Unsupervised Approach to Evaluate Sentence-Level Fluency: Do We Really Need Reference?

Gopichand Kanumolu*, **Lokesh Madasu***, **Pavan Baswani***, **Ananya Mukherjee***, **Manish Shrivastava**
Language Technologies Research Center, KCIS, IIIT Hyderabad, India.
{gopichand.kanumolu, lokesh.madasu}@research.iiit.ac.in
{pavan.baswani, ananya.mukherjee}@research.iiit.ac.in
m.shrivastava@iiit.ac.in

Abstract

Fluency is a crucial goal of all Natural Language Generation (NLG) systems. Widely used automatic evaluation metrics fall short in capturing the fluency of machine-generated text. Assessing the fluency of NLG systems poses a challenge since these models are not limited to simply reusing words from the input but may also generate abstractions. Existing reference-based fluency evaluations, such as word overlap measures, often exhibit weak correlations with human judgments. This paper adapts an existing unsupervised technique for measuring text fluency without the need for any reference. Our approach leverages various word embeddings and trains language models using Recurrent Neural Network (RNN) architectures. We also experiment with other available multilingual Language Models (LMs). To assess the performance of the models, we conduct a comparative analysis across *10 Indic languages*, correlating the obtained fluency scores with human judgments. Our code and human-annotated benchmark test-set for fluency is available at https://github.com/AnanyaCoder/TextFluencyForIndicLanaguges.

1 Introduction

Fluency measures the quality of the generated text output from the model without considering the reference text (Moorkens et al., 2018). It accounts for grammar, spelling, word choice, and style characteristics. As stated by Martindale and Carpuat, 2018, maintaining text fluency avoids misapprehensions, makes interactions more realistic, and leads to higher user satisfaction and trust. Thus, fluency evaluation is essential for developing better models or screening unacceptable generations.

Measuring fluency is important for evaluating the performance of NLG tasks like summarization, paraphrase generation, image captioning, etc. Fluency evaluation can be done by humans or automated metrics. For manual evaluation, proficiency in the target language is necessary. Besides, it is time-consuming, expensive, and requires a lot of human effort.

Fluency Scale	Example
Perfect Fluency	You must include exercise in your daily routine.
Acceptable Fluency	Delighted with the respsons from the crowd.
Low Quality Fluency	Government clears ... related to land acquisition ... the irrigaton project.
Incomprehensible	Six after dead construction in collapse wall.

Table 1: Examples of classifying sentences as per fluency scale. Misspelt words are highlighted in red and missing words are denoted by ....

To the best of our knowledge, there are no specific automatic evaluation metrics to measure text fluency without reference text. However, researchers use lexical overlap metrics to evaluate text quality in terms of fluency with the help of reference text (Lin and Och, 2004; Papineni et al., 2002). As a result, fluency is often manually assessed, which is expensive, laborious, and irreproducible.

Fluency evaluation of sentences has been a linguistic ability of humans and has been an arguable subject for many decades in linguistics, psychology, and cognitive science. The question has been raised whether the grammatical knowledge underlying this ability is probabilistic or categorical (Chomsky, 1957; Manning, 2003; Sprouse, 2007). In a similar context, Lau et al. 2017 have illustrated that neural language models (LM) can be used to model human acceptability judgments. Kann et al. 2018 proposed **S**yntactic **L**og-**O**dds **R**atio (**SLOR**) score, which leverages *sentence log probability*, normalized by *unigram probability* and *sentence length*,

* Authors contributed equally

to correlate well with human ratings at the sentence level. They investigated the practical implications of Lau et al. 2017's findings for fluency evaluation of NLG, using the task of automatic compression. They also introduced a) WPSLOR: a WordPiece (Wu et al., 2016)-based version of SLOR and b) ROUGE-LM: a combination of WPSLOR and ROUGE (Lin and Och, 2004), the latter being a reference-based fluency evaluation approach. We extend their work by applying the syntactic log odds ratio to the 6 Indo-Aryan[1] and 4 Dravidian languages[2]. Our main motivation is to investigate this approach for morphologically rich languages.

