Mukhyansh: A Headline Generation Dataset for Indic Languages

Lokesh Madasu*, Gopichand Kanumolu*, Nirmal Surange*, Manish Shrivastava
Language Technologies Research Center, KCIS, IIIT Hyderabad, India.
{lokesh.madasu, gopichand.kanumolu, nirmal.surange}@research.iiit.ac.in
m.shrivastava@iiit.ac.in

Abstract

The task of headline generation within the realm of Natural Language Processing (NLP) holds immense significance, as it strives to distill the true essence of textual content into concise and attention-grabbing summaries. While noteworthy progress has been made in headline generation for widely spoken languages like English, there persist numerous challenges when it comes to generating headlines in low-resource languages, such as the rich and diverse Indian languages. A prominent obstacle that specifically hinders headline generation in Indian languages is the scarcity of high-quality annotated data. To address this crucial gap, we proudly present Mukhyansh, an extensive multilingual dataset, tailored for Indian language headline generation. Comprising an impressive collection of over 3.39 million article-headline pairs, Mukhyansh spans across eight prominent Indian languages, namely Telugu, Tamil, Kannada, Malayalam, Hindi, Bengali, Marathi, and Gujarati. We present a comprehensive evaluation of several state-of-the-art baseline models. Additionally, through an empirical analysis of existing works, we demonstrate that Mukhyansh outperforms all other models, achieving an impressive average ROUGE-L score of 31.43 across all 8 languages.

1 Introduction

Headline generation plays a crucial role in summarizing news articles and capturing readers' attention. The task of headline generation involves automatically generating informative and captivating headlines that accurately capture the essence of the underlying text. Headline generation is challenging due to two major factors: firstly, headlines must accurately represent the content of the text while being concise. This requires a fine balance between capturing the key information and maintaining brevity. Secondly, headlines often need to be attention-grabbing, compelling readers to click and read further. This necessitates the use of persuasive language, creativity, and an understanding of rhetorical devices.

In recent years, the NLP community has achieved remarkable strides in the development of headline-generation models. However, the focus has primarily been on English and other widely spoken languages, inadvertently leaving a significant void in the realm of headline generation for Indian languages. While datasets like Gigaword (Graff et al., 2003; Napoles et al., 2012) have emerged as prominent resources, comprising an impressive collection of over 4 million news article-headline pairs, it is crucial to acknowledge that they are limited to English and fail to capture the intricacies and linguistic nuances of Indian languages.

India, with its rich linguistic diversity, boasts a staggering array of over 22 officially recognized languages, each with its own distinct grammar, syntax, and vocabulary. Addressing the challenge of headline generation in Indian languages necessitates a deep understanding of the specific linguistic and cultural intricacies inherent in each language.

One of the most significant obstacles hindering headline generation in Indian languages is the scarcity of high-quality annotated data. This scarcity severely limits the effectiveness of model training and impedes the performance of supervised learning approaches, which heavily rely on labeled examples.

Fortunately, recent advancements in neural network architectures, such as transformer-based models, have significantly enhanced the performance of headline generation models. These models possess the ability to encode input text and generate headlines by optimizing various objectives, including semantic coherence, informativeness, and readability. While these models have successfully reduced the dependency on labeled data, they still leverage fine-tuning on specialized headline generation

* Authors contributed equally

datasets to further enhance their performance.

In the context of Bengali language, Salehin et al. (2019); Amin et al. (2021) conducted data collection[1] from various news websites using web scraping techniques. They proposed an RNN-based encoder-decoder model with an attention mechanism for headline generation. Another notable resource for multilingual abstractive summarization, XL-Sum, was introduced by Hasan et al. (2021). The Indian language section of the XL-Sum dataset consists of 251K article-headline pairs sourced from BBC[2]. To further advance research in Natural Language Generation (NLG) for Indian languages, Kumar et al. (2022) proposed the IndicNLG benchmark, encompassing five different NLG tasks, including a headline generation dataset (hereafter referred to as IndicHG dataset). This dataset comprises 1.31 million article-headline pairs across 11 Indian languages. However, our analysis (detailed in Section 4) reveals serious quality issues, such as data contamination, rendering it unsuitable for training robust models. Despite its claimed size, the dataset's problematic samples significantly reduce its effective size by nearly half. To summarize our main contributions:

1. We present a large, multilingual headline-generation dataset "Mukhyansh", comprising over 3.39 million news article-headline pairs across 8 Indian languages; namely Telugu, Tamil, Kannada, Malayalam, Hindi, Bengali, Marathi, and Gujarati. Our data collection methodology involves developing site-specific crawlers, leveraging a deep understanding of news website structures to ensure the acquisition of high-quality data.

2. We employ state-of-the-art baseline models and demonstrate the effectiveness of these models for a diverse range of test sets.

3. We provide further evidence to support our argument regarding the necessity of high-quality data by undertaking a comprehensive comparative analysis, specifically contrasting our research with the existing work, particularly IndicHG.

   The dataset and models are available at: `https://github.com/ltrc/Mukhyansh`

[1]However, the dataset is not made publicly available

[2]`https://www.bbc.com/`

The remaining sections of this paper are structured as follows: Section 2 provides a comprehensive introduction to Mukhyansh. Section 3 delves into the details of our baseline models. In Section 4, we meticulously evaluate the existing work, conduct a comparative analysis of each models' performance on diverse datasets, and present our findings. Section 5 concludes with our key contributions, limitations, and future scope.

2 Mukhyansh

The data collection process for all eight Indian languages involved web scraping from multiple news websites. However, this task posed challenges due to the diverse and dynamic nature of these websites.

Given that each website has its own unique structure, it was crucial to understand the intricacies of each site to extract data accurately, without any loss of information or introduction of noise. To achieve this, we developed site-specific web scrapers tailored to each website. These scrapers were designed to extract the text of news articles, headlines, and the name of the news subdomain. Care was taken to ensure that both the article and headline elements were non-empty and devoid of any unwanted information such as advertisements, URLs pointing to related articles, or embedded social media content.

To avoid any bias towards a particular news style, data was collected from a diverse range of news websites[3]. These websites covered various domains, including state, national, international, entertainment, sports, business, politics, crime, and COVID-19, among others[4]. To ensure the quality of the collected data, additional preprocessing steps were implemented next.

2.1 Preprocessing

In the series of essential preprocessing steps, firstly, we eliminate all special symbols, emojis, and punctuation marks from the dataset. Next, we remove any duplicate article-headline pairs from the dataset. Lead or prefix, wherein the title of an article is derived from the initial sections that typically contain the most crucial information, is a widespread approach adopted by news sites. Although utilizing the lead section can be beneficial

[3]See Appendix A for a detailed list of websites used for scraping

[4]Refer to Table 9 in Appendix for category-wise statistics of the dataset.

