Advancing Human-Like Summarization: Approaches to Text Summarization

Saliq Gowhar, Bhavya Sharma, Ashutosh K Gupta and Anand Kumar Madasamy

Department of Information Technology, National Institute of Technology Karnatka, Surathkal, India

Abstract

Text summarization, a well-explored domain within Natural Language Processing, has witnessed significant progress. The ILSUM shared task, encompassing various languages, such as English, Hindi, Gujarati, and Bengali, concentrates on text summarization. The proposed research focuses on leveraging pretrained sequence-to-sequence models for abstractive summarization specifically in the context of the English language. This paper provides an extensive exposition of our model and approach. Notably, we achieved the top ranking in the English Language subtask. Furthermore, this paper dives into an analysis of various techniques for extractive summarization, presenting their outcomes and drawing comparisons with abstractive summarization.

Keywords

Text Summarization, Sequence-to-Sequence models, Abstractive and Extractive Summarization.

1. Introduction

In this ever-expanding digital age, textual information has grown exponentially, due to which effective information retrieval and comprehension has become a major challenge. Text Summarization, a vital and a heavily researched prospect of Natural Language Processing, has emerged as a crucial solution to this challenge. It aims to condense large texts into concise human-like summaries, providing the readers with key information while sparing them the effort of reading extensive documents of large volumes. With the rapid advancements in NLP and Machine Learning, the area of text summarization has seen rapid growth and advancements despite the absence of large and high-quality datasets.

Text summarization can be either abstractive or extractive. Abstractive summarization being a more efficient form of summarization [1], is a technique where the system generates a summary by understanding the content of the document and then creating a summary using it's own understanding of the document, hence making it a more effective technique capable of generating human-like summaries. It can generate a summary which contains words that may or may not be available in the original document. Several pretrained sequence-to-sequence models exist which can be used for abstractive summarization including T5[2], BART[3], ProphetNet[4] and Pegasus[5]. Extractive summarization on the other hand is a technique which maintains the original information content of the document [6], and works by selecting and extracting

Forum for Information Retrieval Evaluation, December 15-18 2023, India

saliqgowhar.211ee250@nitk.edu.in (S. Gowhar); shrmabhav.211ai011@nitk.edu.in (B. Sharma); ashutosh.211ai008@nitk.edu.in (A. K. Gupta); m_anandkumar@nitk.edu.in (A. K. Madasamy)

the sentences or phrases directly from the original text that are considered the most important representatives of the document's content. The sentences are ranked as per their importance which can be calculated using various algorithms like TextRank[7], TF-IDF[7] and K-Means[8].

In this study, we have implemented abstractive and extractive summarization methods for English language within the framework of the FIRE shared task 2023 - ILSUM [16], making use of the dataset furnished by the event organizers. Key takeaways of the task can be found in [17]. We conducted summarization on both the raw and preprocessed data, utilizing the ILSUM 2022 dataset for evaluation. Abstractive summarization was executed using Google's T5 transformer model, while extractive summarization was implemented with TF-IDF, Term Frequency, K-Means and BERT based algorithms. In the context of the shared task, we exclusively submitted results derived from abstractive techniques, reserving the application of extractive summarization methods solely for comparative analysis. Evaluation has been done using ROUGE-N scores along with their respective precision, recall and F1 measures.

2. Related Work

Ranganathan et al.[9], used fine-tuned T5 transformer model for abstractive summarization on the UCI drug review and BBC datasets. Lalitha et al.[10], fine-tuned T5, BART, and Pegasus for abstractive summarization of medical documents using the SUMPUBMED dataset. Jadeja et al.[11], performed a comparative analysis between state-of-the-art text summarizers including T5 and Pegasus, using the WikiHow dataset, and evaluations have been made manually by humans and as well as using metrics like ROUGE and BLEU. Ladhak et al.[12], introduced WikiLingua, a multilingual dataset containing article-summary pairs in 18 distinct languages. In their experiments, they fine-tuned the mBART model using this dataset.

Aljević et al.[13], proposed a novel graph based approach for extractive summarization that involves transforming a given text into a network of interconnected sentences and utilizes a computationally efficient selectivity measure to assess the significance of these graph nodes. Jewani et al.[14] performed a brief study and comparisons of major extractive summarization techniques including TF-IDF, Clustering, Fuzzy logic, Neural Network and Graph based approaches. Souza et al.[15] propose a novel multi-view approach for extractive summarization by treating it as a binary classification problem.

