Enhancing News Article Summarization Through Post-Processing Techniques

A Fathima Firose[1], Reshma Sheik[2], KathijaAfrose A[3] and S. Jaya Nirmala[4]

[1]*Thiagarajar College of Engineering, Madurai, India*
[2]*Thangal Kunju Musaliar College of Engineering, Kollam, India*
[3]*Indian Institute of Technology, Jodhpur, India*
[4]*National Institute of Technology, Trichy, India*

Abstract

This study explores the enhancement of automatic news article summarization by incorporating a series of post-processing techniques into pre-trained language models. The work focuses on improving summary quality by addressing common issues such as redundancy and lack of conciseness through repetition avoidance, sentence length control, and the inclusion of article headlines in the summarization process. Using fine-tuned models across four languages—English, Tamil, Telugu, and Kannada—the study benchmarks the impact of these techniques on summary quality. Evaluation metrics such as ROUGE and BERTScore were used to measure improvements in both informativeness and fluency. The results show that applying post-processing techniques significantly enhances the performance of the models, yielding more coherent, concise, and contextually relevant summaries. These findings provide a strong framework for improving summarization models in multilingual settings, with implications for more effective news content generation and summarization in various languages.

Keywords

News Article Summarization, Post-Processing Techniques, Redundancy, Repetition Avoidance, Sentence Length Control, Pre-Trained Language Models, ROUGE Metrics, BERTScore, Summary Quality, Informative Summaries, Contextual Relevance

1. Introduction

The significance of effective summarization techniques has grown as the volume of digital content continues to expand. Summarization, a critical aspect of natural language processing (NLP), allows for the concise representation of information, making it easier for users to digest large amounts of text quickly. Despite significant advancements in text generation models, challenges such as repetition and coherence persist, particularly in tasks like summarization and machine translation. For instance, Fu et al. [1] highlight the widespread occurrence of repetitive phrases across various text generation applications, underscoring a fundamental issue that detracts from the quality of generated content. Previous studies have attempted to address these challenges through innovative methodologies. Dabre et al. [2] introduced IndicBART, a multilingual sequence-to-sequence model tailored for Indic languages, demonstrating the effectiveness of using orthographic similarities for enhanced performance in both translation and summarization tasks. Furthermore, Xu et al. [3] explored the tendency of neural models to generate repetitive text sequences, elucidating the self-reinforcing nature of these repetitions and proposing effective training techniques to mitigate this issue. Techniques such as the use of pre-trained encoders, as described by Liu and Lapata [4], have been shown to significantly improve summarization outcomes by capturing the underlying semantics of text more effectively. Similarly, the PEGASUS model [5] demonstrates the potential of employing self-supervised objectives tailored for abstractive summarization, achieving state-of-the-art performance across multiple datasets.

This research aims to build upon existing work by implementing post-processing techniques, including repetition avoidance, sentence length control, and the addition of article headlines, to optimize the

Forum for Information Retrieval Evaluation, December 12-15, 2024, India
fathimafiroseofficial@gmail.com (A. F. Firose); rezmasheik@gmail.com (R. Sheik); kathijaafrose@gmail.com (K. A); sjaya@nitt.edu (S. J. Nirmala)
0009-0004-0749-1452 (A. F. Firose); 0000-0003-3567-9757 (R. Sheik); 0000-0001-5251-5845 (S. J. Nirmala)

summarization of news articles across multiple languages. A notable gap exists in the effective summarization of low-resource Indic languages, where current models often struggle to produce coherent and informative summaries due to limited training data and resources. This research addresses that challenge by systematically evaluating the impact of the proposed techniques on the summarization performance of Indic languages. The study seeks to contribute to the growing body of knowledge in multilingual text summarization and provide actionable insights for improving summary quality in underrepresented languages. The findings are expected to demonstrate significant improvements in metrics such as ROUGE and BERTScore, underscoring the importance of post-processing in enhancing the performance of language models in generating informative and coherent summaries.