	Dravidian language family				Indo-Aryan language family			
	te	ta	kn	ml	hi	bn	mr	gu
# Pairs collected	1080665	378545	505641	435896	729950	309008	411566	338502
# Duplicates	8024	11546	64116	269	32539	7055	10184	35518
# Pairs after deduplication	1072641	366999	441525	435627	697411	301953	401382	302984
# Pairs with prefix	8756	1712	1983	21633	2656	1302	942	200
# Pairs with multiple-articles	582	0	0	0	0	0	0	0
# Pairs too short	146181	33579	101619	98921	94132	19378	65998	26826
# Pairs after filtering	917122	331708	337923	315072	600623	281273	334442	275958
# Pairs in train	825372	298543	304122	283555	540568	253139	301001	248367
# Pairs in dev	82571	26539	27044	25190	48042	22514	26751	22073
# Pairs in test	9179	6626	6757	6327	12013	5620	6690	5518

Table 1: Statistics of Mukhyansh Preprocessing.

for summary generation, it may inadvertently hinder the model's ability to learn and discriminate between different types of information. By relying solely on the lead, the model may overlook relevant details and nuances present in the subsequent sections of the article. Therefore, we eliminate pairs with prefixes from the dataset. Furthermore, to ensure that only substantial and informative pairs are retained, we apply a minimum-length filter to the dataset. This filter helps eliminate article-headline pairs where the article contains fewer than 20 tokens and/or the headline consists of fewer than 3 tokens. Table 1 provides an overview of the preprocessing statistics for Mukhyansh and the final Train, Dev, and Test splits.

For the final splits, we allocated 90% of the data for training purposes, while the remaining data was dedicated to development and testing. To ensure robust performance and prevent any bias towards specific news categories or domains, stratified sampling techniques were employed when creating our data splits. This approach guarantees that articles from all categories are evenly distributed across the training, development, and test sets. Additional statistical details of the Mukhyansh dataset can be found in Table 10.

2.2 Human Evaluation

In order to evaluate the quality of the Mukhyansh dataset more comprehensively, a human evaluation was conducted. Due to resource constraints and the expenses associated with annotation, this evaluation was limited to the Telugu language data. A total of 500 article-headline pairs were randomly selected and assigned to native-language annotators. They were provided with a set of guidelines, which were based on those utilized in previous studies such as XL-Sum (Hasan et al., 2021) and IndicNLG (Kumar et al., 2022). The evaluation specifically focused on the following properties:

- **Consistent** *True*, If the article and headline are consistent.
- **Inconsistent** *True*, If the headline contains information that is inconsistent with the article.
- **Unfounded** *True*, If the headline contains extra information that cannot be inferred from the article.

We assign each article-headline pair to 3 annotators and the final rating for each pair is selected based on majority voting. We found that 96.8% of the samples were rated *True* for *Consistency*, and the percentage of samples that are rated *Inconsistent*, and *Unfounded* were 0.6%, and 2.6% respectively, which supports our claim of a reliable and good-quality dataset.

The inter-annotator agreement was assessed using a variation of Fleiss' Kappa, proposed by (Randolph, 2005) and it resulted in an encouragingly high score of 0.76, indicating substantial agreement among annotators.

3 Baseline Models

In our research paper, we evaluate the performance of commonly used sequence-to-sequence models as baselines on our dataset. Our implementation includes two categories of models: one based on an RNN encoder-decoder network trained from scratch, and another utilizing fine-tuning with pre-trained transformer encoder-decoder models like mT5 (Xue et al., 2021) and IndicBART (Dabre et al., 2022).

For the RNN architecture, we adopt the recurrent neural network proposed by Sutskever et al. (2014),

L	FastText+GRU			FastText+LSTM			BPEmb+GRU			mT5-small			SSIB		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
te	32.71	15.00	32.02	33.41	14.93	32.70	30.06	14.52	29.31	39.34	21.95	**38.35**	38.42	20.85	37.33
ta	33.52	15.40	32.20	32.64	13.60	31.26	33.28	16.15	32.04	43.22	24.38	**41.18**	43.47	24.50	41.16
kn	26.19	10.53	25.25	23.75	7.94	22.84	24.46	10.68	23.60	34.73	17.88	**33.34**	34.36	17.06	32.59
ml	28.86	13.17	28.17	24.00	8.80	23.44	26.13	13.22	25.36	35.50	20.79	**34.63**	33.21	18.57	32.04
hi	32.97	14.20	29.50	32.34	11.79	28.45	32.24	13.93	28.94	38.26	18.81	33.65	41.05	20.77	**36.18**
bn	18.55	6.15	17.47	15.73	4.00	14.90	10.20	2.31	9.84	22.90	8.87	21.56	23.67	8.84	**22.04**
mr	17.26	5.08	16.83	14.32	3.11	14.04	17.91	6.48	17.54	27.25	12.68	26.41	28.21	12.95	**27.08**
gu	15.61	3.87	14.84	9.98	1.68	9.48	15.68	4.59	14.94	21.80	8.53	20.43	24.77	9.86	**23.05**
Average	25.71	10.43	24.54	23.27	8.23	22.14	23.75	10.24	22.70	32.88	16.74	31.19	33.40	16.68	31.43

Table 2: ROUGE-1,2,L scores of various baseline models of Mukhyansh for each language (L).

with a simple context attention mechanism inspired by Lopyrev (2015), which is a modification of the dot product attention mechanism introduced by Luong et al. (2015). We explore two variations of this model: one using GRU (Cho et al., 2014) in both the encoder and decoder, and the other utilizing LSTM (Hochreiter and Schmidhuber, 1997).

To tackle the challenge of out-of-vocabulary (OOV) words, particularly prevalent in morphologically rich Indian languages, we employ Byte Pair Encoding (BPE) (Gage, 1994). Specifically, we use the GRU architecture[5] mentioned earlier and initialize the model with 300d subword embeddings from BPEmb (Heinzerling and Strube, 2018).

In addition to the above approaches, we also leverage the benefits of transfer learning in headline generation by utilizing pre-trained sequence-to-sequence models such as mT5 and IndicBART. To implement these models, we utilize the scripts[6] provided by Huggingface (Wolf et al., 2020).

mT5: mT5 is a multilingual variant of T5 (Raffel et al., 2020) covering 101 languages. For our baseline, we fine-tune the pre-trained mT5-small model on our dataset.

IndicBART: IndicBART is a multilingual, sequence-to-sequence pre-trained model focusing on 11 Indian languages and English. It is similar to mBART (Liu et al., 2020) in terms of architecture and training methodology. Specifically, we use a variant of IndicBART called separate script IndicBART[7] (hereafter referred to as SSIB) and fine-tune it on our dataset for the task of headline generation.

[5]GRUs use fewer parameters, making them more computationally efficient for our experiments, with limited compute resources.

[6]https://github.com/huggingface/transformers/tree/main/examples/pytorch/summarization

[7]https://huggingface.co/ai4bharat/IndicBARTSS

Parameters	**Seq-Seq + FastText**	**Seq-Seq + BPEmb**	**mT5-small**	**SSIB**
Max Source Length	200	300	1024	1024
Max Target Length	20	30	30	30
Vocabulary Size	40000	40000	250112	64000
Beam Width	5	5	4	4
Batch Size	16	16	16	16
Optimizer	Adam	Adam	Adam	Adam
Learning rate	$1e^{-4}$	$1e^{-4}$	$5e^{-5}$	$5e^{-5}$
(GPU,CPU)	(1,10)	(1,10)	(4,40)	(4,40)

Table 3: Experimental setup of various baseline models.

3.1 Experimental Setup

The LSTM and GRU models used in this research paper consist of 4 stacked layers, with each LSTM/GRU cell containing 600 hidden activation units. To initialize the word embeddings, we employ the 300d pre-trained FastText embeddings (Grave et al., 2018) for each language.

