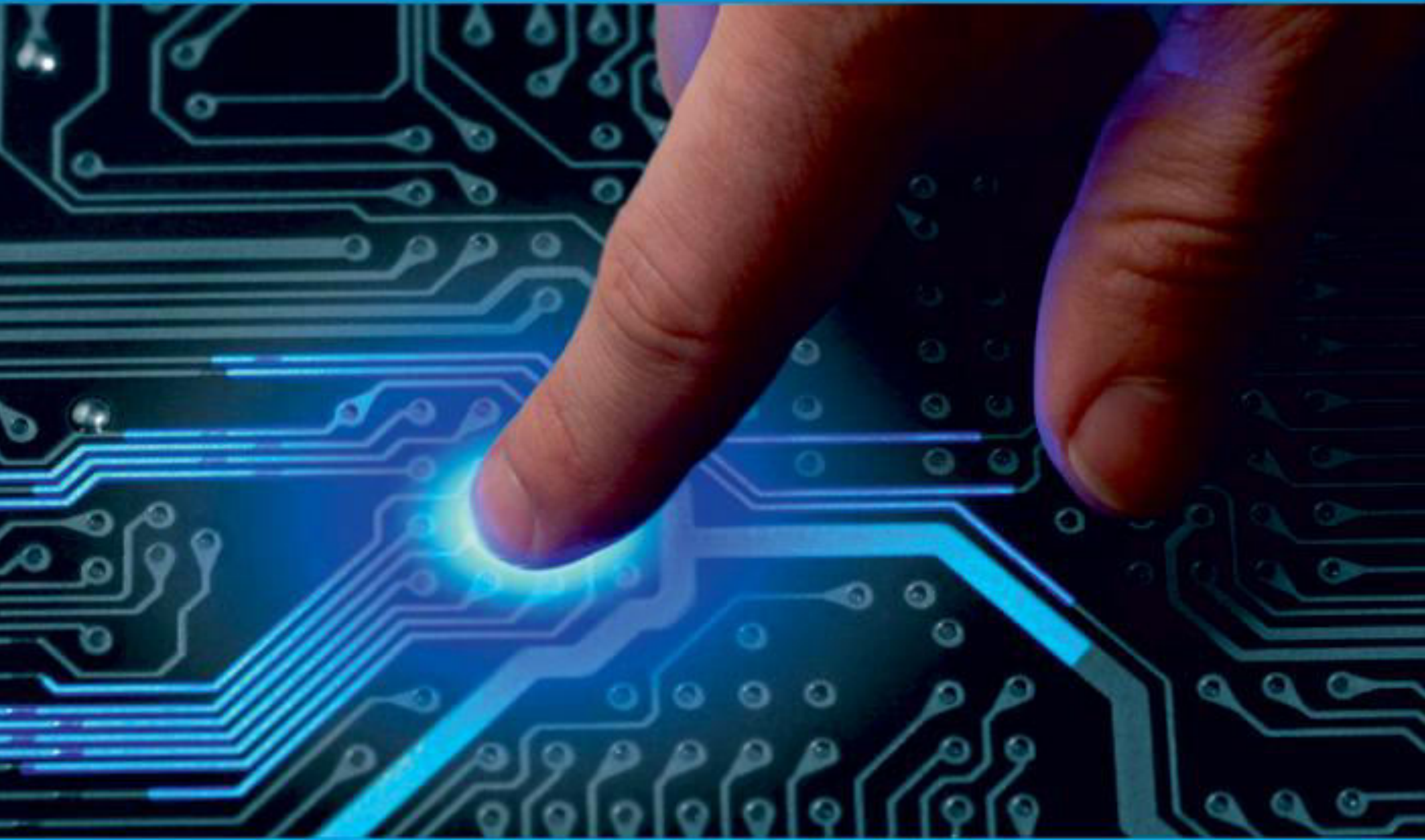
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 6, June 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