2. Related Work

Text summarization has evolved dramatically, with various approaches addressing both extractive and abstractive techniques. Extractive methods select important sentences directly from the text, while abstractive methods generate summaries that may rephrase or condense the content. In recent years, transformer-based models like BERT, GPT, and PEGASUS have shown considerable improvements in text summarization. For example, BERT-based models have been used in generating summaries by pre-training on large corpora of documents, improving the fluency and coherence of generated summaries [4]. PEGASUS introduced gap-sentence generation as a novel pre-training strategy for abstractive summarization, enhancing the quality of generated summaries across multiple domains [5]. However, the issue of repetition, particularly in neural models, has gained increasing attention. Fu et al. [1] analyzed the repetition problem and its causes within neural sequence-to-sequence models, providing key insights into this issue. Xu et al. [3]later proposed solutions by incorporating penalties for repetitive tokens, leading to better performance in abstractive summarization. In addition, Nair and Singh [6] discussed techniques for reducing repetition in abstractive summarization, emphasizing the role of linguistic properties in causing repeated phrases.

Moreover, research in multilingual summarization, especially for low-resource languages such as Indic languages, has been relatively sparse. IndicBART, introduced by Dabre et al. [2], addresses this gap by providing a pre-trained model specifically designed for Indic natural language generation. This advancement is crucial, as low-resource languages often suffer from a lack of comprehensive datasets, which affects model performance. The model demonstrated improved summarization capabilities by utilizing the shared characteristics of Indic languages, thereby filling a critical void in the field. Recent work has also focused on eliminating redundancy in multi-document summarization. Mani and Bloedorn [7] introduced a contextual parameter that avoids redundancy by selecting each term in a phrase only once, improving the summary's informativeness. Additionally, Calvo et al. [8] proposed techniques to handle redundancy in multi-document summarization, highlighting the importance of sentence-level diversity in generated summaries.

While substantial progress has been made in improving text summarization, several gaps remain, especially for low-resource languages. Much of the existing research focuses on resource-rich languages like English, which benefit from large training datasets. This creates a performance disparity when models are applied to low-resource languages, such as those found in the Indic language family. Although IndicBART [2] has made strides in this area, there is still a notable lack of research focused on refining summarization techniques for these languages. Indic languages present unique challenges, such as complex syntactic structures and limited annotated data, which are not adequately addressed by current models. Furthermore, repetition in generated summaries remains an ongoing challenge. While studies like those by Fu et al. [1] and Nair and Singh [6] propose methods for reducing repetition, there is still room for improvement, particularly in multilingual contexts. This research aims to bridge these gaps by focusing on advanced post-processing techniques, including repetition avoidance, sentence length control, and headline integration, specifically tailored for low-resource languages like Indic ones. These post-processing techniques, largely underexplored for Indic languages, are critical for improving the fluency, coherence, and overall quality of generated summaries.

The ILSUM shared tasks have significantly contributed to advancing Indian language summarization. The first shared task, introduced in 2022, focused on summarization challenges and methodologies across multiple Indian languages, establishing a benchmark for future research [9, 10]. In 2023, the task expanded with additional datasets and improved evaluation strategies, providing insights into key challenges and solutions for Indian language summarization [11, 12]. These efforts have paved the way for this year's task, which further enhances the dataset with new languages and additional documents.

3. Methodology

3.1. Dataset

The dataset used in this study is derived from the ILSUM 2024 shared task introduced as part of the FIRE 2024 conference. This year's task expanded the dataset to include additional documents for Dravidian languages such as Tamil, Telugu, and Kannada, alongside enhancements for other languages. These new additions provide a richer and more diverse dataset to advance research in Indian language summarization [13], [14]. The datasets are split into training, validation, and testing sets, as shown in Table 1. For Tamil, Telugu, and Kannada, the IndicBARTSS model was fine-tuned, while for English, the T5-Base model was used. These splits align with the dataset structure provided for the ILSUM 2024 task.

