Named Entity-Aware Abstractive Text Summarization for Hindi Language

Saumay Gupta, Sukomal Pal

Indian Institute of Technology (BHU), Varanasi, India

Abstract

In this study, we introduce a novel approach to text summarization, specifically tailored for the Hindi language, titled Named Entity-Aware Abstractive Text Summarization (NEA-ATS). Our methodology uniquely integrates Named Entity Recognition with advanced pretrained language models, focusing on critical entities such as individuals, locations, and organizations. We use our proposed methodology along with the pretrained models to work on the ILSUM task to provide summaries for Hindi news articles. We secured the first rank for the Hindi summarization task. Our comprehensive evaluation offers valuable insights into enhancing the NEA-ATS methodology in the future, along with determining efficient methods and model for Hindi summarization.

Keywords

Indian language summarization, Named entity aware summarization, Pretrained models, NER

1. Introduction

In the age of information abundance, abstractive text summarization has emerged as a vital tool to distill vast amounts of textual content efficiently. Unlike extractive summarization that pulls exact sentences from the original text, abstractive summarization entails creating fresh sentences to encapsulate the core ideas of the source, demanding an in-depth understanding of its semantics and context[1].

The complexities of the Hindi language, including its intricate syntax, rich morphology, and prevalent code-mixing, present unique challenges for summarization algorithms. To address these, we propose an innovative Named Entity-Aware Abstractive Text Summarization (NEA-ATS) technique for Hindi. Leveraging the prowess of pretrained language models (PLMs) like mBART-50, mT5, and IndicBART, which excel in capturing semantic and contextual nuances across languages, our NEA-ATS method employs named entity recognition (NER) to enhance the substance and clarity of summaries[2]. By integrating the latest Hindi NER model, HiNER-original-muril-base-cased, our system can identify and categorize pivotal entities such as individuals, locations, and organizations, ensuring that summaries focus on salient details while preserving the original text's coherence and essence.[3]

The incorporation of NER is particularly vital given the tendency of state-of-the-art abstractive summarizers to omit or incorrectly substitute named entities, which can result in misleading summaries and the spread of misinformation[3]. Our method addresses these issues

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

saumay.gupta.cse19@itbhu.ac.in (S. Gupta); spal.cse@itbhu.ac.in (S. Pal)

CEUR Workshop Proceedings (CEUR-WS.org)

by initially training the summarization model on the NER task to enhance entity awareness, thereby reducing hallucinations and inaccuracies in the final summary.

The ILSUM 2023 shared task [4][5], centered on creating state-of-the-art models for generating meaningful summaries for new articles, particularly in Indian languages, provides participants with a collection of news articles and their summaries. This dataset encompasses English and three major Indian languages: Hindi, Gujarati, and Bengali.

The first section gives us an introduction of the whole paper, Section 2 describes some recent works related to the fields of summarization and NER. Section 3 briefly explains the task and describes the dataset we are working on. Section 4 explains our NEA-ATS methodology along with the PLMs used for our task. While, Section 5 and 6 describe our experimentation method, the results obtained and the insights we got from our experiment. Section 7 concludes our work.

2. Related work

In this section, we describe some recent works and advancements done in the area of Hindi language summarization and named-entity recognition.

Text summarization is a constant growing field of research and recently summarization for Indian languages, particularly Hindi has picked up a lot of pace. As summarization can either be extractive or abstractive, lots of works can be found that were done in this field.

Chestha et al. in their work [6] present a novel approach where both extractive and abstractive summarization techniques are proposed and compared for Hindi text documents. The study introduces a ward hierarchical agglomerative clustering method. The comparison between the extractive and abstractive methods provides valuable insights into the effectiveness and applicability of these techniques in processing Hindi text documents.

Kumar et al. [7] introduce a Generative Adversarial Network (GAN)-based model for abstractive text summarization. The model is composed of a generator and two discriminators, where the Similarity Discriminator ensures that the generated summary maintains a high degree of similarity with the original text. This innovative approach addresses the challenge of maintaining coherence and relevance in generated summaries.