During the inference phase, we utilize the beam search strategy with length normalization penalty (Wu et al., 2016). After conducting experiments with various penalty values, we found that a penalty of 0.1 for Telugu, Tamil, Kannada, and Malayalam, and no length normalization for other languages, yielded superior results. To prevent overfitting, we employ early stopping.

Conversely, due to limited computational resources, for the pre-trained models we fine-tuned them on our data for 10 epochs. The model checkpoint with the highest validation score is selected to generate predictions on the test set.

To assess the models' performance, we utilize the multilingual ROUGE metric (Hasan et al., 2021)[8]. Further details regarding the experimental setup and parameter configurations for all the models can be found in Table 3.

[8]https://github.com/csebuetnlp/xl-sum/tree/master/multilingual_rouge_scoring

3.2 Results

Table 2 presents the ROUGE-1, 2, L (R-1, R-2, R-L) scores achieved by different baseline models on Mukhyansh. The best R-L score for each language is highlighted in bold. Notably, the SSIB and mT5-small models outperformed all the sequence-to-sequence models trained from scratch. The superior performance of SSIB and mT5-small can be attributed to their pre-training on a large corpus.

It is worth mentioning that the GRU variant of the sequence-to-sequence model, utilizing FastText embeddings, yielded satisfactory results with a smaller parameter count (64 Million) compared to SSIB (244 Million) and mT5-small (300 Million).

IndicHG Performance			
L	**Reported**	**Reproduced**	**Unbiased**
te	41.97	22.37	19.47
ta	46.52	32.96	33.79
kn	73.19	42.79	21.64
ml	60.51	35.64	26.79
hi	34.49	24.12	22.68
bn	37.95	22.54	20.28
mr	40.78	21.28	20.14
gu	31.80	22.68	22.61
Average	45.90	28.05	23.42
Performance drop		17.85	22.48

Table 4: Performance Comparison of various versions of IndicHG: Reported, IndicHG* and IndicHG_Unbiased.

4 Existing Dataset Evaluation

Due to the unavailability of publicly accessible data from existing monolingual works, our evaluation is limited to the recent multilingual datasets, namely XL-Sum and IndicHG. While XL-Sum focuses on extreme summarization, it is important to note that the summaries provided may consist of more than one sentence. Additionally, concerns have been raised by Urlana et al. (2022) regarding the quality of summaries in the Indian language section of XL-Sum. Consequently, our evaluation is primarily centered on the IndicHG dataset[9].

To validate the reported results in IndicNLG regarding headline generation, we conduct a series of experiments on the IndicHG dataset, accompanied by comprehensive quantitative and qualitative analyses. As discussed in the subsequent sub-sections, our investigation has uncovered significant quality issues with the HG dataset of IndicNLG. Despite the valuable contributions of IndicNLG to the field of language generation for various Indic languages, it is imperative to address these issues before deeming the IndicHG dataset suitable for training robust models.

4.1 Reproducing IndicHG Results

We initiate our experiments with an attempt to replicate the findings of IndicHG for the eight Indian languages mentioned. Following their paper's methodology and hyper-parameter settings, we meticulously fine-tune the SSIB model, (hereafter, referred to as *IndicHG**). In order to obtain a more reliable assessment of the model's performance and evaluate the consistency of the results, we conducted the same experiment five times with different initial seeds. Subsequently, we calculate the mean and standard deviation of the ROUGE-L scores[10] obtained on the test set. Table 4 presents these mean ROUGE-L scores alongside their reported[11] counterparts.

As depicted in the final row of Table 4, there is an average reduction of 17.85 in the ROUGE-L scores across the eight languages. This substantial decrease raises concerns regarding the reproducibility of the original findings and emphasizes the necessity for further investigation.

4.2 Quantitative Analysis

We initiate the analysis by implementing preprocessing steps for the IndicHG dataset, including checks for prefixes, duplicates, and minimum length. In addition to the eight languages we are focusing on, we extended the preprocessing to include the remaining three languages of IndicHG: Oriya, Punjabi, and Assamese.

Surprisingly, despite claims to the contrary, our analysis reveals that the IndicHG dataset contains a significant number of duplicate article-headline pairs in the training, development, and test splits for most languages. Out of the total 1.31 million pairs, approximately 0.67 million (51.23%) are duplicates. Moreover, it is ideal for a dataset to have no overlap or common samples among the train-

[9]IndicNLG data for Headline-generation was taken from https://huggingface.co/datasets/ai4bharat/IndicHeadlineGeneration/tree/main/data

[10]Due to space constraints, additional details and the corresponding ROUGE-1, ROUGE-2 scores are reported in Appendix B, Table 12

[11]The reported scores are taken from the monolingual works of IndicHG (Kumar et al., 2022) paper, as the checkpoint is not made public.

L	Train set		Development set			Test set			Total		
	# Pairs	Duplicates (%)	# Pairs	Duplicates (%)	Train Overlap (%)	# Pairs	Duplicates (%)	Train-Dev Overlap (%)	# Pairs	(Duplicates + Overlap) (%)	Remaining
te	21352	8.77	2690	1.52	15.61	2675	1.42	18.61	26717	10.38	23945
ta	60650	51.18	7616	50.22	3.31	7688	50.20	3.62	75954	51.29	36996
kn	132380	87.26	19416	84.29	59.18	3261	6.23	71.17	155057	87.51	19364
ml	10358	22.83	5388	76.26	33.33	5220	76.05	44.22	20966	53.78	9690
hi	208091	3.19	44718	0.76	6.42	44475	0.72	7.83	297284	4.59	283646
bn	113424	69.86	14739	68.02	19.41	14568	67.94	24.30	142731	70.65	41896
mr	114000	69.10	14250	66.95	15.45	14340	67.03	16.15	142590	69.73	43157
gu	199972	75.11	31270	80.04	0.96	31215	80.02	1.28	262457	76.33	62123
pa	48441	0.13	6108	0	0.18	6086	0	0.35	60635	0.16	60540
as	29631	30.05	14592	75.96	58.77	14808	75.97	65.91	59031	60.66	23222
or	58225	48.77	7484	48.97	0.16	7137	48.58	0.42	72846	48.79	37305
								Total:	1316268	51.23	641884

Table 5: IndicHG Analysis: Showing overall duplication and overlap(or data-contamination) percentages.

ing, development, and test splits. However, the statistics presented in Table 5 demonstrate a high level of overlap among these splits for most of the languages, corroborating data contamination. For instance, an article-headline pair[12] from the Kannada language appears 115 times in the training data, 18 times in the development data, and 2 times in the test data.

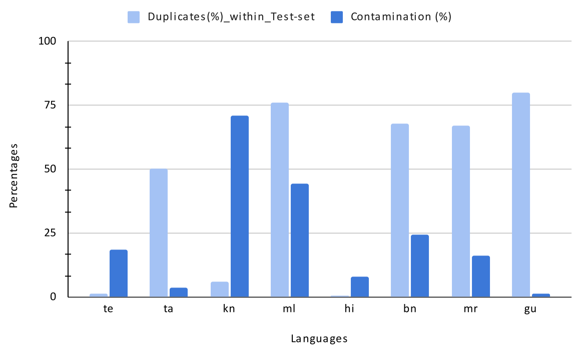

Figure 1: Language-wise data bias in IndicHG test-set.

Data contamination introduces bias in evaluation, as the metrics calculated on the development and test datasets do not accurately represent the model's performance on unseen data. Additionally, we assert that the heavy presence of duplicated data in the dataset may lead models trained on this data to achieve artificially high performance by memorizing the duplicated pairs, thereby hindering their ability to generalize to new, unseen data.