Table 1
Dataset Splits for Training, Validation, and Testing

Language	Model	Training Samples	Validation Samples	Test Samples
Tamil	IndicBARTSS	4,104	456	1,955
Telugu	IndicBARTSS	9,583	1,065	4,564
Kannada	IndicBARTSS	10,694	1,188	5,093
English	T5-Base	9,376	1,500	2,500

3.2. Model Architecture

This study primarily employs **IndicBARTSS**, a multilingual, sequence-to-sequence pre-trained model fine-tuned separately for each language. IndicBARTSS is specifically designed for Indic languages and is built on the mBART architecture. It supports 11 Indian languages and offers a more lightweight and computationally efficient alternative compared to larger models like mBART or mT5. IndicBARTSS was trained on an extensive corpus of over 452 million sentences and 9 billion tokens, covering various linguistic nuances. The model can be adapted to tasks such as summarization, machine translation, and question generation.

The fine-tuning process for IndicBARTSS involved using the aforementioned training, validation, and test sets for Tamil, Telugu, and Kannada. Each language was handled separately, using the respective script to avoid any transliteration challenges. The model was optimized to generate summaries that maintain the integrity of the source language while minimizing redundancy and repetition, which is often a challenge in multilingual summarization tasks. For English, the **T5-Base** model was employed, featuring 220 million parameters. This model reframes natural language processing tasks into a unified text-to-text format, where the input and output are always text. The T5-Base model has been applied to a wide range of NLP tasks, including document summarization, machine translation, and question answering, making it a strong candidate for summarization tasks in this study. Both models were optimized using supervised fine-tuning, focusing on minimizing the loss between predicted and ground-truth summaries. The models were evaluated using standard metrics such as ROUGE to assess the quality and faithfulness of the generated summaries.

3.3. Summarization Techniques

The summarization experiments were conducted in four different runs, each utilizing distinct post-processing techniques to enhance the quality and diversity of generated summaries. The same procedures were applied to all the languages in the dataset: Tamil, Telugu, Kannada, and English (the latter utilizing the T5-Base model).

3.3.1. Run 0: Summarization Without Post-Processing

In this configuration, summaries were generated directly from the model without any additional post-processing techniques. The IndicBARTSS model, fine-tuned on each respective language dataset, was employed to generate the summaries. The model's output was kept at a maximum token length of 128 to ensure brevity. This run serves as a baseline to evaluate the intrinsic capabilities of the model in summarizing text without the interference of external constraints. The main goal was to assess how effectively the model can produce coherent, contextually accurate summaries with no extra operations for handling issues like repetition or sentence length control.

3.3.2. Run 1: Post-Processing with Repetition Avoidance

For the first run, a post-processing step was applied to minimize repetition in the generated summaries. Repetition is a known issue in sequence-to-sequence models, especially for tasks involving longer texts. To address this, diverse beam search was incorporated. Specifically, a beam search mechanism with 6 beams was utilized, divided into 3 beam groups, and a diversity penalty of 0.7 was introduced to encourage diversity across beams. Additionally, the `no_repeat_ngram_size` parameter was set to 3, preventing the model from repeating the same trigrams. This configuration aimed to produce more varied and non-repetitive outputs while preserving the meaning of the original articles.

3.3.3. Run 2: Post-Processing with Sentence Length Control

In the second experimental setup, a different post-processing constraint was applied, focusing on controlling the length of the generated summaries. Limiting the summary to two sentences was a key objective. Given that some summaries might be excessively verbose, this post-processing step sought to enhance conciseness while retaining the essential information. The model was modified to terminate early when two sentences were generated, employing a stop condition within the text generation process. The result of this approach was a set of summaries that aligned with length restrictions, providing succinct overviews while maintaining clarity.

3.3.4. Run 3: Heading Integration and Sentence Length Control

In the third run, the heading of the article is integrated during the preprocessing step to enrich the context for the model before generating the summary. This step aims to align the summary more closely with the article's main theme or headline. Additionally, the sentence length control mechanism from Run 2 is retained, ensuring that the summaries are concise and focused on the most important content. The combination of heading integration and sentence length control enables the model to generate more focused summaries that incorporate the article's title as a contextual anchor, while limiting output length to enhance clarity and relevance.

3.4. Evaluation Metrics

The evaluation of summarization models in the conducted experiments employs several established metrics, including **ROUGE** (Recall-Oriented Understudy for Gisting Evaluation) and **BERTScore**. These metrics provide quantitative assessments of the alignment between generated summaries and reference summaries.