The work [8] by Rishab et al. tackles the challenge of limited datasets in Indian regional languages, particularly Hindi. They introduce two innovative deep learning models for text summarization, following an abstractive methodology. These models utilize attention mechanisms and are built upon a Stacked LSTM Sequence To Sequence (Seq2Seq) framework. The research is noteworthy for its emphasis on regional linguistic diversity and for addressing the issue of dataset shortages in this field.

Richa et al. [9] explore the Named Entity Recognition (NER) process in detail, emphasizing its importance in Information Extraction. It describes a dual-phase procedure for NER, involving the detection and categorization of named entities into established groups. These groups encompass a variety of types, including individuals, locations, organizations, numeric expressions, temporal expressions, among others.. The use of neural language models and Conditional Random Fields (CRF) for Hindi language NER signifies a notable advancement in applying machine learning techniques to Indian languages.

The recent rise in Indic language datasets like XL-Sum [10], WikiLingua [11], MassiveSumm [12] and many others has also helped create Indic language summarization become an active area of research.

3. Task Description

Our task involves training a model and creating concise summaries for articles for the Hindi language, which can be either extractive or abstractive. The dataset employed in this study consists of article-headline pairs, which have been collected from several leading newspapers across the country. The dataset provided covers English and major Indian languages such as Hindi, Gujarati and Bengali. We use the combined Hindi dataset (ILSUM 2022[13] and 2023[4][5]) for our work and describe its structure below. Notably, while previous studies in other languages have utilized news article-headline pairs, the dataset presents a unique challenge due to the presence of code-mixing (mixing multiple different languages) and script mixing. Very few works have been done till now to summarization code-mixed Hindi articles.

Some examples of such code-mixing and script mixing in both headlines and articles, are illustrated below:

- 31 दिसंबर 2022 तक SBFC फाइनेंस 16 राज्यों और दो केंद्र शासित प्रदेशों के 105 से अधिक शहरों में मौजूद है।
- 1957 के DMC एक्ट यानी Delhi Municipal Corporation Act के अनुसार, DMC एक्ट के अनुसार, कुत्ते का रजिस्ट्रेशन एक साल के लिए मान्य होता है।

The dataset was checked on various factors mentioned in [14] and checks were done on cases such as:

1. Empty records
2. Duplicated Entries
3. Classification of summaries into extractive, semi-extractive, and abstractive categories.
4. Identification of article-summary pair where the first sentence of the article is the same as the summary.
5. Evaluation of summary size to ensure it does not resemble the length of the actual article.

Of the 21,225 records, one was found to contain an empty article composed solely of a newline character, which was subsequently removed from the dataset. No duplicated entries were detected. The distribution of summary types is presented in Table 1, with summaries categorized as extractive if they precisely match sentences from the article, abstractive if no sentence matches, and semi-extractive if some sentences match and some do not. Remarkably, approximately 80% of the summaries fell into the abstractive or semi-extractive categories. Consequently, the decision was made to employ abstractive summarization techniques for the dataset.

Descriptive statistics for the dataset are provided in Table 2 and Table 3. Multiple tokenizers, IndicNLP[15], AlbertTokenizer[16], T5Tokenizer[17] and mBart50Tokenizer[18] were employed for dataset analysis, yielding valuable insights into the dataset and tokenization processes. Additionally, these insights guided the selection of hyperparameters for the models used to train the dataset.

Table 1
Dataset check statistics

Check	Count	Percentage
Empty records	0	0
Duplicate records	0	0
Extractive summary	3515	16.6
Semi-extractive summary	7055	33.2
Abstractive summary	10654	50.2
First sentence same	1352	6.4

Table 2
Dataset count statistics - words

Parameters	IndicNLP		AlbertTokenizer		T5Tokenizer		mBart50Tokenizer	
Pair Count	21224							
	Text	Summary	Text	Summary	Text	Summary	Text	Summary
Average words	646	43	579	53	999	78	630	58
Min words	17	6	25	13	33	21	29	15
Max words	5034	113	1024	138	2048	207	1024	145

Table 3
Dataset count statistics - sentences

Parameters	Text	Summary
Average sentences	29	2
Min sentences	1	1
Max sentences	282	10

4. Methodology

In news article summarization, two key approaches are extractive and abstractive summarization. In our experiment, we use the abstractive method of summarization to provide summaries for the Hindi news article. We also experiment with providing named-entity attention to the model to improve the summaries. Firstly, we describe the models we employed for the purpose of text summarization and NER and then we explain the methodology we experimented with to provide named-entity aware summaries.