Data scarcity is a very common problem faced by the languages in the Indian subcontinent (Joshi et al., 2020). In addition, the quality of the available datasets is highly questionable. Existing monolingual corpora have several issues like presence of non-unique sentences, junk/unwanted characters; requires additional cleaning and hence there is an overhead of pre-processing and de-noising. Therefore we retrieve clean, filtered data from regional news websites (see Appendix Table 7) and train multiple LMs using RNN (Hochreiter and Schmidhuber, 1997; Cho et al., 2014) and transformer architectures (Devlin et al., 2019) leveraging various embedding techniques (Bojanowski et al., 2017; Heinzerling and Strube, 2018; Kakwani et al., 2020; Khanuja et al., 2021) and compute sentence-level fluency score with syntactic log odds ratio. To assess the quality of the models, it is necessary to compare the scores with humans. However, due to the non-availability of benchmark dataset exclusively for fluency evaluation of the Indian subcontinent languages; we create a corpora for each language along with the designated fluency scores by humans. Proficient native speakers assign these manual scores by following strict guidelines. Table 1 depicts our fluency scale with example sentences. Ultimately we compute the Pearson Product Moment Correlation score of the calculated fluency scores with the human assessments.

Our contributions in this work are in two-fold:

- We release 5K human annotated sentences (500 sentences for each language) which can be further used as a benchmark test-set for fluency evaluation.
- We present our reference-free, unsupervised experiments to measure fluency for 6 Indo-Aryan and 4 Dravidian languages;

[1]Bengali (bn), Gujarati (gu), Hindi (hi), Marathi (mr), Odia (od), Sinhala (si)

[2]Kannada (ka), Malayalam (ml), Telugu (te), Tamil (ta)

2 Existing Approaches

Research in automatic evaluation of NLG systems has increasingly become important. However, the task of measuring *text fluency* remain barely investigated. NLG researchers often use word-overlap based methods to measure text fluency for tasks such as text compression, machine translation, text generation, etc. Existing n-gram overlap metrics like BLEU (Papineni et al., 2002) and ROUGE (Lin and Och, 2004) use reference text to evaluate text quality in terms of fluency.

2.1 Can Readability Scores Measure Fluency?

As text fluency also depends on its vocabulary and sentence structure, we can make an assumption that readability scores can act as a tool to measure fluency. Readability refers to how easy it is to read and understand a text (DuBay, 2004). For many readability measures, the number of syllables is one of the building blocks on which the formula is created. Generally, readability formulas use syllable counting to judge how easy or hard a piece of text is to read. Therefore, any inaccuracies or inconsistencies in counting syllables will have consequences for the accuracy of the readability measurement. In the English readability formula, word length, often measured by the number of syllables, is the key in how difficult or easy a text is to read. In formulas such as Flesch Reading Ease (Kincaid et al., 1975) and Gunning Fog Index (Gunning, 1952), three syllables or more words are categorized as 'hard' words. We cannot apply this in the context of Indian subcontinent languages. In a study of text readability in Bengali, researchers state that in Bengali, polysyllabic words are common in everyday use (Sinha and Anupam, 2014). For example, "Āmādēra" means "ours" in Bengali, which has three syllables and is not a difficult word, but for English readability formula this would count as a complex word. Most of the commonly used words in Indian subcontinent languages are polysyllabic. Hence syllables, both in terms of how to count them and how important they are when you count them, pose a challenge to the development of readability formula for languages of Indian subcontinent.

2.2 LM based Approaches

It is widely believed that much of human cognition is probabilistic (Chater et al., 2006). It is also a

well known fact that a language model is a probabilistic model that assigns probabilities to words and sentences. Therefore, we can safely use Language Model (LM) as a tool to measure fluency. LMs assign higher probabilities to sentences that are syntactically correct and lower probabilities to sentences that are not. Kann et al. 2018 introduced LM-based approaches (SLOR, WPSLOR, ROUGE-LM) to compute the fluency score by normalizing sentence probability with unigram probability and sentence length. They proved that ROUGE-LM showcased better correlation with humans. However, ROUGE-LM is a reference-based approach that demands gold references. Finding reliable references for evaluating the fluency of various NLG tasks is not always possible.