ROUGE includes a set of measures that evaluate the quality of generated summaries by comparing them to reference summaries, primarily focusing on n-gram overlap. **ROUGE-1** measures the overlap of unigrams (individual words), **ROUGE-2** assesses bigram (two-word) overlaps, and **ROUGE-L** evaluates the longest common subsequence between the generated and reference summaries. These metrics emphasize recall, precision, and F1 scores, offering insights into the relevance and completeness of the summaries produced by the models. Due to its effectiveness and widespread use, ROUGE remains a key tool for summarization evaluation [15]. **BERTScore** utilizes pre-trained BERT models to assess the quality of generated text. Unlike ROUGE, which relies on exact word matching, BERTScore compares generated and reference texts using contextual embeddings, capturing deeper semantic similarities. This approach is particularly effective for abstractive summarization, where maintaining semantic integrity is important. BERTScore provides precision, recall, and F1 scores, indicating how well the generated summaries convey the intended meaning of the reference texts [16].

4. Experiments

4.1. Experimental Setup

The experiments were conducted using Google Colab, utilizing a T4 GPU with high RAM capacity, which facilitated efficient model training and evaluation. The process began with fine-tuning the IndicBARTSS model and the T5-Base model on the respective datasets, which included Tamil, Telugu, Kannada, and English text. A batch size of 2 was employed during training and inference to balance memory usage and processing speed. This choice allowed the models to handle the complexity of the tasks while ensuring that the GPU resources were utilized effectively. After fine-tuning, the models were saved to the Hugging Face Model Hub, providing an accessible format for further experimentation. Subsequent tests were conducted across four distinct runs, labeled from Run 0 to Run 3, to evaluate the impact of various summarization techniques and post-processing methods on the quality of generated summaries. Each run applied a different approach, ranging from generating summaries without any modifications to integrating post-processing techniques such as repetition avoidance, sentence length control, and the combination of headings with articles.

4.2. Results

The performance of the multilingual summarization system was evaluated across four languages: Tamil, Telugu, Kannada, and English. The evaluation metrics used include ROUGE-1, ROUGE-2, ROUGE-4, and ROUGE-L scores to assess the overlap between system-generated summaries and reference summaries, while BERTScore was employed to gauge the semantic similarity between the summaries. Tables 2and 3 present a detailed breakdown of the results for each language and run.

For Tamil, the results reveal that Run1, which employed post-processing with repetition avoidance, achieved the highest ROUGE scores across all metrics, particularly in ROUGE-1 (0.218) and ROUGE-L (0.2091). This suggests that the repetition avoidance technique applied in this run proved to be the most effective in fine-tuning sentence structure while maintaining content integrity. Although Run2, which applied post-processing with sentence-length control, demonstrated close performance, it fell slightly behind, particularly in ROUGE-4 and ROUGE-L, indicating that while it handled n-gram overlap well, its ability to capture the overall structure of the summaries was slightly reduced. Run3, which focused on heading integration and sentence-length control, showed a notable decline in performance (ROUGE-1 of 0.208), suggesting that this method may have resulted in oversimplified summaries that missed key content aspects (see Table 2). In terms of BERTScore, Run2 emerged as the strongest performer, with an F1 score of 0.729, indicating a slight edge in semantic similarity (see Table 3). This suggests that the combination of sentence simplification with repetition avoidance led to better alignment with reference summaries in terms of meaning. While Run1 followed closely with an F1 score of 0.7267, Run3 lagged behind, confirming the limitations of its post-processing technique for Tamil summaries.

Table 2
Performance Evaluation of Models Across Languages Using ROUGE Metrics

Language	Run Name	ROUGE-1	ROUGE-2	ROUGE-4	ROUGE-L
Tamil	Run1	0.2180	0.1336	0.0905	0.2091
	Run2	0.2164	0.1317	0.0872	0.2078
	Run3	0.2080	0.1177	0.0729	0.1963
Kannada	Run1	0.2284	0.1446	0.1032	0.2218
	Run2	0.2066	0.1275	0.0870	0.1997
	Run3	0.2243	0.1353	0.0927	0.2167
Telugu	Run1	0.3146	0.2318	0.1802	0.3079
	Run2	0.2903	0.2122	0.1633	0.2838
	Run3	0.2510	0.1497	0.0877	0.2420
English	Run1	0.3435	0.1879	0.1357	0.2964
	Run2	0.3217	0.1589	0.1011	0.2705
	Run3	0.3388	0.1702	0.1036	0.2834