4.1. Models used

In this section, we describe the various PLMs used for our task of named-entity aware Hindi summarization. PLMs, particularly deep transformer-based ones, have been instrumental in advancing abstractive text summarization. These models, equipped with extensive knowledge and vast parameters, have led to significant progress in the field of Natural Language Processing (NLP). This progress has empowered text summarization by enabling the generation of summaries that closely resemble human-authored content.[19]

- **mBART-50**[18] is a multilingual Seq2Seq model that undergoes pre-training with the 'Multilingual Denoising Pretraining' objective. This model demonstrates the feasibility of developing multilingual translation models via multilingual fine-tuning. Unlike conventional fine-tuning in a single direction, this approach fine-tunes a pretrained model across multiple directions simultaneously. mBART-50 extends the capabilities of the original mBART model and spanning a total of 50 languages. We used the mBART-Large-50 model having 610M parameters for fine-tuning.
- **mT5**[17] a multilingual version of T5. This model has been pretrained on a Common Crawl-based dataset that has 101 languages. It operates within a unified 'text-to-text' framework, making it exceptionally versatile for various language tasks. Renowned for its exceptional performance in multilingual applications, mT5 stands out for its capacity to understand and generate text in numerous languages, making it a valuable tool for a wide range of language-related tasks. We use only the mT5-base version of the model having 580M parameters. Due to memory constraints, we were unable to use the mT5 large version which had 1.2B parameters.
- **IndicBART**[16], a multilingual sequence-to-sequence pretrained model, prioritizes Indic languages and English. It accommodates 11 Indian languages, leveraging the mBART architecture and orthographic similarities among Indic scripts for improved transfer learning. Notably, its smaller size(244M) compared to models like mBART and mT5(-base) makes it a computationally efficient choice for fine-tuning and decoding tasks. Alongside this model we also used the **IndiBARTSS**[16] model which was specifically pretrained for short summarization tasks.

In addition to the above models which provide Hindi text summarization we use another PLM which provides us with the state-of-the-art ability of named entity recognition.

- **HiNER-original-muril-base-cased**[20] is a MuRIL based model fine-tuned on the NER dataset HiNER (Hindi Named Entity Recognition) [20]. The dataset was compiled from diverse government information webpages, and involved manual annotation of these sentences. It comprises sentences extracted from ILCI and various other sources. This model was used to get the named entities in an article.

4.2. Named Entity-Aware Summarization

NER is a task in NLP that focuses on detecting and categorizing essential information (entities) within a text into predetermined classes, including people's names, organizations, places, dates, and more. NER systems are designed to locate named entities in a body of text and classify them according to a fixed set of categories, providing a way to extract structured information from unstructured text sources.

In the domain of abstractive summarization for news articles, the inclusion of NER emerges as a crucial factor in enhancing the quality and depth of generated summaries. NER allows the model to identify and prioritize key entities such as individuals, locations, and organizations within the text. This strategic integration empowers the summarization process not only to distill information but also to furnish context by highlighting vital elements. The motivation behind incorporating NER lies in the ambition to create summaries that are both concise