3. Corpus Description

The dataset released for this task was created by extracting data from several leading Indian newspaper websites. We utilised the English language dataset released under ILSUM 2.0 in 2023 for our experimentation which consisted of train, test and validation datasets. The train set consists of 28,347 news-article and summary pairs, along with the ids and Headings, whereas the test set contains 2895 news article along with the respective ids and headings. In case of the test set, the summaries were not provided and kept hidden officially for evaluation purposes. Hence for our own evaluation purposes, we made use of the official test dataset released under ILSUM 1.0 in 2022 for English Language which consists of 4487 articles along with their ids,

headings and human-reference summaries. Given this dataset, the task was to generate fixed length summaries overcoming the challenge of script mixing.

4. Model Description

T5 (Text-To-Text Transfer Transformer) [2], is a versatile and powerful transformer-based architecture for natural language processing. It is unique in that it treats all NLP tasks as text-to-text tasks, which allows it to perform exceptionally well on a wide range of language understanding and generation tasks, from translation and summarization to question-answering and text classification. T5 has been pre-trained on the Colossal Clean Crawled Corpus (C4), which is a mixture of unsupervised and supervised data, making it capable of handling various tasks with the same underlying architecture. For our experimentations, we used the T5-Base variant, which is characterized by a model checkpoint containing 220 million parameters.

5. Methodology

5.1. Preprocessing

We perform our experiments on both the original dataset as well as the preprocessed dataset. As part of preprocessing we performed multiple steps which include:

- Lowercasing and conversion to string format.
- Removal of numerical digits and special characters from the text.
- Replacing newline characters with spaces.
- Replacing consecutive occurrences of special characters wih single spaces.
- Removal of emoticons from the text.

Furthermore, we organized our data to match the format that T5 expects for summarization tasks. We focused solely on the ”Article” and ”Summary” columns and removed any other columns. We also renamed the ”Article” column as ’ctext’ and the ”Summary” column as ’text.’ After that, we added the prefix ’summarize: ’ to the start of each article.

5.2. Creating a Custom Dataset Object

We create a Custom Dataset object for our data that is used particularly for text summarization in transformer based architectures.

- We initialise various attributes which include the tokenizer used, text to be summarized, the reference summaries, maximum length of source text, maximum length of target text and the dataframe.
- We retrieve the individual data samples from the dataset provided the index of the sample. Tokenization is performed for both the articles and summaries using the T5Tokenizer and the tokenized input and target sequences are obtained with a maximum length and padding to ensure consistent shapes for model input.

- The resulting tokenized sequences include the Input IDs and the attention masks. We return a dictionary for each sample containing the following keys:
  - source_ids: The input IDs for the source text.
  - source_mask: The attention mask for the source text.
  - target_ids: The input IDs for the target text.
  - target_mask: The attention mask for the target text.

5.3. Setting up the parameters

We initialize a weights & biases (wandb) project to keep track of our experiments and set up the wandb configuration for our exerimentation. In this experimentation, we have trained the model on the entire dataset using the Hyper-parameter settings given in Table 1.

Table 1
Parameter settings for T5

Parameters	Values
Epochs	8
Max source length	512
Max target length	75
Batch Size	2
Learning Rate	5e-5
Beams	4
Length penalty	1
Repetition penalty	2.5

6. Results

As per the official results for the ILSUM 2023 task, our team NITK-AI (SCaLAR[1]) was able to achieve notable scores. Specifically, our performance in terms of the ROUGE metrics was as follows: a ROUGE-1 score of 0.3321, a ROUGE-2 score of 0.1731, a ROUGE-4 score of 0.121, and a ROUGE-L score of 0.282. Additionally, when assessing our results using the BERT Score, we obtained a recall score of 0.8752, a precision score of 0.8684, and an F1 measure of 0.8716. The official ROUGE scores are given in Table 2 and official BERT scores are given in Table 3.

7. Comparative analysis

We evaluated the performance of T5 model on ILSUM 2022 test data using ROUGE-N metrics on both the original dataset as well as the preprocessed dataset. The results obtained are given in Table 4.

[1]https://scalar-nitk.github.io/website/

Table 2
Official ROUGE score results

Team Name	Rouge-1 F1	Rouge-2 F1	Rouge-4 F1	Rouge-L F1
NITK-AI (SCaLAR)	0.3321	0.1731	0.121	0.282
Eclipse	0.3022	0.1111	0.042	0.2504
BITS Pilani	0.2354	0.0604	0.0147	0.182
ASH	0.137	0.017	0.0004	0.1181
ILSUM_2023_SANGITA	0	0	0	0

Table 3
Official BERT score results

Team Name	Bert_Score_P	Bert_Score_R	Bert_Score_F
NITK-AI (SCaLAR)	0.8752	0.8684	0.8716
Eclipse	0.8505	0.8733	0.8616
BITS Pilani	0.8724	0.8462	0.8589
ASH	0.8277	0.8036	0.8153
ILSUM_2023_SANGITA	0	0	0