☐ 9940 572 462 ☎ 6381 907 438 ✉ ijircce@gmail.com @ www.ijircce.com

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542 |

|| Volume 9, Issue 6, June 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0906174 |

Abstractive Summarization Using Sequence-to-Sequence Models

Vaishnavi Madhavaram [1], D Koteswara Rao [2]

Department of Computer Science & Engineering, MGIT, Hyderabad, Telangana, India[1, 2]

ABSTRACT: The task of creating a short and concise summary of the text document is called as text summarization. This provides the core content of the long text in a condensed form. It can be done or divided into Extractive and Abstractive based Summarization. Extractive method includes copying words directly from the source document. Abstractive methods include rephrasing the sentences in the corpus and generate a summary in a more human-like way. This paper mainly focuses on the Abstractive Text Summarization. In this paper, a sequence-to-sequence (Seq2Seq) model is used for ATS, which follows an Encoder-Decoder Architecture. To improve the performance of Seq2Seq model, an Attention Mechanism, Pointer-Generator and Coverage mechanism are used. It helps to read longer text, to overcome the Out-Of-Vocabulary and to reduce repetitiveness, respectively.

KEYWORDS:AbstractiveText Summarization, Sequence-to-Sequence model, Bi-LSTM, Attention, Pointer-Generator, Coverage, ROUGE.

I. INTRODUCTION

In today's world, everyone is surrounded by data. Data servers have connected the entire planet like a spider web. The Web can be seen as the biggest resource of textual contents like books, journals, and newspapers etc. This huge amount of information is generally left unused if it is not made available to the users in an efficient way. This can be done by decreasing the size of this enormous data and provide the user with a short summary that maintains both the meaning and semantics and grammatical correctness of the textual content of the original document. Text Summarization is an attempt to decrease the size of a corpus while keeping its valuable content.

The process of creating short but relevant summaries from a long text document is called as Text Summarization. It provides the most important and required content of the long text document in a condensed form. The available methods of text summarization are either Extractive Text Summarization (ETS) and Abstractive Text Summarization (ATS). ETS [2] involves extracting sentences or phrases of the original text based on weights assigned to important words. Whereas ATS [3] includes training neural networks to understand the whole context of the source text and generates a summary by rephrasing sentences based on that understanding. Abstractive Text Summarization is generally obtained using the recent Sequence-to-Sequence Models. Seq2Seq model is a neural network that follows an Encoder-Decoder Architecture. The first part i.e. an encoder reads the text from the original document and converts it into a context vector. The Decoder then produces the summary.

The traditional Seq2Seq model has many challenges. It may not be applicable for longer texts, out-of-vocabulary words cannot be generated and displayed as unknown tokens and repeated phrases in the summary. To overcome the drawbacks of the Seq2Seq Model, an attention mechanism, pointer network and coverage is used. This help the Seq2Seq model to cover longer sentences, to produce the Out-Of-Vocabulary (OOV) words and reduce the repetitive summary sentences respectively. This improved model can b called as the Pointer-Generator model, which decides whether the next word in the summary should be either selected from the vocabulary based on understanding or copied from the source text itself.

In this paper, the CNN/Daily Mail dataset [9] is used. It contains online news articles paired with multiple key sentences that can form the summaries. Most of the ATS works focused on single line summary generation, we focus on generating a multi-sentence summary. Instead of using the anonymized dataset as used in Nallapati et al. [1], AST uses non-anonymized dataset with minimum pre-processing done.

II. LITERATURE SURVEY

Sequence-to-Sequence based model [2] has become the mainstream of Abstractive Text Summarization and Translation problems. Jingjing Chen and Fucheng You [4] used a Seq2Seq model to generate the summary. Both the encoder and decoder used a single Long Short-Term Memory (LSTM) layer. Then they calculated the semantic similarity between source and generated summary, using Jaccard similarity. Though the model produced better semantic similarity and readability, it failed to identify the OOV words.