To support our arguments, we take several steps. Firstly, we eliminate all duplicate pairs from each of the training, development, and test splits of the IndicHG dataset. To deal with data contamination, the following 2 variations were attempted:

1. To ensure the integrity of the test set, a straightforward approach was adopted, which involved excluding any pairs that were already present in the corresponding train/dev sets. Additionally, any pairs in the dev set that were already present in the train set were also removed. This approach effectively eliminated data contamination and allowed the training set to remain as large as possible. These splits were then utilized to reproduce the IndicHG results as *IndicHG_Unbiased*. Notably, this dataset exhibited a significant decrease in average R-L score, with a decrease of 22.48 compared to the score reported in the original IndicNLG paper (Kumar et al., 2022), resulting in an average R-L score of 23.42; as outlined in Table 4.

   To evaluate the specific impact of data contamination, we divided the IndicHG test set into two subsets. The first subset consisted of pairs from the IndicHG test set that were also present in the corresponding train or dev sets. The second subset comprised the remaining (unique) pairs from the original test set. Figure 2 shows the R-L score comparison[13] for these two test subsets, referred to as *Overlaps* and *Without_Overlap* respectively, against those obtained from the total (original) test set. The results unequivocally support the claim that data contamination indeed leads to artificial high performance.

2. As an alternative approach, pairs present in the training set that also appeared in the corresponding dev and test sets were eliminated. Similarly, pairs in the dev set that were already present in the test set were excluded. Additionally, pairs were filtered out if the headline was found in the article's prefix, or if the pairs were too short. This method aimed to ensure that the new

[12] https://tinyurl.com/2p85mayt

[13] For details refer to Table 13

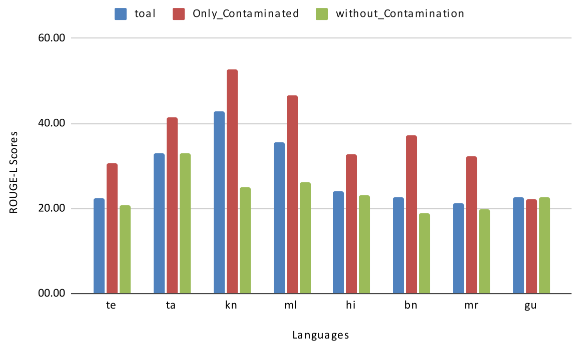

Figure 2: ROUGE-L scores for subsets of IndicHG Test set.

test set closely resembled the original set while eliminating problematic cases. The stepwise statistics of this filteration process and final split counts are provided in Table 6. Further statistics of the resulting filtered dataset, referred to as *IndicHG_filtered*, can be found in Table 11.

While it may seem intuitive that a larger training set would lead to better model training, our findings suggest that both of the aforementioned approaches yield similar scores. Consequently, we have decided to utilize the *IndicHG_filtered* version for all future cross-comparisons. This is primarily because its test set bears closer resemblance to the original test set. Section 4.4 describes further experimentation conducted using this dataset.

4.3 Qualitative Analysis:

To conduct a qualitative analysis, we begin by manually evaluating a random selection of article-headline pairs from the IndicHG Telugu dataset[14]. This dataset comprises articles collected from approximately 22 different Telugu news websites. To ensure a comprehensive evaluation, we assess at least five random pairs from each website. Our evaluation brings to light certain issues that indicate a lack of site-specific scraping implementation in IndicHG. The identified issues are as follows:

1. Unwanted information (noise) is present at the beginning of the article.
2. Headline is out of the context of the article.
3. The article part of a pair, itself contains multiple other article-headline pairs.

These quality issues in the article-headline pairs can significantly impact the performance of models. When the headline is contextually unrelated to the article, the generated headlines by the model are inaccurate, resulting in subpar performance. Likewise, the presence of multiple articles within a single article introduces irrelevant information, causing the model to focus on only a fraction of the total content.

For each of the aforementioned issues, we meticulously document the corresponding source website. Subsequently, we employ simple scripts, regular expressions, and other techniques to further examine all the article-headline pairs from these source websites. Among all the issues observed, the most prevalent is the occurrence of multiple articles within a single article (issue-3). By employing basic regular expressions, we were able to detect a total of 5773 such pairs, although not capturing all instances, primarily sourced from the Andhra Bhoomi website[15], which constitutes 30% of the Telugu IndicHG dataset. Considering the significant quantity of such pairs, we further update our *IndicHG_filtered* dataset by eliminating these pairs.

For further examples and additional details regarding all other identified problematic cases, please refer to Appendix B.2.

4.4 Experiments and Analysis

In order to assess the effectiveness of different models, we fine-tune the SSIB model[16] using a range of specifically crafted training and test sets:

1. First, we fine-tune a model on the IndicHG_filtered dataset and evaluate its performance on the corresponding filtered test set, while ensuring that the fine-tuning hyperparameters remain consistent with those described in the IndicNLG paper. The results, as presented in Table 7, demonstrate the true performance of IndicHG when only good quality unique pairs are considered. It is evident that the ROUGE-L scores decrease significantly compared to the scores produced by the biased data (i.e. unfiltered IndicHG). Next, other models were also tested on IndicHG_filtered test set. See, Table 7. Notably, while testing IndicHG* model on IndicHG_filtered test set, we are bound to get

[14] Manual evaluation was restricted to Telugu, due to limited language experts/resources.

[15] http://www.andhrabhoomi.net/

[16] Unless otherwise stated, all experiments conducted in this study were based on the SSIB model.

	Dravidian language family				Indo-Aryan language family			
	te	ta	kn	ml	hi	bn	mr	gu
Total # Pairs	26717	75954	155057	20966	297284	142731	142590	262457
# Duplicates	2772	38958	135693	11276	13638	100835	99433	200334
# Pairs after deduplication	23945	36996	19364	9690	283646	41896	43157	62123
# Pairs with prefix	669	796	5	22	19	336	6	92
# Pairs with multiple-articles	5773	0	0	0	0	0	0	0
# Pairs too short	30	5	7	8	1470	5	8	7
# Pairs after filtering	17473	36195	19352	9660	282157	41555	43143	62024
# Pairs in train	13539	28750	13602	7235	194627	32435	33772	49566
# Pairs in dev	1903	3702	2693	1177	43604	4480	4644	6228
# Pairs in test	2031	3743	3057	1248	43926	4640	4727	6230

Table 6: IndicHG_filtered dataset creation statistics.