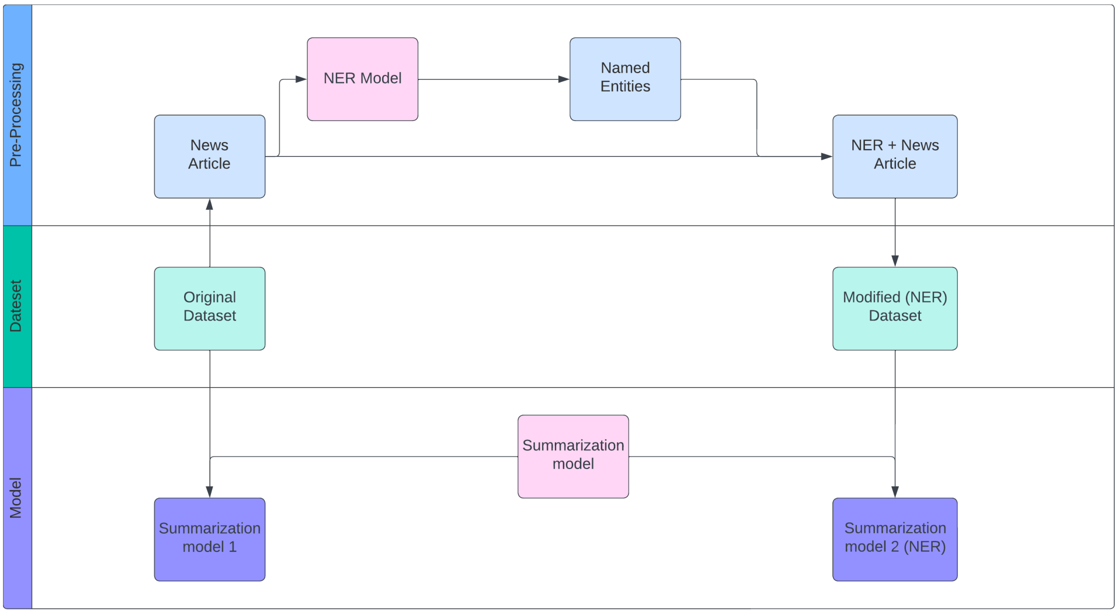

Figure 1: Methodology Flowchart

and enriched with essential details, ensuring coherence and capturing critical contextual nuances from the source news articles. The NER-augmented abstractive summarization approach aims to produce reader-friendly and comprehensive summaries, encapsulating the fundamental essence of news stories while underscoring significant entities.

We experiment with the NEA-ATS method where the named entities are appended to the article to provide attention to named entities along with the string 'entity: ' to show that the sentence is a named entity sentence. Let us consider the set of all articles as X, for an input article $x \in X$, we use an NER model $N(x)$ to get a set of unique named entities $E = \{e_i \mid e_i \in N(x), i \in N\}$ (article is constrained by the maximum tokens allowed for the summarization model). We then create a sentence using the named entities and append it to the start of the article creating the modified article x' given by $x' = \{entity : e_1, e_2, e_3...., e_n | x \mid e_i \in E\}$. These modified articles are then fed to the summarization model along with the summary y to fine-tune the model. Adding the named entities in the start of the article makes the model focus more on named entities as they are considered to be of higher importance (first input to the model). This process can be visible in Fig. 1.

An example of applying the above method and getting the modified article is (starting sentences of the article are shown):

- **Original:** केरल के एर्नाकुलम जिले में 5 साल की बच्ची से रेप के बाद गला दबाकर हत्या कर दी गई। आरोपी ने बच्ची का शव बोरे में डालकर डंपिंग ग्राउंड में फेंक दिया था। पुलिस ने आरोपी शख्स को गिरफ्तार कर लिया है। घटना शुक्रवार शाम की है। पुलिस ने शनिवार को मीडिया को इसकी जानकारी दी।
- **Modified Article:** entity: केरल, एर्नाकुलम, 5 साल, शुक्रवार शाम, शनिवार, विवेक कुमार, रात 9. 30 बजे, बिहार, सुबह, कांग्रेस, विधानसभा, वीडी सतीशन, सुधाकरण। केरल के एर्नाकुलम जिले

Table 4
Training parameters for summarization models

Parameters	IndicBART	IndicBARTSS	mBART-50	mT5
Max Source Length	1024	1024	1024	1024
Max Target Length	75	75	75	100
Max Epochs	10	10	10	10
Batch size	4	4	16	16
Vocab Size	64014	64015	250054	250112

> में 5 साल की बच्ची से रेप के बाद गला दबाकर हत्या कर दी गई। आरोपी ने बच्ची का शव बोरे में डालकर डंपिंग ग्राउंड में फेंक दिया था। पुलिस ने आरोपी शख्स को गिरफ्तार कर लिया है। घटना शुक्रवार शाम की है। पुलिस ने शनिवार को मीडिया को इसकी जानकारी दी।

5. Experiments

As we only had training dataset for our experiment, we split the dataset randomly into 5 folds. This splitting is done so in a sense that we utilize 4 folds (80%) of the split for training and the rest (20%) for validation. In the k-fold cross validation process, the training and validation was done 5 times, where in each fold, a different fold is picked for validation and the rest for training. As we did not have access to the validation dataset, it was necessary to split the training dataset as k-fold cross validation helps us in choosing a good split along with generalizing the model.