In [5], Z Hao and others put forward a feature enhanced Seq2Seq model for summarization task. This model can identify the salient features in a better way and store such global features to generate higher-quality summaries. But this model also could not handle the OOV problem, and repetitiveness as the baseline model.

P Hanunggul and Suyanto in their work in [6], used attention mechanism with Seq2Seq model and compared the global attention, and local attention models using ROUGE scores. Global attention produces better ROUGE-1, local attention produces higher ROUGE-2. Out-of-Vocabulary words cannot be identified and thus resulted in low ROUGE scores.

Abigail See, Liu and Manning proposed the Pointer-Generator Model [12] which will decide whether the next word in the summary should be either chosen from vocabulary or copied from the text. W Nie used an improved pointer-generator network in [13]. By using a multi-hop attention mechanism, they improved the way the model understood the words and context, and its ability to generate words. On CNN/Daily Mail dataset, the model generated meaningful and fluent summaries with higher ROUGE scores.

The model proposed by Abigail [13] has overcome the main drawbacks of a Seq2Seq model and thus in this paper, a Seq2Seq Model with Attention, Pointer network and coverage is used.

III. METHODOLOGY

Some pre-processing techniques applied to the CNN/Daily Mail dataset include tokenization, changing to lower case, adding missing periods, and start or stop tags to each sentence in the summary, and binarization. We used a Seq2Seq model as the baseline model with an Encoder containing a Bidirectional LSTM (Bi-LSTM) layer and the decoder containing an LSTM layer. Both the encoder layer and decoder layer share a common embedding layer. The baseline model is then added with an attention layer, pointer network, and a coverage mechanism. Figure 1 shows the architecture of the used Pointer-Generator model proposed in [13].

A. ATTENTION MECHANISM

For longer texts, it is very difficult to include all the contents of the source into a single context vector, so by using an multi-hop attention mechanism proposed by Bahdanau et al. (2015) [11], multiple vectors can be encoded from the source or the original text. This helps the baseline model to cope with longer sentences. The attention distribution a^t is calculated as proposed by Bahdanau in [11]:

$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + b_{attn}) \qquad (1)$$

$$a^t = softmax(e^t) \qquad (2)$$

$$h_t^* = \sum_i a_i^t h_i \qquad (3)$$

$$P_{vocab} = softmax(V'(\, V[\, s_t,\, h_t^*] + b\,) + \, b') \qquad (4)$$

In the Eq. 1, $h_i$is the hidden state, s_t is the decoder state, and v, W_h, W_s and $b_{attn}$are learnable parameters. For every decoder timestep t, attention distribution a^t, Eq. 2 is the used to get the context vector h_t^* using the Eq. 3 and then the vocabulary distribution is calculated as in Eq. 4, where V', V, b and b' are learnable parameters. P_{vocab} is a probability distribution of words in the generated vocabulary.

