CHAT SUMMARIZATION

Padmaja Phadke

ABSTRACT

This paper aims to conduct a comparative analysis of various models for dialogue summarization, a process that involves condensing actual conversations while preserving essential details. We present a methodology for training three distinct models - Pegasus, BART, and T5 - using the SAMSum dataset, which consists of conversational sentences. Our objective is to assess the accuracy and efficiency of each model in generating summaries. To evaluate the performance of the trained models, we employ the ROUGE metric, which measures the quality of automatic summaries by comparing them to reference summaries based on n-gram overlap and linguistic units. Our experiments on the SAMSum dataset reveal notable differences in the summarization capabilities of the three models. Our findings indicate that the T5 model consistently outperforms Pegasus and BART, achieving higher ROUGE scores across various evaluation metrics. These results provide valuable insights into the effectiveness of different transformer models for dialogue summarization tasks.

1. INTRODUCTION

Abstractive summarization serves as a fundamental task within natural language processing (NLP), condensing original text into a concise form while preserving crucial information. While previous abstractive summarization methods have shown significant advancements, their focus has primarily revolved around news datasets. Recent research, however, has shifted towards abstractive conversation summarization, which encompasses interactions between multiple speakers. This novel approach aims to offer succinct summaries of conversations involving two or more participants, aiding in content comprehension for both participants and non-participants alike.

Traditional abstractive summarization models are primarily tailored for single-participant document summarization, rendering them less suitable for conversation summarization due to the intricate dialogue structures. Conversations often entail vital information dispersed across various utterances from multiple participants. Moreover, dialogues involve dynamic information exchange, frequent topic shifts, and possess a relatively informal and repetitive nature.

Abstractive summarization technology predominantly relies on sequence-to-sequence, encoder-decoder, and Text-To-Text Transfer Transformer models. To overcome the challenges posed by conversation summarization, fine-tuning pre-trained transformer models has emerged as a popular strategy. This approach entails adapting pre-trained models on domain-specific datasets to enhance their performance on specific tasks. In this study, our focus lies on fine-tuning a pre-trained transformer model for sentence

summarization using the SAMSum dataset. The SAMSum dataset comprises human-annotated summaries of conversations, rendering it an ideal resource for training and evaluating summarization models on conversational data. We aim to leverage three models, namely Pegasus, T5, and BART, to train and assess their efficacy in producing quality outputs.

To evaluate the effectiveness of our fine-tuned model, we utilize ROUGE metrics, a well-established method for assessing summarization model performance. These metrics measure the degree of overlap between the generated summaries and human reference summaries, focusing on key aspects such as ROUGE1, ROUGE2, ROUGEL, and ROUGELSUM scores. It's important to note that while higher ROUGE scores are generally considered favorable, the interpretation of what constitutes a "good" ROUGE score can differ based on the specific needs and standards of the task at hand.

Furthermore, we deploy the fine-tuned model through a Flask interface to enhance accessibility for a broader audience. Our approach holds promise for enhancing the efficacy of pre-trained transformer models in summarizing conversational data.

2. PROPOSED METHODS

2.1 Dataset Description

We utilized the SAMSum dataset, developed in 2019 by the Samsung R&D Institute. This dataset, sourced from customer service chat conversations, is highly regarded due to its human annotation. SAMSum encompasses diverse natural conversations, including casual chatter, gossip, political discussions, and academic consultations among colleagues. Comprising 16,000 dialogue instances, each accompanied by a manually crafted summary, SAMSum is partitioned into 14,732 training samples, 818 validation samples, and 819 test samples, with a split ratio of 90:5:5 for training, validation, and testing, respectively. Leveraging this dataset across various models has demonstrated notable enhancements in text summarization performance. The following Table 1 illustrates the partitioning of the SAMSum dataset and how it is divided into different subsets.

Dialogue	Train set	Valid set	Test set
Sample numbers	14732	818	819
Speakers number avg.	2.40	2.39	2.36
Dialogue turns avg.	11.17	10.83	11.25
Summaries length avg	23.44	23.42	23.12

Table 1

2.2 Model Description

We're employing three models—T5, Pegasus, and BART—for chat summarization tasks, all trained on the SAMSum dataset. Below, we present an overview of these three models.