Test sets	**Models Fine-tuned on**	**Language**								**Average**
		te	ta	kn	ml	hi	bn	mr	gu	
IndicHG	IndicHG_filtered	17.80	33.46	22.98	25.09	24.52	19.18	21.35	22.87	23.41
	Mukhyansh	**27.05**	**35.05**	**28.20**	**29.23**	**26.84**	17.65	**26.31**	19.86	**26.27**
	Mukhyansh_small	21.46	30.68	23.09	24.04	23.39	15.66	22.22	18.59	22.39
IndicHG_filtered	IndicHG* (with overlap)	20.58	32.50	41.89	33.29	23.92	21.78	21.93	22.51	27.30
	IndicHG_filtered	16.66	32.85	22.91	25.11	24.61	18.97	21.38	22.86	23.17
	Mukhyansh	**24.00**	**34.96**	**27.98**	**29.28**	**26.95**	17.66	**26.33**	19.85	**25.88**
	Mukhyansh_small	19.67	30.54	22.98	24.00	23.50	15.66	22.24	18.59	22.15
Mukhyansh	IndicHG*	19.83	29.31	20.61	19.51	23.95	14.80	15.29	16.19	19.94
	IndicHG_filtered	17.53	29.43	18.66	20.44	26.07	16.13	15.73	16.56	20.07
	Mukhyansh	**37.33**	**41.16**	**32.59**	**32.04**	**36.18**	**22.04**	**27.08**	**23.05**	**31.43**
	Mukhyansh_small	28.66	36.01	26.11	26.73	32.39	18.96	22.59	20.49	26.49

Table 7: Performance comparison (by ROUGE-L) of various models.

biased (high) scores. This is because in case of IndicHG_filtered, the training set itself was prepared without overlapping pairs (leaving them intact in the corresponding test set). Keeping this bias aside, our Mukhyansh model outperforms all the others.

2. To further investigate the impact of quality vs. quantity, we prepare a smaller version of the Mukhyansh dataset. In order to create the new train, dev, and test sets, separate random sampling is performed over the original train, dev, and test sets of Mukhyansh. A model, called *Mukhyansh_small*, is then fine-tuned only on this smaller train set, and tested against other models, see Table 7.

This cross-comparison was then concluded by testing Mukhyansh's SSIB baseline against all other test sets. And as evident by the R-L scores (highlighted as bold) in Table 7 Mukhyansh outperforms almost all the other models.

We acknowledge the multilingual models as the limitation and future scope of this work. Due to limited compute-resources we could not fine-tune any multilingual models. However, we believe that multilingual fine-tuning on Mukhyansh dataset would give new state-of-the-art models.

5 Conclusion

Headline generation in low-resource languages, such as Indian languages, faces significant challenges due to the scarcity of large, high-quality annotated data. Our work address this gap by introducing Mukhyansh, a comprehensive multilingual dataset comprising over 3.39 million article-headline pairs across eight prominent Indian languages. The importance of our work is substantiated by empirical analysis of existing works, uncovering critical data quality issues. Through extensive experimentation, we demonstrate the superiority of Mukhyansh and our SSIB baseline model, surpassing all existing works in Indian language headline generation. This achievement highlights the effectiveness of Mukhyansh in advancing research efforts in low-resource language processing and establishes it as a valuable resource for future exploration and innovation in this field.

6 Ethics Statement

The distribution of the dataset collected from the web raises ethical considerations. We acknowledge that the copyright of the news articles collected from various websites remains with the original creators. Considering that each website may have its own policies regarding data distribution or public availability, we offer researchers the URLs and web scraping scripts necessary to reproduce the data, ensuring transparency and encouraging proper attribution through the release of the list of URLs under the Creative Commons license[17]. To ensure the reproducibility of the model results, we plan to release various baseline model checkpoints used for headline generation at a later date.

References

Ruhul Amin, Nabila Sabrin Sworna, Md Nazmul Khan Liton, and Nahid Hossain. 2021. Abstractive headline generation from bangla news articles using seq2seq rnns with global attention. In *2021 International Conference on Science Contemporary Technologies (ICSCT)*, pages 1–5.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. pages 103–111.

Raj Dabre, Himani Shrotriya, Anoop Kunchukuttan, Ratish Puduppully, Mitesh Khapra, and Pratyush Kumar. 2022. IndicBART: A pre-trained model for indic natural language generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1849–1863, Dublin, Ireland. Association for Computational Linguistics.

Philip Gage. 1994. A new algorithm for data compression. *C Users Journal*, 12(2):23–38.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. XL-sum: Large-scale multilingual abstractive summarization for 44 languages.

[17] https://creativecommons.org/licenses/by/4.0/

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Aman Kumar, Himani Shrotriya, Prachi Sahu, Amogh Mishra, Raj Dabre, Ratish Puduppully, Anoop Kunchukuttan, Mitesh M. Khapra, and Pratyush Kumar. 2022. IndicNLG benchmark: Multilingual datasets for diverse NLG tasks in Indic languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5363–5394, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Konstantin Lopyrev. 2015. Generating news headlines with recurrent neural networks. *arXiv preprint arXiv:1512.01712*.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. 2012. Annotated Gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 95–100, Montréal, Canada. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Justus J Randolph. 2005. Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa. *Online submission*.

Mushfiqus Salehin, Ashik Ahamed Aman Rafat, Fazle Rabby Khan, and Sheikh Abujar. 2019. Generating bengali news headlines: An attentive approach with sequence-to-sequence networks. In *2019 8th International Conference System Modeling and Advancement in Research Trends (SMART)*, pages 256–261.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. 27.

Ashok Urlana, Nirmal Surange, Pavan Baswani, Priyanka Ravva, and Manish Shrivastava. 2022. TeSum: Human-generated abstractive summarization corpus for Telugu. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5712–5722, Marseille, France. European Language Resources Association.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. 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*, pages 483–498, Online. Association for Computational Linguistics.

A Mukhyansh Dataset Additional Details

- Websites used for scraping: To make the dataset more diverse, the data is scraped from a total of 47 websites across all 8 languages, and the list of websites is provided in Table 8.
- To eliminate any bias towards particular news categories we make sure that the scraped dataset covers diverse set of news categories. The category/domain-wise statistics of the Mukhyansh dataset are presented in Table 9.
- To evaluate the task's abstractive nature and difficulty, we compute the percentage of novel n-grams and employ extractive baselines like LEAD-1 and EXT-ORACLE ROUGE-L (R-L) scores. The "percentage of novel n-grams" indicates the proportion of n-grams present in the headline but not found in the article, quantifying the level of uniqueness in the generated summary. Specifically, LEAD-1 R-L calculates the similarity between the first sentence of the article and the reference headline, while EXT-ORACLE R-L computes scores by selecting the sentence from the article that achieves the highest R-L scores with the reference headline. The resulting scores along with other statistics are detailed in Table 10

S.No	L	Website	S.No	L	Website	S.No	L	Website
1	te	https://www.ap7am.com/telugu-news	17	kn	https://kannadanewsnow.com/kannada/	33	ml	https://eveningkerala.com/
2	te	https://www.prabhanews.com/	18	kn	https://hosadigantha.com/	34	hi	https://www.jagran.com/
3	te	https://www.suryaa.com/index.html	19	kn	https://kannada.asianetnews.com/	35	hi	https://www.khaskhabar.com/
4	te	https://www.manatelangana.news/	20	kn	https://newskannada.com/	36	hi	https://www.indiatv.in/
5	te	http://www.andhrabhoomi.net/	21	kn	https://www.kannadaprabha.com/	37	bn	https://www.anandabazar.com/
6	te	https://prajasakti.com/	22	kn	https://www.sahilonline.net/ka	38	bn	https://www.sangbadpratidin.in/
7	te	https://www.vaartha.com/	23	kn	https://www.udayavani.com/	39	bn	https://bengali.abplive.com/live-tv
8	te	https://10tv.in/	24	kn	http://vishwavani.news/	40	bn	https://uttarbangasambad.com/
9	te	https://www.hmtvlive.com/	25	kn	https://ainlivenews.com/	41	bn	https://bangla.asianetnews.com/
10	ta	https://www.hindutamil.in/	26	kn	https://vaarte.com/	42	mr	https://www.lokmat.com/
11	ta	https://www.polimernews.com/	27	kn	https://btvkannada.com/	43	mr	https://prahaar.in/
12	ta	https://tamil.asianetnews.com/	28	ml	https://www.eastcoastdaily.com/	44	mr	https://marathi.abplive.com/
13	ta	https://www.updatenews360.com/	29	ml	https://suprabhaatham.com/	45	gu	https://sandesh.com/
14	kn	https://kannadadunia.com/	30	ml	https://www.bignewslive.com/	46	gu	https://www.gujaratsamachar.com/
15	kn	https://eesanje.com/	31	ml	https://www.malayalamexpress.in/	47	gu	https://gujarati.news18.com/
16	kn	https://www.vijayavani.net/	32	ml	https://dailyindianherald.com/			

Table 8: List of websites used for creating Mukhyansh.