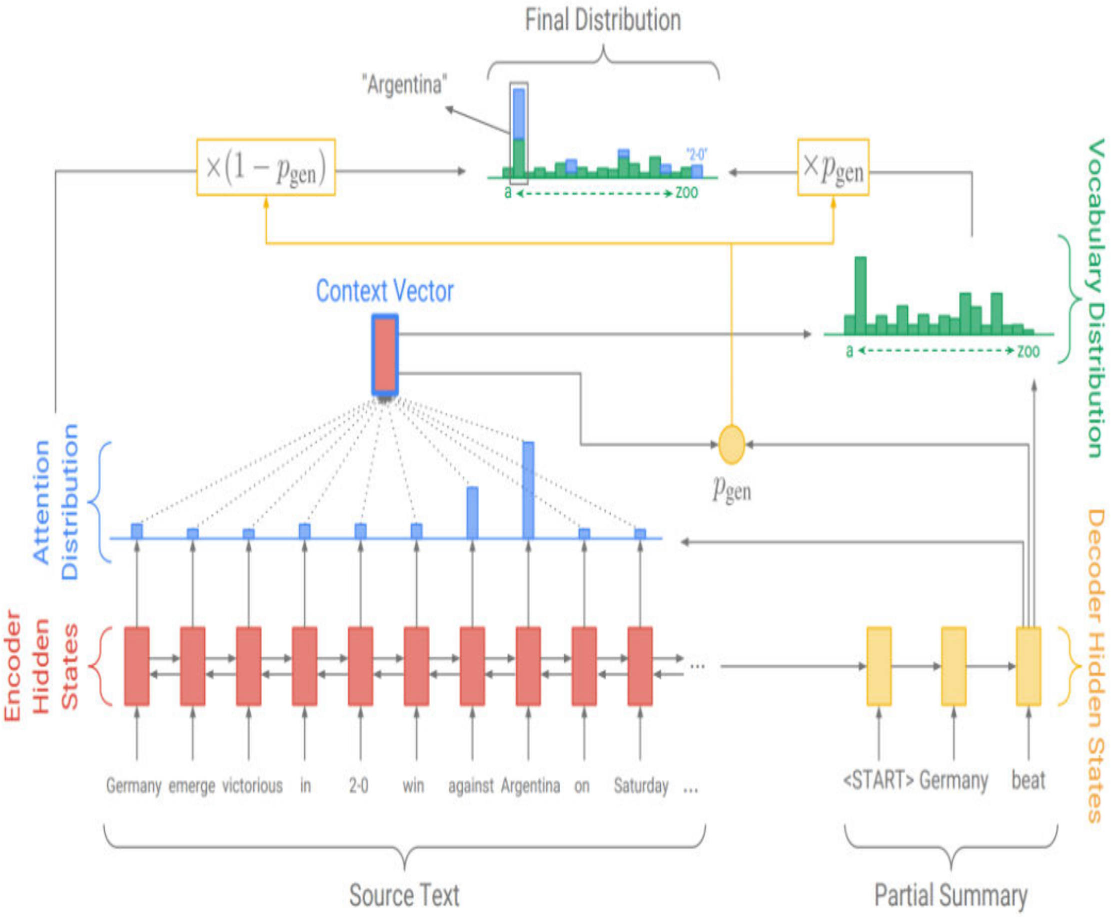


Fig 1: Architecture of Pointer-Generator Model

B. POINTER-GENERATOR

A traditional Seq2Seq model generates factual details inaccurately for rare or OOV words such as names, dates, and places. To overcome this OOV problem, a pointer network is added to the baseline model with Attention mechanism, which can be termed as Pointer-Generator model [12]. It decides whether the next word in the summary should be either selected from the vocabulary based on understanding or copied from the source text itself.

In this model, for every decoder timestep t, after calculating the attention distribution and context vector, the generation probability is also calculated using context vector h_t^*, decoder state s_t and decoder input x_t using Eq. 5.

$$p_{gen} = \sigma(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr}) \qquad (5)$$

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_i^t \qquad (6)$$

where vectors w_{h^*}, w_s, w_x and b_{ptr} are learnable parameters. The final distribution is calculated using Eq. 6. It points to the OOV word in the original document in the case when $P_{vocab}(w)$ is zero, else it points to a word that is not present in the original document when $\sum_{i:w_i=w} a_i^t$ is zero.

C. COVERAGE MECHANISM

The baseline model or Pointer-Generator model fail to address the problem of repetition, especially in the case of multi-sentence summary generation. This problem can be addressed using a coverage mechanism as proposed in [13]. When the coverage mechanism is added to the pointer-generator model, a coverage vector c_t is calculated at every decoder timestep using Eq. 7. It is the summation of all the previous timesteps' attention distributions.

$$c^t = \sum_{t'=0}^{t-1} a^{t'} \qquad (7)$$

The coverage vector is added to the attention mechanism, so that the generation of the next word in the summary is done by considering the previous decisions or words generated in the summary. Thus, this prevents the model from frequently pointing to the same areas and producing the same words and phrases again and again by defining a coverage loss.