2.2.1 Pegasus

The Pegasus model follows a sequence-to-sequence framework with an encoder-decoder architecture, allowing it to efficiently process input sequences and generate corresponding output summaries. During pre-training, Pegasus employs a dual objective approach, consisting of Masked Language Modeling (MLM) and Gap Sentence Generation (GSG). While MLM tasks involve predicting masked tokens within the input sequence, GSG tasks focus on generating missing sentences within the document. By combining these two objectives, Pegasus gains a comprehensive understanding of the input text and learns to produce informative and concise summaries. The utilization of MLM and GSG objectives during pre-training enhances Pegasus's ability to capture key information from the input documents and generate coherent summaries. This approach ensures that the model effectively compresses the input content while preserving its meaning and structure in the generated summaries.

2.2.2 T5 Model

The T5-base model, developed by Google's AI research team, stands as a testament to the remarkable advancements in natural language processing (NLP) enabled by transformer-based architectures. T5-base represents a paradigm shift in NLP by adopting a text-to-text approach, where both input and output are represented as text strings, providing a unified framework for various tasks. This approach allows T5-base to seamlessly handle a wide range of NLP tasks, including text categorization, language translation, summarization, and question answering. Through extensive pre-training on diverse datasets such as Common Crawl and Wikipedia, T5-base learns to map input text sequences to output text sequences, gaining a deep understanding of natural language patterns and semantics. The model's proficiency across different tasks, coupled with its state-of-the-art performance, has positioned T5-base as a cornerstone in the field of NLP. Moreover, T5-base's compatibility with standard computing resources, including CPUs and GPUs, makes it accessible for researchers and practitioners alike, paving the way for further advancements and applications in the realm of text processing and understanding.

2.3.3 BART

BART, an acronym for Bidirectional and Auto-Regressive Transformers, is a transformer-based encoder-decoder model developed for a variety of natural language processing (NLP) tasks. Unlike traditional sequence-to-sequence models, BART features both a bidirectional encoder and an autoregressive decoder, allowing it to efficiently process

input sequences bidirectionally while generating output sequences autoregressively. During pre-training, BART leverages a unique approach by training on corrupted text with a reconstruction objective. This process involves corrupting the input text and training the model to reconstruct the original text, enabling BART to learn robust representations of language and effectively handle noisy or imperfect data. BART is particularly effective for text generation tasks such as summarization and translation, as well as comprehension tasks like text classification and question answering. Its bidirectional encoder enables comprehensive understanding of input text, while the autoregressive decoder facilitates accurate and fluent generation of output sequences.

2.3 Algorithm

1. Load the Pegasus model and tokenizer.
2. Load the SamSum dataset.
3. Preprocess the dataset by converting examples to model-compatible features.
4. Set up the training pipeline with data collator, training arguments, and Trainer.
5. Fine-tune the Pegasus model on the SamSum dataset.
6. Evaluate the fine-tuned model on the test set using the ROUGE metric.
7. Save the fine-tuned model and tokenizer.
8. Load the saved model and tokenizer.
9. Generate a summary using the fine-tuned model and compare it to the reference summary.

2.3 ROUGE SCORE

2.3.1 ROUGE-1: Measures the overlap of unigrams (individual words) between the system-generated summary and the reference summary.

$$ROUGE-1 = \frac{\sum_i^N min(\text{count}_{\text{match}}(gram_i), \text{count}_{\text{reference}}(gram_i))}{\sum_i^N \text{count}_{\text{reference}}(gram_i)}$$

2.3.2 ROUGE-2: Measures the overlap of bigrams (pairs of adjacent words) between the system-generated summary and the reference summary.

$$ROUGE-2 = \frac{\sum_i^{N-1} min(\text{count}_{\text{match}}(gram_i), \text{count}_{\text{reference}}(gram_i))}{\sum_i^{N-1} \text{count}_{\text{reference}}(gram_i)}$$

2.3.3 ROUGE-L: Measures the longest common subsequence (LCS) between the system-generated summary and the reference summary, considering word order.

$$ROUGE-2 = \frac{\sum_i^{N-1} min(\text{count}_{\text{match}}(gram_i), \text{count}_{\text{reference}}(gram_i))}{\sum_i^{N-1} \text{count}_{\text{reference}}(gram_i)}$$

2.3.4 ROUGE-Lsum: Computes the F1 score (harmonic mean of precision and recall) based on ROUGE-L, reflecting both precision and recall in the evaluation.

$$ROUGE - Lsum = F1_{\text{ROUGE-L}} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