Table 4
ROUGE Metrics for T5 model on 2022 dataset

Dataset	Sub-Metric	ROUGE-1	ROUGE-2	ROUGE-L
Original Dataset	Recall	0.432	0.335	0.406
	Precision	0.488	0.376	0.457
	F1-Measure	0.451	0.350	0.424
Pre-processed Dataset	Recall	0.321	0.185	0.289
	Precision	0.313	0.175	0.282
	F1-Measure	0.310	0.176	0.280

Additionally, we conducted a comparative analysis that involved evaluating the performance of the T5 model for abstractive summarization and comparing it with several extractive summarization techniques including TF-IDF, Frequency based approach, K-Means and BERT based approach. This analysis was done using the same ILSUM 2022 dataset using the same ROUGE metrics, but this time we only used the original dataset for comparative analsyis as it gave us the best results using T5.

In frequency based approach, we tokenize the sentences of an article and rank the sentences based on the frequency of its words from highest to lowest. In TF-IDF based approach, we tokenize the article into individual sentences, followed by creating a TF-IDF matrix, which assigns weights to words in each sentence. We then compute the cosine similarity between each sentence and entire document, measuring how similar each sentence is to the overall content. Finally, we rank the sentences based on these scores and choose the top n sentences as the summary. K-means for extractive summarization involves representing sentences numerically using Word2Vec, clustering them with k-means, and selecting representative sentences, often centroids or those closest to centroids, to create the summary. In BERT based approach, we take the context embedding of the entire article and of every sentence then select the top sentences

with the highest similarity with the article context. Additionally, we tried our novel method where instead of providing the entire atricle as input to the T5 model, we only provided the summary from BERT as the input, and performed abstractive summarization on the same. This resulted in increase in overall ROUGE scores, hence proving to be an effective and time efficient approach. The results obtained using these methods are given in Table 5. It can cleared be deduced that abstractive summarization gives better results as compared to the above mentioned extractive summarization approaches but almost similar results with our novel approach.

Table 5
ROUGE metrics using Extractive Summarization on 2022 dataset

Approach	Sub-Metric	ROUGE-1	ROUGE-2	ROUGE-L
Term Frequency	Recall	0.222	0.107	0.196
	Precision	0.223	0.097	0.193
	F1-measure	0.214	0.098	0.187
TF-IDF	Recall	0.340	0.180	0.313
	Precision	0.188	0.086	0.171
	F1-measure	0.218	0.101	0.199
K-Means	Recall	0.477	0.239	0.425
	Precision	0.092	0.019	0.075
	F1-measure	0.146	0.034	0.121
BERT	Recall	0.793	0.704	0.776
	Precision	0.266	0.189	0.259
	F1-measure	0.377	0.280	0.368
BERT + T5	Recall	0.398	0.296	0.370
	Precision	0.476	0.358	0.443
	F1-measure	0.426	0.317	0.396

8. Multilingual Summarization

We conducted experiments to perform Abstractive Summarization on Hindi language using ILSUM 2023 Hindi data for training and ILSUM 2022 Hindi data for evaluation. We made use of pre-trained sequence-to-sequence models including MT5 and IndicBart, and trained them on our entire training dataset for effective results. IndicBART is a multilingual, sequence-to-sequence pre-trained model focusing on Indic languages and English. It currently supports 11 Indian languages and is based on the mBART architecture. It is trained on Indian Language Corpora containing 452 million sentences and 9 Billion tokens. Multilingual T5 (mT5) is a variant of the Text-to-Text Transfer Transformer (T5) model designed to handle diverse languages. It is trained on a wide range of languages, allowing it to perform various natural language processing tasks, such as translation, summarization, and question-answering, across multiple linguistic contexts. In this experimentation, we have trained the MT5 model on the entire dataset using the Hyper-parameter settings given in Table 6.

The results obtained on Hindi dataset using the models mentioned are given in Table 7.

Table 6
Parameter settings for MT5

Parameters	Values
Epochs	10
Max source length	512
Max target length	75
Batch Size	2
Learning Rate	5e-5
Beams	4
Length penalty	1
Repetition penalty	2.5

Table 7
ROUGE metrics on Hindi dataset

Model	Sub-Metric	ROUGE-1	ROUGE-2	ROUGE-L
MT5	Recall	0.437	0.274	0.386
	Precision	0.539	0.353	0.473
	F1-measure	0.472	0.300	0.416
IndicBart	Recall	0.731	0.615	0.676
	Precision	0.461	0.341	0.424
	F1-measure	0.557	0.431	0.513

9. Conclusion and Future works

In this paper, we present our work on performing summarization of English text as a part of the Forum for Information Retrieval Evaluation 2023 shared task, ILSUM. We conducted experiments using the T5 transformer-based model for abstractive summarization, achieving significant results. Additionally, we explored extractive summarization techniques and conducted a comparative analysis between abstractive and extractive methods, demonstrating the superior efficiency of abstractive approaches. Due to computational constraints, we submitted results only for English language, securing first position in the subtask as well.