In this paper, the combination of three mechanisms can produce higher quality summaries. The encoder reads from the original document word-by-word and then produces the encoder hidden states. The previous generated or pointed word of the summary is given as the input to the decoder, and this is used to update the next decoder hidden state. The attention distribution is calculated using the encoder hidden states and decoder hidden states. The attention distribution helps the network to choose the next word in the summary by showing the most relevant words. The attention distribution is used to calculate the context vector which indicates the text read by the encoder up to that timestep. The context vector and decoder hidden state help in getting vocabulary. Also, the generation probability is calculated which is used to combine the vocabulary distribution (generating) and attention distribution (pointing) into the final distribution P.

IV. EXPERIMENTAL RESULTS

The baseline model is Seq2Seq model with Attention Mechanism. This can be extended to Pointer-Generator and coverage model by setting the respective flags while training. The size of the vocabulary for pointer-generator model is only 50000 words. The trainable parameters for baseline Seq2Seq model are 21,499,600 parameters. A few more parameters are added up by the pointer network (1153 parameters), and coverage mechanism (512 parameters).

Initially in the training phase, the Pointer-Generator model is trained without coverage. After training it till a considerably low loss, it is further trained for a further 3000 iterations (about 2 hours) with the coverage mechanism enabled. The Seq2Seq Model with Attention, Pointer Network, and Coverage Mechanism is trained with the CNN/Daily Mail training dataset that consists of 287227 news articles, for 12.4 epochs where each epoch consists of 17952 training steps and each training step trains a batch of size 16. It took 5 days to complete the training of the model using a GPU Tesla T4 in Google Colab.

Each of these models are then tested with almost 11490 text files. Later, they are evaluated using ROUGE scores. The ROUGE-1, ROUGE-2 and ROUGE-L of each model tested on the test data.