Thus, our research is inclined towards *unsupervised* and *reference-free evaluation* of text fluency. Here, we compute fluency using several combinations of LMs trained using various embeddings. Our work mainly follows the path started by Kann et al. 2018 and performs intensive experiments on 6 Indo-Aryan and 4 Dravidian languages to determine fluency at a sentence level.

3 Reference-Free Sentence-level Fluency Evaluation

In this work, we make an attempt to extend the existing reference-free approach by applying to various morphologically rich languages to get the sentence-level fluency score from the trained language model.

Figure 1 illustrates our approach in detail. Initially, an input sentence is tokenized, and each token is fed to the embedding layer. Using these embeddings, the language model assigns a probability to the given sentence. On the other hand, unigram probability is computed using a list of tokens. To determine sentence fluency, we compute the Syntactic Log-Odds Ratio (SLOR) score by subtracting unigram probability from the sentence probability and further normalizing it by the sentence length (refer Eqn. 1).

3.1 Language Models

Language Modeling (LM) is the task of predicting what word will come next, or, more broadly, a system that computes the likelihood of a sentence (word sequence) or the probability of the next word in a text sequence. The simplest language model is the N-gram, and its performance is restricted by its simplicity and incapable of achieving fluency, sufficient linguistic variation, and proper writing style for extended documents. For these reasons, despite their complexity, neural networks (NN) are being considered as the next primary standard. Recurrent Neural Networks (RNN) have become a base architecture for any sequence. RNNs are currently regarded as the usual text architecture, but they have their own issues: they cannot recall prior content for lengthy periods of time and struggle to construct long relevant text sequences due to exploding or disappearing gradient concerns. As a result, new architectures, such as Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Units (GRU) (Cho et al., 2014), were created and became the state-of-the-art solution for many language generation problems.

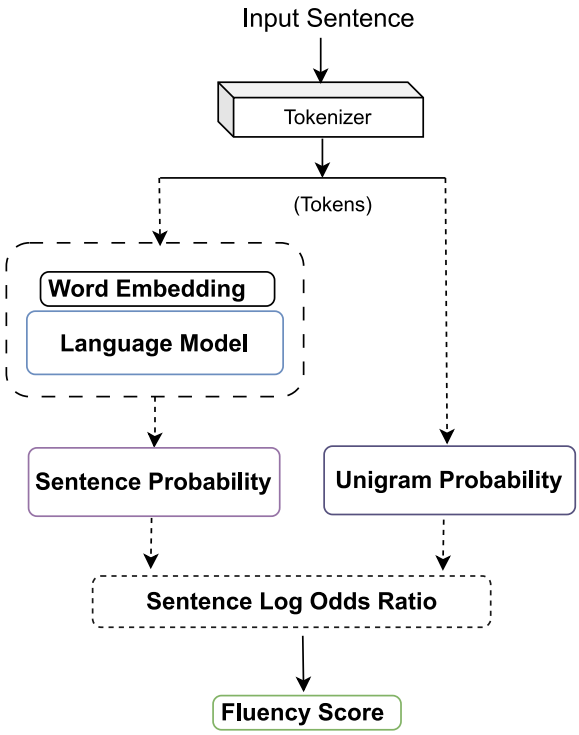

Figure 1: Illustration of our model architecture.

3.2 Text Representation

In this subsection, we discuss the various text representation techniques used in our experiments.

3.2.1 FastText Embeddings

FastText (Bojanowski et al., 2017) is an extended version of the Mikolov et al. 2013's embedding. It is based on skip-gram model and uses subword information to generate the embeddings for a given word. Instead of learning vectors for words directly,

FastText represents each word as an n-gram of characters, which aids it to provide embeddings for rare words.

3.2.2 Byte-Pair Embeddings (BPEmb)

BPEmb (Heinzerling and Strube, 2018) is pre-trained on Wikipedia using the Byte-Pair Encoding technique. Due to the advantage of being subword embeddings, it has the ability to generate embeddings for out-of-vocabulary words.

