Text Summarization With RNN

Kevin Kedrick
Computer Science Department, School
of Computer science
Bina Nusantara University
Jakarta, Indonesia
kevin.kedrick@binus.ac.id

Francis Nicolas Tjan
Computer Science Department, School
of Computer science
Bina Nusantara University
Jakarta, Indonesia
francis.tjan@binus.ac.id

Rayes Jordan Pradana
Computer Science Department, School
of Computer science
Bina Nusantara University
Jakarta, Indonesia
rayes.pradana@binus.ac.id

Abstract—In an age of information overload, efficient text summarization is critical, and this research looks into the usage of Recurrent Neural Networks (RNNs) for this purpose. The study, which focuses on reducing voluminous content into succinct, understandable summaries, recognizes the limitations of existing methodologies and investigates the adaptability of RNNs, which are known for their sequential dependency modeling. The research applies a pre-trained transformer (t5-small) and thorough preprocessing on the Indosum dataset. Experiments using the ROUGE metric show that the model is effective with greater values in metrics such as 'rouge1,' 'rouge2,' and 'rougeL.' This study advances text summarization approaches by demonstrating the capability of RNNs in processing different textual material.

Keywords— RNN, RNNs, Summarization, Summarization with RNN, Text summarization

I. Introduction

In this era where information is easily accessible from the internet, vast amounts of textual data are generated every day and the need for efficient text summarization techniques are now important. The need for a system that can extract important insights from large amounts of textual content has grown as people and businesses struggle to gain insights from large amounts of data. This study explores the field of text summarization using Recurrent Neural Networks (RNNs).

Text summarization is the process of shortening long text into brief yet meaningful summaries that help users to quickly understand the important information contained within the long text. Although conventional techniques show us some efficiency, the ever increasing intricacy and variety of textual data urges us to use more sophisticated and flexible methodologies. RNNs, known for their ability to model sequential dependencies, show promise as a text summarizing method.

This paper aims to create a model for text summarization using RNN. By examining

and understanding the intricacies of RNNs and their unique capabilities, this paper hopes to be able to create a model that will be able to distinguish important and meaningful summaries for a variety of text sources.

II. Literature Review

In recent years, the field of abstractive text summarization has seen significant advancements in deep learning-based approaches. Suleiman and Awajan [1] conducted a comprehensive review of recent approaches, datasets, evaluation measures, and challenges in deep learning-based abstractive text summarization. They categorized the approaches into single-sentence and multi-sentence summary methods, comparing them based on architecture, dataset, dataset preprocessing, evaluation, and results. The review highlighted the prevalence of recurrent neural networks (RNN) and attention mechanisms in the reviewed approaches, with some models utilizing long short-term memory (LSTM) and gated recurrent unit (GRU) to address specific challenges. The authors also discussed the increasing importance of text summarization due to the abundance of online data and emphasized the need for both extractive and abstractive methods to generate high-quality summaries. Additionally, they addressed the challenges encountered in employing various approaches and provided insights into potential solutions. This review serves as a valuable resource for researchers and practitioners in the field of abstractive text summarization, offering a comprehensive overview of the current landscape and future directions.

Song et al. [2] proposed a novel approach to abstractive text summarization using a combination of convolutional neural networks (CNNs) and long short-term memory (LSTM) networks. The study evaluated the performance of the proposed LSTM-CNN model on multi-sentence summaries generated from human-generated abstractive summary bullets from CNN and DailyMail websites. The results showed that the LSTM-CNN model outperformed existing state-of-the-art models in terms of ROUGE scores, indicating its efficacy in generating high-quality abstractive summaries.

Another study on abstractive text summarization was conducted by Masum et al. [3]. The study proposed a method for creating an automatic text summarizer that follows the abstractive method and is able to respond to short length text summaries. The researchers used the Amazon fine food reviews dataset to train their model, which consisted of a bi-directional RNN with LSTMs in the encoding layer and an attention model in the decoding layer. They applied the sequence to sequence model to generate a short summary of food descriptions. The study discussed various necessary factors concerning text summarization, including text processing, vocabulary counting, missing word counting, word embedding, and the efficiency of the model. The proposed method was able to successfully reduce the training loss and create a short summary of English to English text. The study highlights

the importance of improving the efficacy of text summarization models to generate high-quality abstractive summaries.

III. Methodology

This research utilizes the dataset Indosum from nusacrowd. The dataset contains a total of 18,774 records where each record contains 3 columns: ‘ID’. ‘Document’, and ‘Summary’. ‘ID’ refers to the identification number of each record, ‘Document’ refers to the full news text of the records, and ‘Summary’ refers to the summarized version of the ‘Document’ column. While each record contains different types of text, they all come from different sources of news. The dataset is split for training, validating, and testing in a 75/20/5 ratio.

The model that we are using is the Recurrent Neural Network (RNN). RNN is a type of artificial neural network which uses sequential data as training data to learn language translation, speech recognition, and other things. The main characteristic of RNN is that it changes the current input and output based on its “memory” (prior input and output), unlike some deep learning neural networks where the output is independent with the input. This is because since RNN is used mainly for natural language processing, the words in a sentence must be in a specific position and therefore, the RNN needs to determine the position of each word to predict the next word in the sequence.

Another feature that is special to RNN is that it shares parameters across each layer of the network. This means that in RNN the weights are all the same in each layer. However the weights are still changed when backpropagation and gradient descent are happening.