B IndicHG Analysis

B.1 Reproduced Results

In this section, we present the results of the experiment conducted to reproduce the results in the IndicNLG paper by fine-tuning the SSIB model on the IndicHG dataset. We report the mean and standard deviation of R-1, R-2, and R-L scores across multiple runs (i.e. using 5 different seeds to initialize the model). Table 12 provides the detailed statistics.

B.2 Problem Cases

In this section we present various issues that are present in IndicHG dataset. Table 13 gives language-wise ROUGE-L scores for overlapping and non-overlapping pairs of the IndicHG test set against the scores of the total test set. Figure 3 dipicts the percentages of duplication remained in train, dev and test splits of IndicHG after removing all overlapping pairs. An example of the prefix

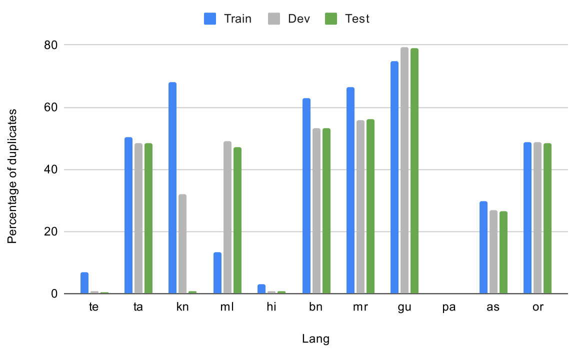

Figure 3: Duplication percentage within IndicHG train,dev,test splits.

case is presented in Table 14.

Sample article-headline pairs pertaining to the issues mentioned in section 4.3 are presented in Table 15 and Table 16. For better readability, instead of the Telugu script, we transliterate the text into Latin characters using ISO 15919 standard code.

The analysis of article-headline pairs of BBC Telugu[18], BBC Tamil[19] websites that are present in IndicHG dataset is detailed in Table 17.

C Examples of Model generated Headlines

This section presents the examples of headlines generated by various baseline models fine-tuned on Mukhyansh. Table 18 and Table 19 presents Hindi, Telugu examples respectively.

[18] https://www.bbc.com/telugu
[19] https://www.bbc.com/tamil

News Category	Category-wise counts of article-headline pairs for each language							
	te	ta	kn	ml	hi	bn	mr	gu
state	698059	133599	163857	144491	-	143804	184045	123183
national	91787	80711	61170	92833	314528	42913	72182	53248
entertainment	59244	31265	22697	14939	80202	31470	2819	19710
international	24262	29463	26092	34008	29668	20552	15347	37682
sports	19933	26186	18775	10204	78190	30676	29947	19337
business	13495	12874	8747	3446	60524	775	10379	21884
crime	8917	6656	7541	7064	8052	-	16489	-
covid	1425	6470	14147	4348	-	4205	-	-
politics	-	4484	5816	843	29459	346	3234	-
other	-	-	9081	2896	-	6532	-	914

Table 9: Category wise statistics of Mukhyansh

L	Total Pairs	Avg sents in article	Avg tokens in article	Avg tokens in headline	Total Tokens		Unique Tokens		% novel n-gram				Lead-1	EXT-ORACLE
					articles	headlines	articles	headlines	n=1	n=2	n=3	n=4	R-L	R-L
te	917122	7.97	103.64	7.42	95.05M	6.80M	2.3M	376K	36.63	62.87	82.10	91.41	23.54	33.21
ta	331708	15.47	218.99	11.50	72.64M	3.82M	1.8M	225K	33.02	55.12	73.75	85.05	32.70	39.33
kn	337923	10.94	154.77	9.03	52.3M	3.05M	1.9M	222K	41.30	65.88	82.73	91.45	19.66	30.08
ml	315072	10.26	115.45	9.54	36.37M	3.01M	2.5M	351K	36.14	55.59	71.20	81.73	34.60	41.94
hi	600623	14.54	303.05	13.45	182.02M	8.08M	1.3M	137K	20.31	47.20	67.96	81.27	25.99	35.02
bn	281273	19.41	244.78	10.10	68.85M	2.84M	0.9M	135K	37.60	67.60	84.31	92.27	15.51	30.50
mr	334442	17.71	271.02	8.41	90.64M	2.81M	1.9M	241K	37.11	64.73	82.66	91.66	13.88	28.34
gu	275958	16.45	284.39	12.46	78.48M	3.44M	1.7M	197K	38.24	65.81	82.08	90.54	12.21	28.72

Table 10: Mukhyansh dataset statistics in detail.

L	Total Pairs	Avg sents in article	Avg tokens in article	Avg tokens in headline	Total Tokens		Unique Tokens		% novel n-gram				Lead-1	EXT-ORACLE
					articles	titles	articles	titles	n=1	n=2	n=3	n=4	R-L	R-L
te	17473	13.99	185.09	7.97	3.2M	139K	238K	35.6K	36.26	65.66	85.87	94.08	15.23	29.30
ta	36195	13.43	181.77	11.76	6.58M	425K	311K	51.9K	32.94	54.89	70.17	78.96	33.46	40.10
kn	19352	11.49	189.16	9.22	3.66M	178K	237K	34.1K	33.02	57.47	75.89	86.59	18.19	29.73
ml	9660	13.55	168.38	10.08	1.63M	97K	232K	29K	39.15	61.48	77.78	87.04	26.40	35.79
hi	282157	18.25	397.08	12.55	112.04M	3.5M	543K	74.9K	20.79	49.86	71.06	83.56	21.99	32.52
bn	41555	14.65	239.55	11.27	9.95M	468K	245K	47.2K	38.02	64.35	80.91	89.33	13.95	27.39
mr	43143	13.61	205.31	8.57	8.86M	369K	258K	51K	31.45	57.76	77.44	86.59	13.08	32.55
gu	62024	12.31	226.64	11.20	14.06M	694K	425K	81.5K	35.85	60.69	76.75	85.94	15.52	29.39

Table 11: IndicHG_filtered dataset statistics in detail.