3.2.3 IndicBERT

IndicBERT (Kakwani et al., 2020) is a multilingual ALBERT (Lan et al., 2019) model trained on large-scale corpora, covering 12 Indian languages. It is pre-trained on AI4Bharat's monolingual corpus containing 8.9B tokens.

3.2.4 MuRIL

MuRIL (Multilingual Representations for Indian Languages) (Khanuja et al., 2021) is a BERT based multilingual language model especially built for 16 Indian languages that is trained using large text corpora of Indian languages.

3.3 Fluency Measure with LM

3.3.1 Syntactic Log-Odds Ratio (SLOR)

SLOR (Kann et al., 2018) assigns a score to a given sentence that is computed using the log-probability of the sentence given by the language model, normalized by its unigram log-probability and length. Sentence SLOR score can be computed using the Equation 1.

$$SLOR(S) = \frac{1}{|S|}(ln(PM(S)) - ln(p_u(S)))$$
$$p_u(S) = \prod_{t \in S} p(t) \quad (1)$$

Where, |S| is the length of the sentence in terms of tokens; PM(S) is the sentence probability returned by the model; $P_u(S)$ is the unigram probability; p(t) denotes the unconditional probability of a token 't' given no context.

3.3.2 WordPiece SLOR (WPSLOR)

WPSLOR is the modified SLOR score computed by considering the word pieces instead of words. WordPieces produce a smaller vocabulary, which reduces model size and training time while improvising the handling of rare words by partitioning them into more frequent segments. However, they contain more information than characters.

4 Dataset Description

4.1 Data Collection

We mined text from web sources for 10 regional news websites (refer Appendix Table. 7) using the Python libraries: Requests[3], BeautifulSoup[4] (Zheng et al., 2015) and Selenium[5]. Figure 2 depicts the data collection pipeline adapted in our work. This data collection is performed in a controlled setting to filter unwanted ads, HTML tags, URLs and other embedded social media content by maintaining a site-specific scraping script. These scripts extract only the news text from sources and avoid most of the noisy data as a primary filter. Later the raw text is cleaned and processed, resulting in a total of 1.025M samples[6]. From the processed and filtered samples, for each language, we distributed 100K samples for training, 1K for validation, 1K for testing. In addition, for the language-wise fluency test set, we translated 500 English samples for human annotations[7] (see Section 4.2).

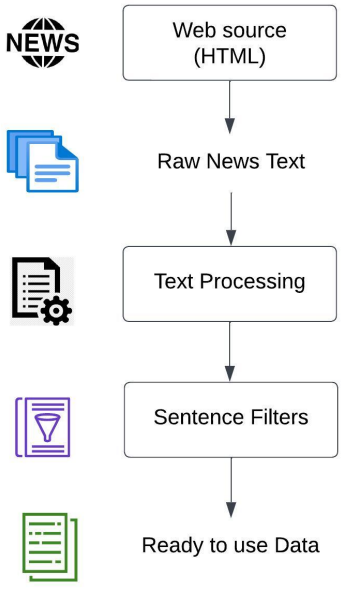

Figure 2: Data collection pipeline

4.1.1 Preprocessing

Data pre-processing is crucial in building any machine learning (ML) model. We build custom pre-processors to extract the sentences from the articles.

[3] https://pypi.org/project/requests/
[4] https://pypi.org/project/beautifulsoup4/
[5] https://pypi.org/project/selenium/
[6] 10 * (100K + 1K + 1K + 500) = 10,25,000
[7] 500 * 10 Languages = 5000

Score	Fluency Quality	Guideline
3	Perfect	Perfectly fluent sentence without any syntactic or grammatical error. **Example**: Government clears issues related to land acquisition of the irrigation project.
2	Moderate	Sentence is predominantly fluent but contains either a) misspelt word or b) missing word or c) multiple occurrence of a word. **Example**: Delighted with the repsonse from the crowd.
1	Low Quality	Partially fluent sentence: a) only half of the sentence is fluent or b) more than 1 missing words or c) more than 1 misspelt words or d) contains individual fluent word-groups with missing coherence between them. **Example**: Government clears ... related to land acquisition ... the irrigatoin project.
0	Incomprehensible	Inarticulate/ non-fluent sentence **Example**: Six after dead construction in collapse wall.