Table 3
Performance Evaluation of Models Across Languages Using BERTScore Metrics

Language	Run Name	BERTScore-Precision	BERTScore-Recall	BERTScore-F1
Tamil	Run1	0.7138	0.7418	0.7267
	Run2	0.7301	0.7292	0.7290
	Run3	0.7081	0.7366	0.7212
Kannada	Run1	0.7274	0.7438	0.7349
	Run2	0.7355	0.7234	0.7289
	Run3	0.7263	0.7384	0.7317
Telugu	Run1	0.7488	0.7637	0.7555
	Run2	0.7599	0.7388	0.7486
	Run3	0.7313	0.7275	0.7284
English	Run1	0.8673	0.8802	0.8735
	Run2	0.8659	0.8751	0.8702
	Run3	0.8701	0.8789	0.8742

For Kannada, the results show that Run1 achieved the highest ROUGE scores across all metrics, particularly in ROUGE-1 (0.2284) and ROUGE-L (0.2218). This suggests that the post-processing technique applied in Run1, which focused on repetition avoidance while maintaining content accuracy, was the most effective. Although Run2 showed close performance, it fell slightly behind, particularly in ROUGE-2 and ROUGE-4, indicating that while it controlled sentence length well, it had a reduced ability to capture the overall structure and detail of the summaries. Run3, which also applied sentence-length control, showed a slight decline in performance, with ROUGE-1 at 0.2243 and ROUGE-L at 0.2167, suggesting that stricter sentence-length constraints might have led to summaries with fewer details and a less coherent structure (see Table 2). In terms of BERTScore, Run1 again performed the best, achieving an F1 score of 0.7349, indicating the best alignment with the reference summaries in terms of meaning. Run2 had a close F1 score of 0.7289, while Run3 had a lower F1 score of 0.7317, confirming that limiting sentence length, without focusing on repetition avoidance, reduced the semantic quality of the summaries (see Table 3).

For Telugu, the results show that Run1, which employed post-processing with repetition avoidance, performed the best across all ROUGE metrics, with ROUGE-1 at 0.7488 and ROUGE-L at 0.7555. This indicates that the repetition avoidance technique effectively captured the main content and structure of the summaries. Run2, which applied post-processing with sentence-length control, had slightly lower scores, with ROUGE-L at 0.7486. While it helped manage sentence lengths, it might have slightly affected the overall flow and coherence. Run3, which combined heading integration and sentence-length control, had the lowest scores, with ROUGE-1 at 0.7313 and ROUGE-L at 0.7284, suggesting that this

approach, while structured, led to less detailed summaries (see Table 2). In terms of BERTScore, Run1 again had the highest F1 score (0.7555), showing the best alignment with the reference summaries in terms of meaning. Run2 followed closely with an F1 score of 0.7486, while Run3 had the lowest F1 (0.7284), confirming that the combined heading integration and sentence-length control reduced the semantic quality of the summaries (see Table 3).

For English, the results indicate that Run1 achieved the highest ROUGE scores across all metrics, particularly in ROUGE-1 (0.3435) and ROUGE-L (0.2964). This suggests that the post-processing technique used in Run1, which focused on reducing repetition while maintaining key content, was the most effective for English summaries. Run2 showed slightly lower performance, particularly in ROUGE-2 and ROUGE-4, with ROUGE-2 at 0.1589 and ROUGE-4 at 0.1011, indicating that although it controlled sentence length effectively, it slightly compromised the capture of n-gram overlap and summary structure. Run3, which also applied sentence-length control, showed a slight decline in performance (ROUGE-1 of 0.3388 and ROUGE-L of 0.2834), suggesting that stricter length constraints led to less detailed and less coherent summaries (see Table 2). In terms of BERTScore, Run1 again emerged as the strongest performer with an F1 score of 0.8735, showing the best alignment with the reference summaries in terms of meaning. Run3 closely followed with an F1 score of 0.8742, while Run2 had a slightly lower F1 score of 0.8702, confirming that sentence-length control with repetition avoidance yielded the best semantic similarity (see Table 3).