As part of our future research within this project, we plan to explore other transformer-based models for abstractive summarization, such as PEGASUS and BART. Furthermore, we aim to extend our work to cover other Indian languages, including Bengali, Hindi, and Gujarati, using multilingual transformer models like mT5 and IndicBART. We also intend to conduct comparative analyses involving large language models (LLMs) such as Llama 2 and perform a deepened error analysis. We anticipate that this work will provide valuable insights and directions for future research in this domain.

Acknowledgments

We would like to express our sincere gratitude to the organizers of the ILSUM Shared Task and Forum for Information Retrieval Evaluation (FIRE) for curating a high-quality dataset of Indian

language texts, paving the way for high quality research in the field.

References

[1] N. Moratanch and S. Chitrakala, "A survey on abstractive text summarization," 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India, 2016, pp. 1-7, doi: 10.1109/ICCPCT.2016.7530193.

[2] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, et al., Exploring the limits of transfer learning with a unified text-to-text transformer., J.Mach. Learn. Res. 21 (2020) 1–67.

[3] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019)

[4] W. Qi, Y. Yan, Y. Gong, D. Liu, N. Duan, J. Chen, R. Zhang, M. Zhou, Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training, arXiv preprint arXiv:2001.04063 (2020).

[5] J. Zhang, Y. Zhao, M. Saleh, P. Liu, Pegasus: Pre-training with extracted gap-sentences for abstractive summarization, in: International Conference on Machine Learning, PMLR, 2020, pp. 11328–11339.

[6] S. R. Rahimi, A. T. Mozhdehi and M. Abdolahi, "An overview on extractive text summarization," 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, Iran, 2017, pp. 0054-0062, doi: 10.1109/KBEI.2017.8324874.

[7] S. Zaware, D. Patadiya, A. Gaikwad, S. Gulhane and A. Thakare, "Text Summarization using TF-IDF and Textrank algorithm," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 1399-1407, doi: 10.1109/ICOEI51242.2021.9453071.

[8] K. Shetty and J. S. Kallimani, "Automatic extractive text summarization using K-means clustering," 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), Mysuru, India, 2017, pp. 1-9, doi: 10.1109/ICEECCOT.2017.8284627.

[9] J. Ranganathan and G. Abuka, "Text Summarization using Transformer Model," 2022 Ninth International Conference on Social Networks Analysis, Management and Security (SNAMS), Milan, Italy, 2022, pp. 1-5, doi: 10.1109/SNAMS58071.2022.10062698.

[10] E. Lalitha, K. Ramani, D. Shahida, E. V. S. Deepak, M. H. Bindu and D. Shaikshavali, "Text Summarization of Medical Documents using Abstractive Techniques," 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2023, pp. 939-943, doi: 10.1109/ICAAIC56838.2023.10140885.

[11] D. Jadeja, A. Khetri, A. Mittal and D. K. Vishwakarma, "Comparative Analysis of Transformer Models on WikiHow Dataset," 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2022, pp. 655-658, doi: 10.1109/ICSCDS53736.2022.9761043.

[12] Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A New Benchmark Dataset for Cross-Lingual Abstractive Summarization. In Findings of

the Association for Computational Linguistics: EMNLP 2020, pages 4034–4048, Online. Association for Computational Linguistics.

[13] D. Aljević, L. Todorovski and S. Martinčić-Ipšić, "Extractive Text Summarization Based on Selectivity Ranking," 2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), Kocaeli, Turkey, 2021, pp. 1-6, doi: 10.1109/INISTA52262.2021.9548408.

[14] K. Jewani, O. Damankar, N. Janyani, D. Mhatre and S. Gangwani, "A Brief Study on Approaches for Extractive Summarization," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 2021, pp. 601-608, doi: 10.1109/ICAIS50930.2021.9396031.

[15] C. M. Souza, M. R. G. Meireles and R. Vimieiro, "A multi-view extractive text summarization approach for long scientific articles," 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 2022, pp. 01-08, doi: 10.1109/IJCNN55064.2022.9892526.

[16] S. Satapara, P. Mehta, S. Modha, and D. Ganguly, "Indian language summarization at fire 2023," in *Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2023, Goa, India. December 15-18, 2023*, ACM, 2023.

[17] S. Satapara, P. Mehta, S. Modha, and D. Ganguly, "Key takeaways from the second shared task on indian language summarization (ilsum 2023)," in *Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation, Goa, India. December 15-18, 2023* (K. Ghosh, T. Mandl, P. Majumder, and M. Mitra, eds.), CEUR Workshop Proceedings, CEUR-WS.org, 2023.