We use the PyTorch and HuggingFace libraries to train and test our models, where we trained the models for a maximum of 10 epochs and selected the epoch giving the best validation metrics along with checking for overfitting and underfitting. We used the ROUGE[21] metrics (namely: ROUGE-1, ROUGE-2 and ROUGE-L) to compute the scores and evaluate the models. As the standard ROUGE module does not support Indian languages, we use the Multilingual ROUGE module[10] for our work.

The named entities are extracted from the source articles using the HINER-original-muril-base-cased[20] pretrained model, while we use four different models, mBART-50[18], mT5[17], IndicBART[16] and IndicBARTSS[16] to fine-tune on the task of summarization. Each of these four different models are trained on both the original dataset and the modified dataset having named entities along with 5-fold cross validation. The model versions giving best validation results were picked to provide the summaries for the test dataset. Hyperparameters for the models were initially selected by observing the trends in the dataset and how the models worked internally. These were more finely tuned when we started training the models. The final parameters used to train the models are given in Table 4. The default learning rate of $5e-5$ was used to train all the models, while a beam size of 4 was taken to generate the predictions.

6. Results and Discussion

The validation metrics for each fold for all the models are given in Table 5 to Table 8. Each time the model was separately fine-tuned from scratch for each fold for both the named-entity

Table 5
Validation metrics for IndicBART

Fold	Model Type	ROUGE-1	ROUGE-2	ROUGE-L
Fold 1	Original	**57.6**	**48.8**	**54.2**
	Named entity aware	57.2	48.5	53.9
Fold 2	Original	56.3	47.5	53.0
	Named entity aware	55.7	47.0	52.4
Fold 3	Original	57.2	48.4	53.9
	Named entity aware	56.0	47.2	52.6
Fold 4	Original	56.4	47.6	52.9
	Named entity aware	55.6	46.6	52.1
Fold 5	Original	56.7	47.7	53.2
	Named entity aware	56.4	47.5	52.9

Table 6
Validation metrics for IndicBARTSS

Fold	Model Type	ROUGE-1	ROUGE-2	ROUGE-L
Fold 1	Original	**57.3**	**48.6**	**53.9**
	Named entity aware	56.3	47.7	52.9
Fold 2	Original	55.4	46.5	52.0
	Named entity aware	55.2	46.4	51.7
Fold 3	Original	56.4	47.6	52.9
	Named entity aware	55.5	46.7	52.1
Fold 4	Original	56.1	47.3	52.7
	Named entity aware	55.0	46.2	51.5
Fold 5	Original	56.7	47.8	53.3
	Named entity aware	55.5	46.4	51.9

Table 7
Validation metrics for mBART-50

Fold	Model Type	ROUGE-1	ROUGE-2	ROUGE-L
Fold 1	Original	**58.0**	**48.5**	**54.6**
	Named entity aware	57.0	46.8	53.4
Fold 2	Original	57.0	47.5	53.5
	Named entity aware	56.4	47.0	53.0

based and original articles. The validation results were compared and the top 5 performing models were chosen to generate the summary (limited to 75 or 100 tokens) for the test dataset. Some examples of the generated summaries are:

- Aam Aadmi Party (AAP) MP Raghav Chadha Controversy – संसद परिसर में मंगलवार को आम आदमी पार्टी (AAP) सांसद राघव चड्ढा के ऊपर कौआ बैठ गया। राघव उस समय फोन पर बात कर रहे थे। वे मानसून सत्र से वापस लौट रहे थे, तभी ये घटना हुई