3. EXPERIMENTS

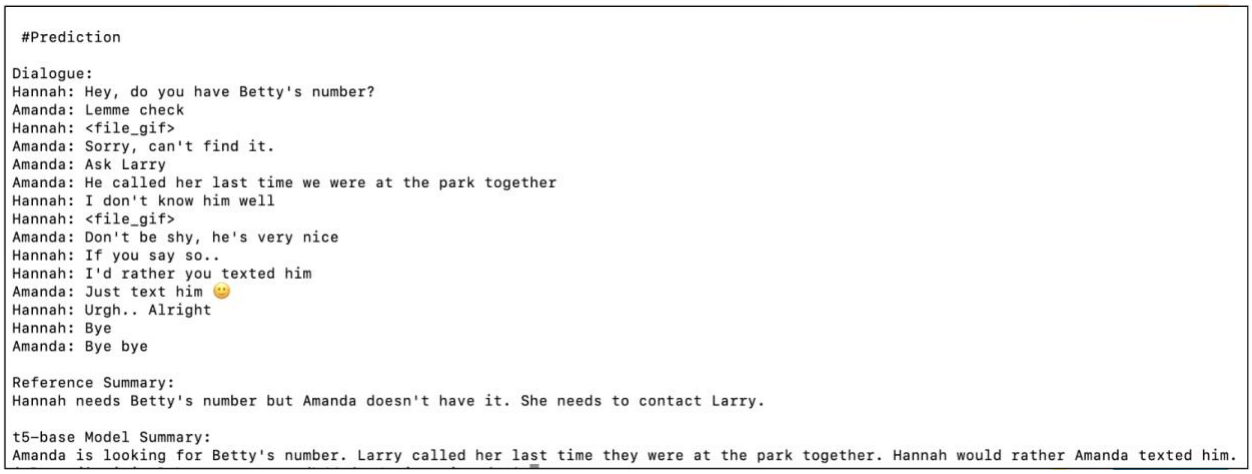
During the experimental phase, we conducted training sessions for three distinct models utilizing the SAMSum dataset over a span of 25 epochs. Subsequently, we meticulously calculated the ROUGE scores to evaluate their performance. The resultant table encapsulates the ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-Lsum metrics for each model as mentioned in Table 2

Model/Score	ROUGE1	ROUGE2	ROUGEL	ROUGELSum
Pegasus	0.01816	0.000336	0.018052	0.018114
Bart	0.012431	0.000227	0.0124	0.012417
T5- base	0.028476	0.00049	0.028014	0.028042

Table 2

These scores offer valuable insights into the efficacy of each model's summarization capabilities, shedding light on their individual strengths and areas for potential improvement.

The ROUGE scores appear relatively low, which could be attributed to the limited number of epochs. It's worth noting that these scores may improve with additional epochs. Notably, the Bart model yielded lower scores across all ROUGE metrics, while the Pegasus model performed comparatively better. However, among the three models, T5 demonstrated the highest performance, showing promising potential for further exploration.

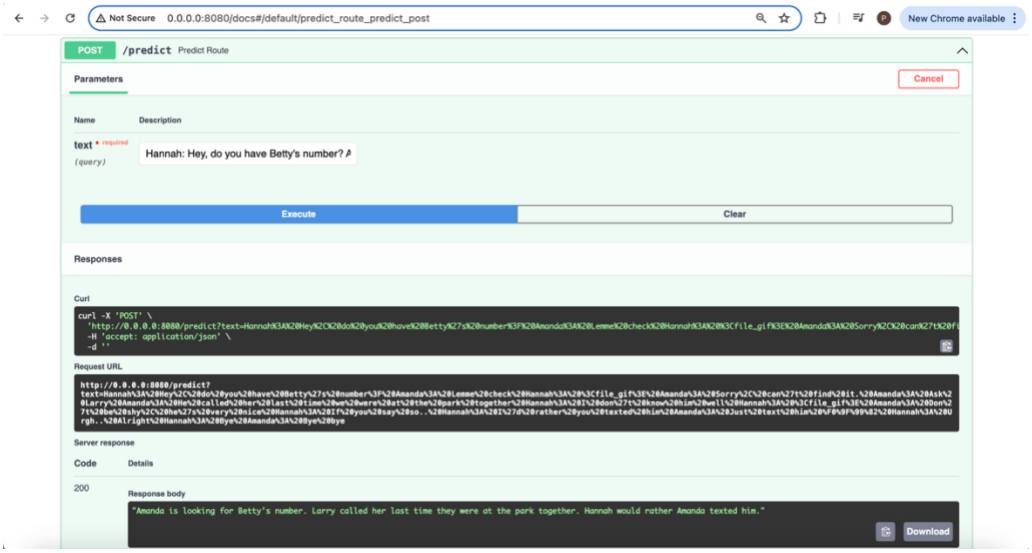


Example 1

Here is an instance of T5 summarization, illustrating a conversation between two individuals alongside the actual summary. Additionally, after training the T5-base model for 25 epochs, the model-generated summary is presented. While the current performance is satisfactory, further enhancement could be achieved by increasing the number of training epochs.