5. Discussion

5.1. Implications

The results of our multilingual summarization task provide several key implications for future research and applications in text summarization, particularly in the context of Indic languages. The findings indicate that post-processing techniques significantly influence the quality of generated summaries, with varying effectiveness across languages. For example, in Tamil and Telugu, balancing sentence restructuring and repetition avoidance proved most beneficial, leading to higher ROUGE scores and improved semantic coherence. This suggests that language-specific fine-tuning of post-processing methods is critical, as general approaches may not capture the unique linguistic nuances inherent in different languages.

In the case of Kannada and English, techniques that prioritized simplicity without oversimplification produced better results, emphasizing that reducing sentence complexity while maintaining critical content is crucial. For future research, these insights highlight the need to develop more adaptive post-processing techniques that can dynamically adjust to the linguistic and structural requirements of the target language. Additionally, the strong performance of certain runs in terms of BERTScore points to the potential of exploring semantic-focused evaluation metrics further, which could complement traditional n-gram-based evaluations like ROUGE in producing more meaningful and human-readable summaries. The findings have practical implications for real-world applications, especially for industries or platforms relying on automated summarization of multilingual content, such as news articles, governmental reports, and legal documents. The ability to fine-tune summarization systems for specific languages will enhance the reliability and readability of summaries in low-resource languages like Tamil, Telugu, and Kannada, making it feasible to deploy these systems in diverse linguistic environments.

5.2. Limitations

While the study offers valuable insights into post-processing techniques for multilingual summarization, certain limitations should be acknowledged. First, the evaluation was limited to four languages: Tamil, Telugu, Kannada, and English, which may constrain the generalizability of the findings to other Indic or non-Indic languages. Different languages, particularly those with complex grammatical structures or varying sentence patterns, may require further adjustments to the post-processing methods tested here. As a result, our approach may not be universally applicable without modifications for these

other languages. Another limitation concerns the dataset size and domain. The training and validation data for each language were limited to specific text domains, which may impact the summarization system's generalizability across various content types, such as technical documents, literary texts, or conversational content. Future research could address this by incorporating more diverse datasets to u whether the post-processing strategies remain effective across broader contexts.

Moreover, while we focused on ROUGE and BERTScore metrics for evaluation, these metrics, though widely used, may not capture all aspects of summary quality. ROUGE primarily focuses on n-gram overlap, which might overlook summaries that are semantically equivalent but phrased differently. Similarly, BERTScore focuses on semantic similarity but may not always correlate with human judgment of summary readability or informativeness. Expanding the evaluation criteria to include human assessment of summaries could provide a more comprehensive understanding of the summarization system's performance and practical utility. Finally, computational constraints posed a limitation in terms of the number of post-processing variations tested. Future research could explore a broader range of techniques, such as sentence fusion or controlled generation, to further enhance the quality of the generated summaries.

6. Conclusion

This study investigated the influence of various post-processing techniques on the performance of multilingual summarization systems, specifically focusing on Tamil, Telugu, Kannada, and English datasets. The results reveal that the effectiveness of summarization strategies is markedly contingent upon the specific language and the post-processing methodologies employed. Notably, a balanced approach that integrates sentence restructuring with repetition avoidance yielded optimal outcomes for Tamil and Telugu, as evidenced by superior ROUGE and BERTScore metrics. In contrast, the simplification of sentence structure while preserving critical content emerged as the most effective strategy for Kannada and English. These findings highlight the necessity for tailored, language-specific approaches in multilingual summarization tasks, underscoring the importance of adapting techniques to accommodate linguistic and structural peculiarities inherent in each language.