Table 2: Annotation guidelines with fluency scores and examples. Misspelt words are highlighted in red and missing words are denoted by ...

During this process, we clean the sentences by removing special symbols, junk characters, etc. The additional *spaces, tab-spaces and new-lines* are replaced with a single space. The pre-processed sentences are filtered based on the number of tokens at the sentence level. It was also taken care that the sentences do not contain any English text. We consider the sentences having tokens[8] in the range of 8-25 (to avoid too short/long sentences).

4.2 Data Annotation

We collect the human annotated data with the help of proficient native speakers who are computer science graduates with a background in the field of NLP. The annotators were provided with guidelines and were asked to assign a fluency score to each sentence.

4.2.1 Need for Data Annotation

To validate our experiments, we require a standard human annotated test set with fluency scores for each sentence. To the best of our knowledge, there are no human annotated datasets available for measuring fluency of text generated by NLG systems.

4.2.2 Annotation Guidelines

There are no annotation guidelines on assigning a fluency score to a sentence. After careful observation of outputs generated by NLG system, we identified spelling mistakes, redundant words, and coherence issues as common mistakes. With these observations, we penalize the fluency score for the sentence which contains most of the above-mentioned issues. Table 2 details the proposed annotation guidelines for fluency scale.

4.2.3 Statistics of Annotated Data

Native speakers from 10 languages have voluntarily participated in the annotation exercise. Each annotator was made familiar with the guidelines and the process was clearly mentioned. Each individual was assigned with a total of 500 sentences [9] to post-edit and induce errors based on the guidelines. To have a balanced distribution, the annotators were instructed to induce errors in such a manner that the ultimate submission contained 200 sentences with a fluency score of 3 and 100 sentences each with fluency scores of 2, 1, and 0. Therefore, we obtain 40% fluent and 60% non-fluent sentences.

5 Experiments and Discussion

In this section, we describe the various implementation details and discuss the models' performance.

[8] used space as a delimiter

[9] The sentences are translated using Google Translate (https://translate.google.com/)

5.1 Model Implementation

Figure 1, provides the overview of the proposed methodology to measure text fluency. We train an LM on scraped news articles from various news websites (see Appendix Table 7). To train the LM, we use the three variants of the Recurrent Neural Network (RNN) architecture such as LSTM, GRU and Bi-LSTM with the four different embedding techniques: FastText, BPE, IndicBERT and MuRIL embeddings. All the models are trained with 1 GPU support. Each RNN variant has 2 layers with 512 hidden activation units in each layer and a sequence of 128. To tune the LM parameters, we employ Categorical Cross Entropy as the loss function and Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001. We adopt early stopping criteria based on the validation loss as a regularization step to avoid over-fitting.

5.2 Zero-shot and Fine-tuned Experiments

We additionally experimented using existing pre-trained multilingual BERT LM[10] (Devlin et al., 2019) and Muril LM[11] (Khanuja et al., 2021).

Using these models, we performed zero-shot inferences and also fine-tuned them. The pre-trained model is fine-tuned on the same training data for 5 epochs with a learning rate of $2e^{-5}$, batch size of 8 and AdamW optimizer (Loshchilov and Hutter, 2017).

5.3 Results and Analysis

Using these various trained and fine-tuned models, we independently compute fluency by calculating the SLOR score. To evaluate the models, we compute the Pearson product-moment correlation (C) (Benesty et al., 2009) between fluency scores (F) and human ratings (H) using Equation 2, where, N is the total number of samples (500 sentences). Pearson correlation estimates the degree of statistical relationship. Consequently, it can be logically reasoned that if a model has a high correlation with human judgments, it implies that the model correlates well with humans.