L	R-1		R-2		R-L	
	mean	std	mean	std	mean	std
te	23.75	1.31	11.98	0.88	22.37	1.28
ta	34.49	0.70	21.06	0.62	32.96	0.74
kn	43.85	1.41	35.89	1.58	42.79	1.43
ml	37.20	1.62	25.59	1.89	35.64	1.72
hi	28.73	0.63	13.42	0.37	24.12	0.62
bn	24.54	0.29	12.58	0.36	22.54	0.31
mr	22.99	0.46	11.26	0.28	21.28	0.38
gu	24.77	0.22	11.87	0.15	22.68	0.35
Average	30.04	0.83	17.96	0.77	28.05	0.85

Table 12: Mean & Standard Deviation of 5 iterations of IndicHG* results

Test sets	Models Fine-tuned on	Language								Average
		te	ta	kn	ml	hi	bn	mr	gu	
IndicHG	IndicHG*	22.37	32.96	42.79	35.64	24.12	22.54	21.28	22.68	28.05
Overlaps (IndicHG)		30.53	41.36	52.63	46.61	32.81	37.12	32.21	22.27	36.94
IndicHG-Overlaps		20.84	32.87	25.00	26.07	23.08	18.79	19.92	22.53	23.64

Table 13: Impact of Overlap on IndicHG Performance (by ROUGE-L).

URL: https://www.bbc.com/telugu/india-48363611
Headline: vaisīpī mèjāriṭīki prajāśāṃti pārṭī gaṃḍikòṭṭiṃdā? òke peruto nilabèṭṭina abhyarthulaku vaccina oṭlènni? - BBC News tèlugu
Article: vaisīpī mèjāriṭīki prajāśāṃti pārṭī gaṃḍikòṭṭiṃdā? òke peruto nilabèṭṭina abhyarthulaku vaccina oṭlènni? 24 me 2019 dīnini kriṃdi vāṭito ṣer ceyaṃḍi ivi bayaṭi liṃklu, kābaṭṭi kòtta viṃḍolo tèravabaḍatāyi ivi bayaṭi liṃklu, kābaṭṭi kòtta viṃḍolo tèravabaḍatāyi ṣer pyānèlnu mūsiveyaṃḍi āṃdhapradeś ènnikallo kee pāl netṛtvaṃloni prajāśāṃti pārṭī cālā coṭla tana abhyarthulanu bariloki diṃpiṃdi. kònni coṭla vaisīpī abhyarthula perlanu polina vyakthulanu bariloki diṃpiṃdane vārtalu vaccāyi. dīnipai vaisīpī pratinidhulu mārci 26na dillīki vacci ènnikala saṃghāniki phiryādu kūḍā ceśāru.dādāpu 35 niyojakavargāllo tama abhyarthulanu polina abhyarthulanu prajāśāṃti poṭīlo nilabèṭṭiṃdani, dīnipai caryalu tīsukovālani koriṃdi.prajāśāṃti ènnikala gurtu ayina hèlikāpṭar kūḍā tama phyān gurtunu poli uṃdani, dīnipainā caryalu tīsukovālani koriṃdi.ayite, kee pāl nilabèṭṭina abhyarthula valla vaisīpīki naṣṭaṃ jarigiṃdā..? e niyojakavargāllo vaisīpī abhyarthula mèjāriṭīpai prabhāvaṃ paḍiṃdi? phalitālu èlā unnāyi? anedi kiṃdi paṭṭikalo cūḍòccu.kramasaṃkhya

Table 14: Example of a headline that is directly present in article's prefix. The text highlighted in cyan color is the prefix information which is the same as the headline, and the one in pink is unwanted information (noise).

URL: https://www.bbc.com/telugu/india-42493669
Headline: daṃgal bāhubali.. rèṃḍū rèṃḍe - BBC News tèlugu
Article: insṭèṃṭ tripul talākku cèllu daśābdālugā èṃto maṃdi musliṃ mahilala vedanaku kāraṇamaina vidhānaṃ..'insṭèṃṭ tripul talāk'.ī islāmik ācārānni rājyāṃga viruddhamani tīrmānistū deśa atyunnata nyāyasthānaṃ āgasṭulo cāritraka tīrpuni vèluvariṃciṃdi.aiduguru sabhyulunna dharmāsanaṃlo mugguru jaḍjilu 'insṭèṃṭ ṭripul talāk' rājyāṃga viruddhamanī , adi mahilalapai vivakṣa cūpedigā uṃdanī perkònnāru.suprīṃ korṭu prakaṭiṃcina ī nirṇayaṃ paṭla deśa prajalu, mukhyaṃgā musliṃ mahilalu harṣaṃ vyaktaṃ ceśāru.pārlamèṃṭulo ṭripul talāk billuni saitaṃ praveśa pèṭṭaḍaṃto dāniki saṃbaṃdhiṃcina caṭṭa rūpakalpanalo maro muṃdaḍugu paḍiṃdi.kānī kònni musliṃ mahilā saṃghālu, āl iṃḍiyā musliṃ parsanal lā borḍto sahā kònni rājakīya pārṭīlu mātraṃ ā billuni vyatirekistunnāyi. bhārat naṃ.1 èkkuva maṃdi bhāratīyulu iṣṭapaḍe krīḍa krikèṭ.kohlī siks kòṭṭinā, bumrā vikèṭ tīsinā adi tama ghanatenannaṭṭu krīḍābhimānulu saṃbara paḍatāru. alāṃṭi abhimānulanu utsāha parice maro arudaina mailurāyini bhārata krikèṭ jaṭṭu ī eḍādi tòlisāri namodu cesiṃdi. sèpṭèṃbarulo prakaṭiṃcina aisīsī ryāṃkullo aṭu ṭèsṭulū, iṭu vanḍellonū bhārat naṃbar.1 sthānānni kaivasaṃ cesukuṃdi. òkesāri ilā rèṃḍu phārmāṭlalo tòli sthānaṃlo nilavaḍaṃ bhārata jaṭṭuki ide mòdaṭisāri.kāgā, rohit śarma ī eḍādi civaralo vanḍello mūḍo dviśatakaṃ sādhiṃci prapaṃca rikārḍu nèlakòlpāḍu.mithālī sārathyaṃlo bhārata mahilā krikèṭ jaṭṭu prapaṃca kap phainalku ceri rannarapgā niliciṃdi. phibravarilo jarigina '2017 blaiṃḍ varalḍ ṭī20' krikèṭ ṭornīni kūḍā bhārata aṃdhula krikèṭ jaṭṭe gèlucukuṃdi. maropakka byāḍmiṃṭanlo tèlugu kurrāḍu kidāṃbi śrīkāṃt kòtta caritra sṛṣṭiṃcāḍu.òka eḍādilo nālugu sūpar sirīs ṭaiṭillu gèlucukunna tòli bhāratīyuḍigā rikārḍu nèlakòlpāḍu. iṃḍònesiyā, āsṭreliyā, ḍènmārk, phrāns deśāllo jarigina sūpar sirīs ṭornīllo śrīkāṃt vijetagā nilicāḍu

Table 15: Example of a headline that is out of context to the article. The text highlighted in cyan is the headline of the article (highlighted in lime), and the text highlighted in yellow is the headline of the article (highlighted in gray). Here, the actual headline has no context in the article.