Future research should focus on developing more flexible post-processing techniques that can adapt to the unique characteristics of different languages. Expanding the study to include more languages, especially those that are less studied, will help determine how well these techniques work across various linguistic contexts. Additionally, incorporating human evaluations alongside traditional metrics like ROUGE and BERTScore could provide deeper insights into the quality of the generated summaries. Further exploration of cross-lingual summarization, where summaries are created in one language based on content in another, also presents an interesting direction for future studies.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] Z. Fu, W. Lam, A. M.-C. So, B. Shi, A theoretical analysis of the repetition problem in text generation, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, 2021, pp. 12848–12856.

[2] R. Dabre, H. Shrotriya, A. Kunchukuttan, R. Puduppully, M. M. Khapra, P. Kumar, Indicbart: A pre-trained model for indic natural language generation, arXiv preprint arXiv:2109.02903 (2021).

[3] J. Xu, X. Liu, J. Yan, D. Cai, H. Li, J. Li, Learning to break the loop: Analyzing and mitigating repetitions for neural text generation, Advances in Neural Information Processing Systems 35 (2022) 3082–3095.

[4] Y. Liu, M. Lapata, Text summarization with pretrained encoders, arXiv preprint arXiv:1908.08345 (2019).
[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] P. Nair, A. K. Singh, On reducing repetition in abstractive summarization, in: Proceedings of the Student Research Workshop Associated with RANLP 2021, 2021, pp. 126–134.
[7] I. Mani, Summarizing similarities and differences among related document, Advances in Automatic Text Summarization (1999).
[8] H. Calvo, P. Carrillo-Mendoza, A. Gelbukh, On redundancy in multi-document summarization, Journal of Intelligent & Fuzzy Systems 34 (2018) 3245–3255.
[9] S. Satapara, P. Mehta, S. Modha, D. Ganguly, Indian language summarization at FIRE 2023, in: D. Ganguly, S. Majumdar, B. Mitra, P. Gupta, S. Gangopadhyay, P. Majumder (Eds.), Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2023, Panjim, India, December 15-18, 2023, ACM, 2023, pp. 27–29. URL: https://doi.org/10.1145/3632754.3634662. doi:10.1145/3632754.3634662.
[10] S. Satapara, P. Mehta, S. Modha, A. Hegde, S. HL, D. Ganguly, Overview of the third shared task on indian language summarization (ilsum 2024), in: K. Ghosh, T. Mandl, P. Majumder, D. Ganguly (Eds.), Working Notes of FIRE 2024 - Forum for Information Retrieval Evaluation, Gandhinagar, India. December 12-15, 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2024.
[11] S. Satapara, P. Mehta, S. Modha, A. Hegde, S. HL, D. Ganguly, Key insights from the third ilsum track at fire 2024, in: Proceedings of the 16th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2024, Gandhiinagar, India. December 12-15, 2024, ACM, 2024.
[12] S. Satapara, P. Mehta, D. Ganguly, S. Modha, Fighting fire with fire: Adversarial prompting to generate a misinformation detection dataset, CoRR abs/2401.04481 (2024). URL: https://doi.org/10.48550/arXiv.2401.04481. doi:10.48550/ARXIV.2401.04481. arXiv:2401.04481.
[13] S. Satapara, B. Modha, S. Modha, P. Mehta, FIRE 2022 ILSUM track: Indian language summarization, in: D. Ganguly, S. Gangopadhyay, M. Mitra, P. Majumder (Eds.), Proceedings of the 14th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE 2022, Kolkata, India, December 9-13, 2022, ACM, 2022, pp. 8–11. URL: https://doi.org/10.1145/3574318.3574328. doi:10.1145/3574318.3574328.
[14] S. Satapara, P. Mehta, S. Modha, D. Ganguly, Key takeaways from the second shared task on indian language summarization (ILSUM 2023), in: K. Ghosh, T. Mandl, P. Majumder, M. Mitra (Eds.), Working Notes of FIRE 2023 - Forum for Information Retrieval Evaluation (FIRE-WN 2023), Goa, India, December 15-18, 2023, volume 3681 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 724–733. URL: https://ceur-ws.org/Vol-3681/T8-1.pdf.
[15] C.-Y. Lin, Rouge: A package for automatic evaluation of summaries, in: Text summarization branches out, 2004, pp. 74–81.
[16] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, Bertscore: Evaluating text generation with bert, arXiv preprint arXiv:1904.09675 (2019).