Table 8
Validation metrics for mT5

Fold	Model Type	ROUGE-1	ROUGE-2	ROUGE-L
Fold 1	Named entity aware	**59.0**	**50.5**	**55.6**

Table 9
ILSUM Test metrics

Model	Model Type	R-1	R-2	R-4	R-L	BERT-p	BERT-r	BERT-F1
mBart-50	Original	**56.25**	**47.15**	**40.32**	**53.73**	**82.26**	80.48	**81.3**
mT5	Named entity aware	54.09	45.92	40.0	51.53	79.7	**80.73**	80.17
IndicBART	Original	53.59	45.51	39.73	51.28	80.85	79.48	80.08
IndicBARTSS	Original	53.28	44.96	39.12	50.84	80.05	80.03	79.98
IndicBART	Named entity aware	29.88	17.07	11.96	24.76	71.53	70.37	70.89

- Mumbai Indians Vs Royal Challengers Bangalore (MI) Vs Royal Challengers Bangalore (RCB) 2023 – मुंबई इंडियंस ने विमेंस प्रीमियर लीग में अपना आखिरी लीग मैच आसानी से जीत लिया है। टीम ने रॉयल चैलेंजर्स बेंगलुरु को 4 विकेट से हराया। मुंबई के डीवाई पाटिल स्टेडियम में मिली इस जी

The models chosen to generate test summary and their evaluation metrics are given in Table 9. BERT score(precision, recall and F1) was used along with the ROUGE scores for evalution.

The evaluation findings indicate that the named entity aware mT5 model yields superior outcomes, surpassing the original mBART-50 model which further outperforms other models. The test metrics given in Table 9 also corroborate this claim, where the named entity aware mT5 model outperforms original mBART-50 on BERT scores, although it performs slightly poorly based on the ROUGE scores. This observation suggests that models like mT5 and mBART-50, which are fine-tuned on base models such as T5 and BART, are more effective than others like IndicBART and IndicBARTSS. However, it's noteworthy that named entity aware models show diminished performance relative to their standard counterparts. This implies that while focusing on named entities, the addition of named entity sentences in front of articles may disrupt the text's semantic integrity, leading to lower metric scores. This trend is further evident in the test results presented in Table 9, where the original mBART-50 model surpasses the named entity aware mT5 model for the ROUGE score albeit slightly.

7. Conclusion

This paper detailed the creation of the NEA-ATS method specifically tailored for Hindi. This method addressed the complexities of Hindi's syntax and morphology, as well as the challenges posed by code-mixing. By incorporating advanced pretrained language models, such as mBART-50[18], mT5[17], and IndicBART[16], the study emphasized the importance of named entity recognition in improving the accuracy and relevance of the summaries.

The approach involved refining summarization models to prioritize named entities, ensuring that generated summaries were focused on key information. Extensive testing, including experiments with datasets in Hindi, illustrated NEA-ATS method's capability in producing detailed and context-aware summaries. Although, integration of named entities sometimes affected textual flow, these instances offered valuable insights for future enhancements. To address this issue, future research could explore alternative methods of entity attention that do not compromise the semantic coherence of text. For ILSUM task [4][5], original mBART-50 and named entity aware mT5 models outperformed other original and named entity aware models.

In summary, this research marks an important advancement in text summarization for Indian languages, especially Hindi. It underscores the critical role of named entity recognition in abstractive summarization and sets a foundation for future explorations in this area.

References

[1] M. Tank, P. Thakkar, Abstractive text summarization using adversarial learning and deep neural network, Multimedia Tools and Applications (2023). URL: https://doi.org/10.1007/s11042-023-17478-0. doi:10.1007/s11042-023-17478-0.

[2] D. Suleiman, A. Awajan, Deep learning based abstractive text summarization: Approaches, datasets, evaluation measures, and challenges, Mathematical Problems in Engineering 2020 (2020) 9365340. URL: https://doi.org/10.1155/2020/9365340. doi:10.1155/2020/9365340.

[3] S. Berezin, T. Batura, Named entity inclusion in abstractive text summarization, in: Third Workshop on Scholarly Document Processing, 2022, p. 158.

[4] S. Satapara, P. Mehta, S. Modha, 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.