I have developed a user interface that allows users to input dialogue and receive a summary. Below is a Screenshot 1 demonstrating the T5 model's capability to summarize a conversation between two individuals along with the generated summary.

I've developed a user interface that allows users to input dialogue and receive a summary. Screenshot 1 demonstrating the T5 model's capability to summarize a conversation between two individuals along with the generated summary.



4. CONCLUSION

In this paper, we introduce three models—T5, Pegasus, and Bart—for extractive text summarization. After becoming acquainted with the Samsum dataset, we trained these models on it and computed the ROUGE scores for evaluation, deriving insights from the resulting table. We then chose the model with the highest ROUGE score to proceed with the user interface development for this project. To enhance the ROUGE metric further, additional training epochs beyond 25 could prove beneficial.

5. REFERENCE

[1] Pengyao Yi and Ruifang Liu "A Relation Enhanced Model for Abstractive Dialogue Summarization." 2022 International Conference on Cyber-enabled Distributed Computing and Knowledge Discovery (CyberC).

[2] Yuejie Lei, Yuanmeng Yan, Zhiyuan Zeng, Keqing He, Ximing Zhang, Weiran Xu "HIERARCHICAL SPEAKER-AWARE SEQUENCE-TO-SEQUENCE MODEL FOR DIALOGUE SUMMARIZATION." ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414547

[3] Rohan Habu, Rohit Ratnaparkhi, Sunita Kulkarni, and Anjali Askhedkar "A Hybrid Extractive-Abstractive Framework with Pre & Post-Processing Techniques To Enhance Text Summarization"

[4Jiangnan Du1,*, Xuan Fu2,†, Jianfeng Li1 Cuiqin Hou1, Qiyu Zhou, Hai-Tao Zheng2,Ping An Technology (Shenzhen) Co., Ltd., ShenZhen, China Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China Pengcheng Laboratory, Shenzhen, China Corresponding author "A Simple Semantics and Topic-aware Method to Enhance Abstractive Summarization"

[5] Tanmayee Behere, Avani Vaidya, Anamika Birhade, Komal Shinde, Pranjali Deshpande, and Sunita Jahirabadkar "TEXT SUMMARIZATION AND CLASSIFICATION OF CONVERSATION DATA BETWEEN SERVICE CHATBOT AND CUSTOMER" 978-1-7281-6823-4/20/$31.00 c2020 IEEE

[6] Dr. Ahmed T. Sadiq, Dr. Yossra H. Ali, and M.Sc. Mohammad Natiq Fadhil"Text summarization for social network conversation" 2013 International Conference on Advanced Computer Science Applications and Technologies

[7] Yuliska and Tetsuya Sakai "A Comparative Study of Deep Learning Approaches for Query-Focused Extractive Multi-Document Summarization" 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT)

[8] Yuri Nakayama, Tsukasa Shiota, and Kazutaka Shimada "Corpus construction for topic-based summarization of multi-party conversation" 2021 International Conference on Asian Language Processing (IALP)

[9] Y. Dong, S. Wang, Z. Gan, Y. Cheng, J. C. K. Cheung, and J. Liu, "Multi-fact correction in abstractive text summarization," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Online: Association for Computational Linguistics, Nov. 2020, pp. 9320–9331. [Online]. Available: https://aclanthology.org/2020.emnlp-main.749

[10] Zhou Zhao, Haojie Pan, Changjie Fan, Yan Liu, Linlin Li, Min Yang, and Deng Cai, "Abstractive meeting summarization via hierarchical adaptive segmental network learning," in The World Wide Web Conference, 2019, pp. 3455–3461.

[11] Guokan Shang, Wensi Ding, Zekun Zhang, Antoine Jean-Pierre Tixier, Polykarpos Meladianos, Michalis Vazirgiannis, and Jean-Pierre Lorr´e, "Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization," arXiv preprint arXiv:1805.05271, 2018.

[12] Alexander M Rush, Sumit Chopra, and Jason Weston, "A neural attention model for abstractive sentence summarization," arXiv preprint arXiv:1509.00685, 2015.

[13] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al., “Abstractive text summarization using sequence-to-sequence rnns and beyond,” arXiv preprint arXiv:1602.06023, 2016.
[14] Shichao Sun and Wenjie Li. 2021. Alleviating exposure bias via contrastive learning for abstractive text summarization. CoRR, abs/2108.11846.
[15] Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1073–1083, Vancouver, Canada