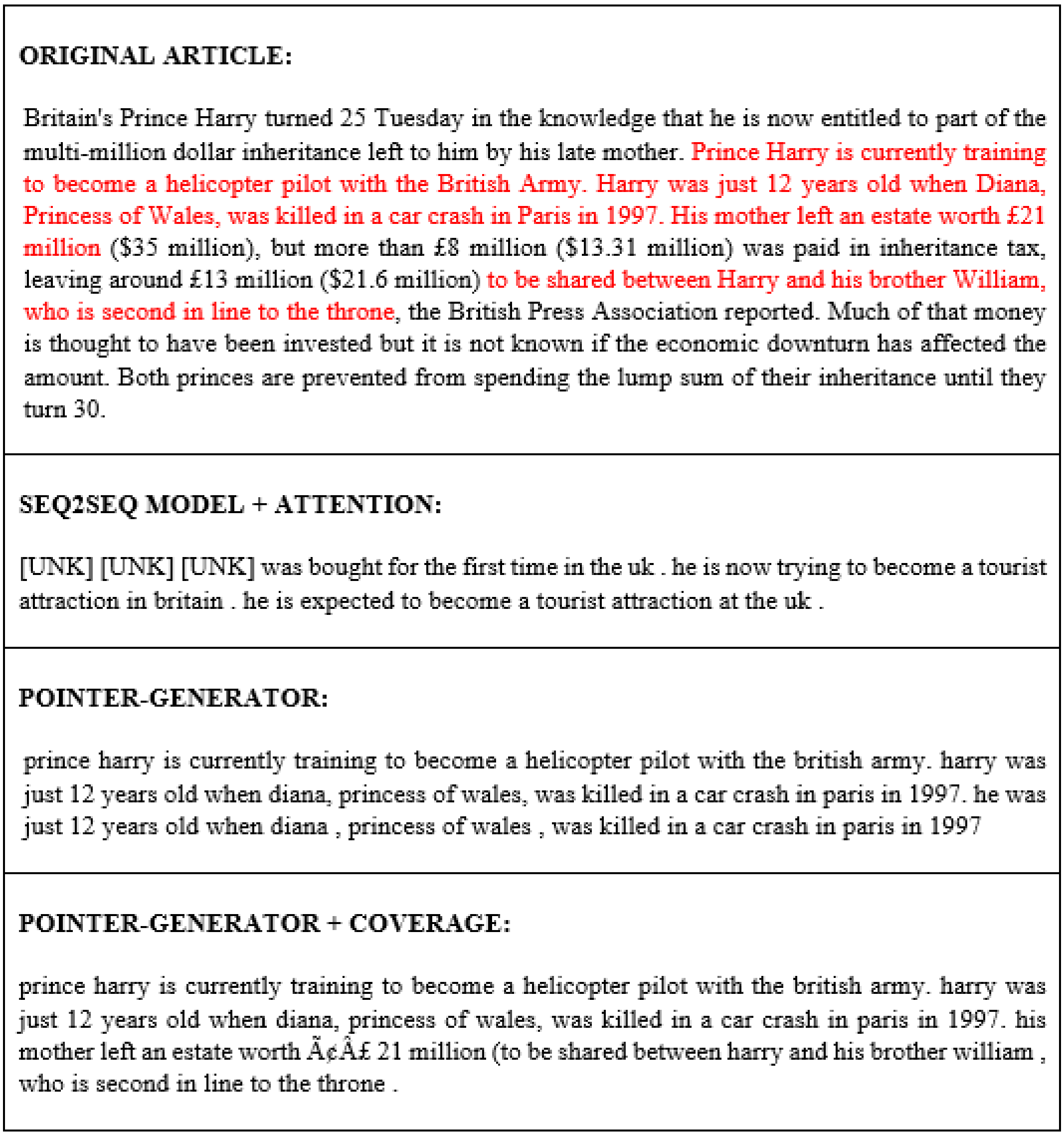
TABLE 1: ROUGE Scores

MODEL	ROUGE-1			ROUGE-2			ROUGE-L		
	F1	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**
Seq2Seq + Attention	0.19	0.18	0.21	0.05	0.05	0.05	0.18	0.17	0.20
Pointer - Generator	0.36	0.38	0.36	0.14	0.15	0.12	0.32	0.34	0.31
Pointer-Generator + Coverage	**0.39**	0.42	0.39	**0.17**	0.18	0.17	**0.36**	0.38	0.36

Table 1 shows the ROUGE scores. The Pointer-Generator with Coverage mechanism model gives the highest ROUGE score. This model obtained better ROUGE-1, ROUGE-2 and ROUGE-L scores of 0.39, 0.17, 0.36 respectively compared to those of Seq2Seq model with Attention and Pointer-Generator models.This model produced semantically correct and meaningful multi-sentence summary efficiently.

Fig 2 shows the output summaries generated by the three models for the same input text. The first model i.e., Seq2Seq model with Attention generates some unknown tokens due to OOV problem. This is rectified by the second model i.e., Pointer-Generator model. Model 1 and 2 both show repetitiveness in the summary. This is solved by model 3 i.e., Pointer-Generator model with Coverage.

Fig 2: Comparison of summaries generated by three models.



ORIGINAL ARTICLE:

Britain's Prince Harry turned 25 Tuesday in the knowledge that he is now entitled to part of the multi-million dollar inheritance left to him by his late mother. Prince Harry is currently training to become a helicopter pilot with the British Army. Harry was just 12 years old when Diana, Princess of Wales, was killed in a car crash in Paris in 1997. His mother left an estate worth £21 million ($35 million), but more than £8 million ($13.31 million) was paid in inheritance tax, leaving around £13 million ($21.6 million) to be shared between Harry and his brother William, who is second in line to the throne, the British Press Association reported. Much of that money is thought to have been invested but it is not known if the economic downturn has affected the amount. Both princes are prevented from spending the lump sum of their inheritance until they turn 30.