[5] 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, Goa, India. December 15-18, 2023, CEUR Workshop Proceedings, CEUR-WS.org, 2023.

[6] C. Kwatra, K. Gupta, Extractive and abstractive summarization for hindi text using hierarchical clustering, in: 2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), 2021, pp. 1–6. doi:10.1109/ICSES52305.2021.9633789.

[7] S. Kumar, A. Solanki, An abstractive text summarization technique using transformer model with self-attention mechanism, Neural Computing and Applications 35 (2023) 18603–18622. URL: https://doi.org/10.1007/s00521-023-08687-7. doi:10.1007/s00521-023-08687-7.

[8] R. Karmakar, K. Nirantar, P. Kurunkar, P. Hiremath, D. Chaudhari, Indian regional language abstractive text summarization using attention-based lstm neural network, in: 2021 International Conference on Intelligent Technologies (CONIT), 2021, pp. 1–8. doi:10.1109/CONIT51480.2021.9498309.

[9] R. Sharma, S. Morwal, B. Agarwal, Named entity recognition using neural language

model and crf for hindi language, Computer Speech Language 74 (2022) 101356. URL: https://www.sciencedirect.com/science/article/pii/S0885230822000055. doi:https://doi.org/10.1016/j.csl.2022.101356.

[10] T. Hasan, A. Bhattacharjee, M. S. Islam, K. Mubasshir, Y.-F. Li, Y.-B. Kang, M. S. Rahman, R. Shahriyar, XL-sum: Large-scale multilingual abstractive summarization for 44 languages, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp. 4693–4703. URL: https://aclanthology.org/2021.findings-acl.413.

[11] C. C. Faisal Ladhak, Esin Durmus, K. McKeown, Wikilingua: A new benchmark dataset for multilingual abstractive summarization, in: Findings of EMNLP, 2020, 2020.

[12] D. Varab, N. Schluter, MassiveSumm: a very large-scale, very multilingual, news summarisation dataset, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 10150–10161. URL: https://aclanthology.org/2021.emnlp-main.797.

[13] S. Satapara, B. Modha, S. Modha, P. Mehta, Fire 2022 ilsum track: Indian language summarization, in: Proceedings of the 14th Annual Meeting of the Forum for Information Retrieval Evaluation, FIRE '22, Association for Computing Machinery, New York, NY, USA, 2023, p. 8–11. URL: https://doi.org/10.1145/3574318.3574328. doi:10.1145/3574318.3574328.

[14] A. Urlana, S. M. Bhatt, N. Surange, M. Shrivastava, Indian language summarization using pretrained sequence-to-sequence models (2022).

[15] A. Kunchukuttan, The IndicNLP Library, https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf, 2020.

[16] R. Dabre, H. Shrotriya, A. Kunchukuttan, R. Puduppully, M. M. Khapra, P. Kumar, Indicbart: A pre-trained model for indic natural language generation, in: Findings of the Association for Computational Linguistics: ACL 2022, 2022, pp. 1849–1863.

[17] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, C. Raffel, mt5: A massively multilingual pre-trained text-to-text transformer, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 483–498.

[18] Y. Tang, C. Tran, X. Li, P.-J. Chen, N. Goyal, V. Chaudhary, J. Gu, A. Fan, Multilingual translation with extensible multilingual pretraining and finetuning, arXiv preprint arXiv:2008.00401 (2020).

[19] A. A. Syed, F. L. Gaol, A. Boediman, T. Matsuo, W. Budiharto, A survey of abstractive text summarization utilising pretrained language models, in: N. T. Nguyen, T. K. Tran, U. Tukayev, T.-P. Hong, B. Trawiński, E. Szczerbicki (Eds.), Intelligent Information and Database Systems, Springer International Publishing, Cham, 2022, pp. 532–544.

[20] R. Murthy, P. Bhattacharjee, R. Sharnagat, J. Khatri, D. Kanojia, P. Bhattacharyya, Hiner: A large hindi named entity recognition dataset, in: Proceedings of the Thirteenth Language Resources and Evaluation Conference, 2022, pp. 4467–4476.

[21] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81. URL: https://aclanthology.org/W04-1013.