URL: http://www.andhrabhoomi.net/content/dudddd
Headline: vimānaṃ ṭāyilèṭlo 3 kilola baṃgāraṃ svādhīnaṃ
Article: muṃbayi: dubāyi nuṃci ikkaḍiki vaccina vimānaṃlo polīsulu sodālu ceyagā ṭāyilèṭlo 3 kilola baṃgāraṃ bayaṭa paḍiṃdi. dubāyi nuṃci vaccina prayāṇikullo èvaro ī baṃgārānni tècci ṭāyilèṭlo vadilesi uṃṭārani polīsulu cèbutunnāru.kasṭams tanikhīllo dòrikipote kesulu pèḍatāranna bhayaṃto ilā baṃgārānni vadilesi uṃṭārani polīsulu anumānistunnāru. bhārat èdugudalalo yūpī kīlakaṃ lakno: bhārat ayidu ṭriliyan ḍālarla ārthika vyavasthagā avatariṃcaḍaṃlo, 2030 nāṭiki prapaṃcaṃloni mūḍu atyaṃta pèdda ārthika vyavasthalalo òkaṭigā èdagaḍaṃlo uttarpradeś òka mukhyamayina pātra poṣistuṃdani rakṣaṇa śākha maṃtri rājnāth siṃg annāru. padi maṃdiki kebinèṭ padavulu bèṃgalūru, phibravari 6: karnāṭakalo kāṃgrès- jeḍīès saṃkīrṇa prabhutvānni kuppakūlci bījepī adhikāraṃloki rāvaḍāniki sahakariṃcina 10 maṃdi phirāyiṃpu dārulaku mukhyamaṃtri yèḍyūrappa maṃtri vargaṃlo kebinèṭ padavulu labhiṃcāyi. 'iṃṭarnèṭ' prāthamika hakkukādu nyūḍhillī, phibravari 6: iṃṭarnèṭ viniyogiṃcukune hakku prāthamika hakku kādani, adi èṃta mātraṃ deśa bhadratato samānamaina prādhānyatanu kaligi unnadi kādani keṃdra maṃtri raviśaṃkar prasād guruvāraṃ rājyasabhalo prakaṭana ceśāru.deśa bhadratā paristhitulanu kūḍā aṃte prādhānyatato pariśīliṃcālsina avasaraṃ uṃdannāru.

Table 16: Example of article-headline pair with multiple unrelated articles and headlines present in the same piece of text. The text highlighted in cyan color is the headline, followed by its article highlighted in yellow.

Error Cases	Telugu	Tamil
# Pairs	1587	3800
# Pairs with headline present in prefix	484	1558
# Pairs with unwanted information in the article	1390	3494
# Pairs with above two issues in common	461	1436
# Pairs with headline that is out of the context to the article	174	184

Table 17: Statistics of problematic pairs of IndicHG dataset.

URL	https://www.jagran.com//news/national-five-children-killed-in-wall-collapse-incidents-10655913.html
Article	**Transliteration:** kauśāṃbīl uttara pradeśa ke kauśāṃbī jile meṃ do alaga-alaga jagahoṃ para divāra girane se pāṃca baccoṃ kī mauta ho gaī । pulisa ne somavāra ko batāyā ki kauśāṃbī jile ke patharāvana gāṃva meṃ ravivāra śāma ko miṭṭī se bane ghara kā divāra acānaka gira gayā divāra ke girane se subhāṣa, usakī bahana lakṣmī aura kuṃdrā kī dabane se mauta ho gaī । pulisa ne batāyā ki dūsarī ghaṭanā ayānā eriyā ke kārakāpura gāṃva meṃ divāra girane se do bacce rajanīśa aura priyaṃkā kī bhī mauta ghaṭanāsthala para hī ho gaī । **Translation:** Kaushambi. In Uttar Pradesh's Kaushambi district, five children died due to wall collapse at two different places. Police said on Monday that the wall of a house made of mud suddenly collapsed in Pathravan village of Kaushambi district on Sunday evening. Subhash, his sister Lakshmi and Kundra died due to the collapse of the wall. Police said that in the second incident, two children Rajneesh and Priyanka also died on the spot due to wall collapse in Karkapur village of Ayana area.
Actual Headline	**Transliteration:** yūpī meṃ divāra girane se pāṃca baccoṃ kī mauta **Translation:** Five children died due to wall collapse in UP
GRU + FastText	**Transliteration:** yūpī meṃ do alaga jagahoṃ para dīvāra girane se 5 baccoṃ kī mauta **Translation:** 5 children died due to wall collapse at two different places in UP
LSTM + FastText	**Transliteration:** yūpī meṃ do alaga hādasoṃ se pāṃca kī mauta **Translation:** Five killed in two separate accidents in UP
GRU + BPEmb	**Transliteration:** saraka hādase meṃ 0 baccoṃ kī mauta **Translation:** 0 children died in road accident
mT5-small	**Transliteration:** uttara pradeśa meṃ do jagahoṃ para divāra girane se 5 baccoṃ kī mauta **Translation:** 5 children died due to wall collapse at two places in Uttar Pradesh
SSIB	**Transliteration:** yūpī ke kauśāṃbī meṃ do alaga alaga jagahoṃ para divāra girane se 5 baccoṃ kī mauta **Translation:** 5 children died due to wall collapse at two different places in UP's Kaushambi

Table 18: Hindi example of headlines generated by various baseline models fine-tuned on Mukhyansh

URL	https://telangana.suryaa.com/telangana-updates-20874-.html
Article	**Transliteration:** bījepī neta baddaṃ bālrèḍḍi maraṇaṃ tīrani loṭani asèṃbli spīkar pocāraṃ śrīnivāsarèḍḍi annāru. baddaṃ bālrèḍḍi pārthivadehānni saṃdarśiṃci nivālularpiṃcāru. anaṃtaraṃ māṭlāḍutū prajala maniṣigā bālrèḍḍi gurtiṃpu tèccukunnārannāru. haidarābād prajalato bālrèḍḍiki avinābhāva saṃbaṃdhaṃ uṃdannāru. bālrèḍḍi kuṭuṃba sabhyulaku tana pragāḍha sānubhūti annāru. **Translation:** Assembly Speaker Pocharam Srinivas Reddy termed the death of BJP leader Baddam Bal Reddy as an irreparable loss. He visited the mortal remains of Baddam Bal Reddy and paid homage to him. Speaking after the meeting, he said that Bal Reddy has gained recognition as a people's man. Bal Reddy has a close relationship with the people of Hyderabad. My deepest condolences to the family members of Bal Reddy.
Actual Headline	**Transliteration:** bālrèḍḍi maraṇaṃ tīrani loṭu **Translation:** Bal Reddy's death is an irreparable loss.
GRU + FastText	**Transliteration:** mṛti tīrani loṭu pocāraṃ **Translation:** Death is an irreparable loss: Pocharam
LSTM + FastText	**Transliteration:** bījepī neta mṛti tīrani loṭu spīkar pocāraṃṃ **Translation:** BJP leader's death is an irreparable loss: Speaker Pocharam
GRU + BPEmb	**Transliteration:** bījepī neta baddaṃ bālrèḍḍi mṛti tīraniloṭu spīkar pocāraṃ śrīnivāsarèḍḍiki saṃtāularpiṃcina asèṃbli spīkar **Translation:** BJP leader Baddam Bal Reddy's death is a sad loss: Assembly Speaker pays condolences to Speaker Pocharam Srinivasa Reddy
mT5-small	**Transliteration:** baddaṃ bāl rèḍḍi maraṇaṃ tīrani loṭu pocāraṃ **Translation:** Baddam Bal Reddy's death is an irreparable loss: Pocharam
SSIB	**Transliteration:** baddaṃ bāl rèḍḍi maraṇaṃ tīrani loṭu spīkar pocāraṃ **Translation:** Baddam Bal Reddy's death is an irreparable loss: Speaker Pocharam

Table 19: Telugu example of headlines generated by various baseline models fine-tuned on Mukhyansh