SEQ2SEQ MODEL + ATTENTION:

[UNK] [UNK] [UNK] was bought for the first time in the uk . he is now trying to become a tourist attraction in britain . he is expected to become a tourist attraction at the uk .

POINTER-GENERATOR:

prince harry is currently training to become a helicopter pilot with the british army. harry was just 12 years old when diana, princess of wales, was killed in a car crash in paris in 1997. he was just 12 years old when diana , princess of wales , was killed in a car crash in paris in 1997

POINTER-GENERATOR + COVERAGE:

prince harry is currently training to become a helicopter pilot with the british army. harry was just 12 years old when diana, princess of wales, was killed in a car crash in paris in 1997. his mother left an estate worth Ã¢Â£ 21 million (to be shared between harry and his brother william , who is second in line to the throne .

V. CONCLUSION

Abstractive Text Summarization is a more complex and difficult problem compared to extractive summarization. For multi-sentence summary generation CNN/Daily Mail dataset is the best to consider. Abstractive Summarization can be done using Sequence-to-Sequence Model which is further improved by adding Attention, a Pointer network and Coverage mechanism. This model has given better results and meaningful summaries than the baseline Seq2Seq model. The Pointer-Generator Model with Coverage reduces inaccuracies and repetition with the higher ROUGE-1 F1-Score as 0.39.

The performance of the pointer-generator with coverage model can be improved by adding a pre-trained embedding layer like GLOVE. This model generates abstractive summaries to a certain level, but the level of abstraction needs more improvement. It takes longer time to generate a summary. In future, work can be done to decrease the execution time of the model.

REFERENCES

[1] R Nallapati, B Zhou, C Santos, B Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond" , Computational Natural Language Learning, 2016

[2] A. R. Mishra, V. K. Panchal and P. Kumar, "Extractive Text Summarization - An effective approach to extract information from Text," 2019 International Conference on contemporary Computing and Informatics (IC3I), 2019, pp. 252-255, doi: 10.1109/IC3I46837.2019.9055636.

[3] Abigail See, Peter J. Liu and Christopher D. Manning, "Get to the point: Summarization with pointer-generator networks", arXiv preprint arXiv:1704.04368, 2017.

[4] J. Chen and F. You, "Text Summarization Generation Based on Semantic Similarity," 2020 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), 2020, pp. 946-949

[5] Z. Hao, J. Ji, T. Xie and B. Xue, "Abstractive Summarization Model with a Feature-Enhanced Seq2Seq Structure," 2020 5th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS), 2020, pp. 163-167

[6] P. M. Hanunggul and S. Suyanto, "The Impact of Local Attention in LSTM for Abstractive Text Summarization," 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2019, pp. 54-57

[7] H. Kim and S. Lee, "A Context based Coverage Model for Abstractive Document Summarization," 2019 International Conference on Information and Communication Technology Convergence (ICTC), 2019, pp. 1129-1132

[8] S. Ren and Z. Zhang, "Pointer-Generator Abstractive Text Summarization Model with Part of Speech Features," 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), 2019, pp. 1-4

[9] Hermann, Karl Moritz et al, "Teaching machines to read and comprehend", Advances in Neural Information Processing Systems, 2015, pp. 1693-1701.

[10] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks," Neural Information Processing Systems, 2014

[11] D Bahdanau, K Cho, and Y Bengio, "Neural machine translation by jointly learning to align and translate," International Conference on Learning Representations, 2015

[12] Z Tu, Z Lu, Y Liu, X Liu, and H Li, "Modeling coverage for neural machine translation," Association for Computational Linguistics, 2016

[13] W. Nie, W. Zhang, X. Li and Y. Yu, "An Abstractive Summarizer Based on Improved Pointer-Generator Network," 2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC), 2019, pp. 515-520, doi: 10.1109/YAC.2019.8787620.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

☎ 9940 572 462 ☎ 6381 907 438 ✉ ijircce@gmail.com

Scan to save the contact details

www.ijircce.com