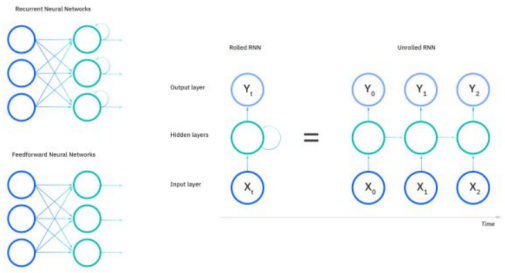

Fig 1. Recurrent Neural Network

A pre-trained transformer (t5-small) will be used for training the model. And before the dataset is used for training, it will be preprocessed first in order to increase accuracy and avoid any outliers. The max length of the summary has also been set as well as each word is turned into tokens/vectors. And since the labels are also text, then there may be a possibility that it exceeds the maximum length set, which is why truncation has also been added. After that, the whole corpus can be tokenized easily using the Dataset.map() function.

IV. Experiment

The metric that is used to evaluate the model’s performance is called the ROUGE metric. The ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric is a set of metric that is mainly used for evaluating summarization or translation text that is usually used in the field of Natural Language Processing by comparing it against a set of references that are human-made. There are various types of

ROUGE, with the most common ones such as ‘rouge1’ which is a unigram (1-gram) based scoring, ‘rouge2’ which is bigram (2-gram) based scoring, ‘rogueL’ which is longest common subsequence based scoring, and ‘rougeLsum’ which splits text using “/n”.

The ROUGE value ranges from 0 to 1. The higher the value is, the more accurate the model is. In this experiment, the metrics that are used as output metrics are ‘Training Loss’, ‘Validation Loss’, ‘rouge1’, ‘rouge2’, ‘rougeL’, ‘rougeLSum’, and ‘Gen Len’. The model was then trained with 4 epochs, with each of the metrics displayed on a table.

Epoch	Training Loss	Validation Loss	Rouge1	Rouge2	Rougel	Rougelsu
1	0.749500	0.522554	0.206100	0.163400	0.203400	0.2034(
2	0.532600	0.492860	0.206300	0.163900	0.203800	0.2037(
3	0.498200	0.484003	0.206500	0.163900	0.203800	0.2038(
4	0.495800	0.480603	0.206500	0.163800	0.203900	0.2038(

Fig 2. Value of Output metrics

After training the model, the input would be the paragraph text that a user wants to summarize.

```
text1 ="""summarize:Peristiwa kebakaran hebat ini terjadi sekitar pukul 11.40 Wita.
Dugaan sementara,api berasal dari hubungan pendek arus listrik dan tidak menimbulkan korban jiwa. 'Ada 13 rumah yang terbakar, semu
Dari jumlah itu, delapan unit rata dengan tanah, tiga rusak berat dan dua rusak ringan,' kata Camat Lambu, M Sidik saat dihubungi
Menurut Sidik, api berasal dari rumah milik Syahbudin.
Kondisi pemukiman yang padat serta bahan bangunan yang mudah terbakar membuat api dengan cepat menjalar ke rumah lainnya.
"""
```

Fig 3. Input text of paragraph

And the model will give an output of the paragraph summarized.

```
[{'summary_text': 'Peristiwa kebakaran hebat ini terjadi sekitar pukul 11.40 Wita. Dugaan sementara, api berasal dari hubungan pendek ar
menimbulkan korban jiwa.'}]
```

Fig 3. Output summary

V. Conclusion

The growing demand for effective text summarizing algorithms in the face of information overload was comprehensively investigated in this study. With the ongoing growth of textual data, the need for systems capable of extracting critical insights from massive databases has become critical. While classic text summarizing methods have their uses, the increasing complexity and diversity of textual data highlights the need for more advanced and adaptive methodologies.

The study's major goal was to use Recurrent Neural Networks (RNNs), which are well-known for their ability to simulate sequential relationships, as a promising approach for text summarization. The project aims to build a model capable of recognizing and generating relevant summaries across varied text sources by delving into the nuances of RNNs and utilizing their unique characteristics. The method made use of the Indosum dataset from nusacrowd, which included 18,774 records with 'ID,' 'Document,' and 'Summary' fields that were meticulously preprocessed. The model was trained using a pre-trained transformer (t5-small), and the evaluation used the ROUGE metric, which included several metrics such as 'Training Loss,' 'Validation Loss,' 'rouge1,' 'rouge2,' 'rougeL,' 'rougeLSum,' and 'Gen Len' throughout four training epochs.
In summary, the ROUGE metrics demonstrated the model's effectiveness in summarizing input texts, with higher values reflecting the RNN-based approach's correctness and success. This study makes a

substantial contribution to the investigation of advanced text summarization algorithms, setting the framework for future advances in this rapidly growing subject.

REFERENCES

[1] Suleiman, Dima, and Arafat Awajan. "Deep learning based abstractive text summarization: approaches, datasets, evaluation measures, and challenges." Mathematical problems in engineering 2020 (2020): 1-29.

[2] Song, Shengli, Haitao Huang, and Tongxiao Ruan. "Abstractive text summarization using LSTM-CNN based deep learning." Multimedia Tools and Applications 78 (2019): 857-875.

[3] Masum, Abu Kaisar Mohammad, et al. "Abstractive method of text summarization with sequence to sequence RNNs." 2019 10th international conference on computing, communication and networking technologies (ICCCNT). IEEE, 2019.