$$C = \frac{(N\sum_{i=1}^{N} H_i F_i - (\sum_{i=1}^{N} H_i)(\sum_{i=1}^{N} F_i))}{\sqrt{N\sum_{i=1}^{N} H_i{}^2 - (\sum_{i=1}^{N} H_i)^2}\sqrt{N\sum_{i=1}^{N} F_i{}^2 - (\sum_{i=1}^{N} F_i)^2}} \quad (2)$$

The correlation of fluency scores of RNN-based LMs with human judgments is reported in Table 3 and Table 4. The findings from the reported tables indicate that the Muril+LSTM model performed the best, followed by RNNs trained using BPEmb. This is supported by the language-wise comparison of correlation scores with humans shown in Figure 3, in which we compare our best-trained model with zero-shot inferences and fine-tuned model. Both the *Muril+LSTM* model and the *Muril model with fine-tuning* demonstrated superior performance compared to other models.

It is worth noting that the effectiveness of the models is directly linked to the amount of data on which they were trained. This is evident from Figure 4, which illustrates the size of the Wikipedia data used for training BPEMb, mBERT, and MuRIL models in terms of the number of articles. Therefore, models trained using MuRIL exhibit improved performance.

Another noteworthy observation is that the Muril+LSTM model, despite being trained on a smaller dataset due to limited computational resources, performed on par with the fine-tuned MuRIL model. It is reasonable to assume that an increase in data size and/or the number of training parameters would further enhance the model's performance, resulting in improved correlation with human judgments.

5.4 Challenges

One area where our models encountered challenges was in handling sentences that contained digits and abbreviations. Due to the complexity of these elements, which often carry specific meanings and context, our models struggled to evaluate the fluency of these outputs. The limited training data and the inherent complexity of digit and abbreviation recognition posed difficulties for the models to effectively capture the intended semantics.

6 Limitations

Given the constraints of limited computation power, we have trained our models using base configurations consisting of 12 layers. While deeper archi-

[10] https://huggingface.co/bert-base-multilingual-uncased. Not available for Odia and Sinhala

[11] https://huggingface.co/google/muril-base-cased. Not available for Sinhala

E	FastText			BPEmb			IndicBERT			MuRIL		
Ln	Bi-LSTM	LSTM	GRU	Bi-LSTM	LSTM	GRU	Bi-LSTM	LSTM	GRU	Bi-LSTM	LSTM	GRU
te	0.317	0.355	0.313	0.403	0.410	0.327	0.307	0.290	0.270	0.400	**0.430**	0.280
ml	0.119	0.123	0.089	0.328	0.279	0.213	0.145	0.182	0.103	0.300	**0.300**	0.170
ta	0.152	0.124	0.134	0.085	0.130	0.098	0.039	0.019	0.020	0.110	**0.140**	0.080
kn	0.201	0.145	0.214	0.381	0.357	0.264	0.398	0.410	0.343	**0.550**	0.520	0.450

Table 3: Pearson correlation scores of RNN-based LMs with human judgements for Dravidian Languages.
E: Embeddings, Ln: Language

E	FastText			BPEmb			IndicBERT			MuRIL		
Ln	Bi-LSTM	LSTM	GRU	Bi-LSTM	LSTM	GRU	Bi-LSTM	LSTM	GRU	Bi-LSTM	LSTM	GRU
bn	0.191	0.202	0.164	0.335	0.284	0.296	0.171	0.176	0.168	0.308	**0.350**	0.260
od	0.030	0.070	-0.010	0.190	0.234	0.240	0.217	0.230	0.234	0.200	**0.200**	0.140
hi	0.188	0.216	0.126	0.459	0.456	0.438	0.481	0.488	0.478	0.579	**0.600**	0.560
gu	0.377	0.381	0.305	0.432	0.371	0.321	0.282	0.286	0.221	0.474	**0.500**	0.340
mr	-0.022	0.008	0	0.403	0.374	0.325	0.515	0.484	0.407	0.500	**0.500**	0.450
si	0.359	0.348	-0.110	0.411	0.422	0.407	-	-	-	-	-	-

Table 4: Pearson correlation scores of RNN-based LMs with human judgements for Indo-Aryan Languages
E: Embeddings, Ln: Language

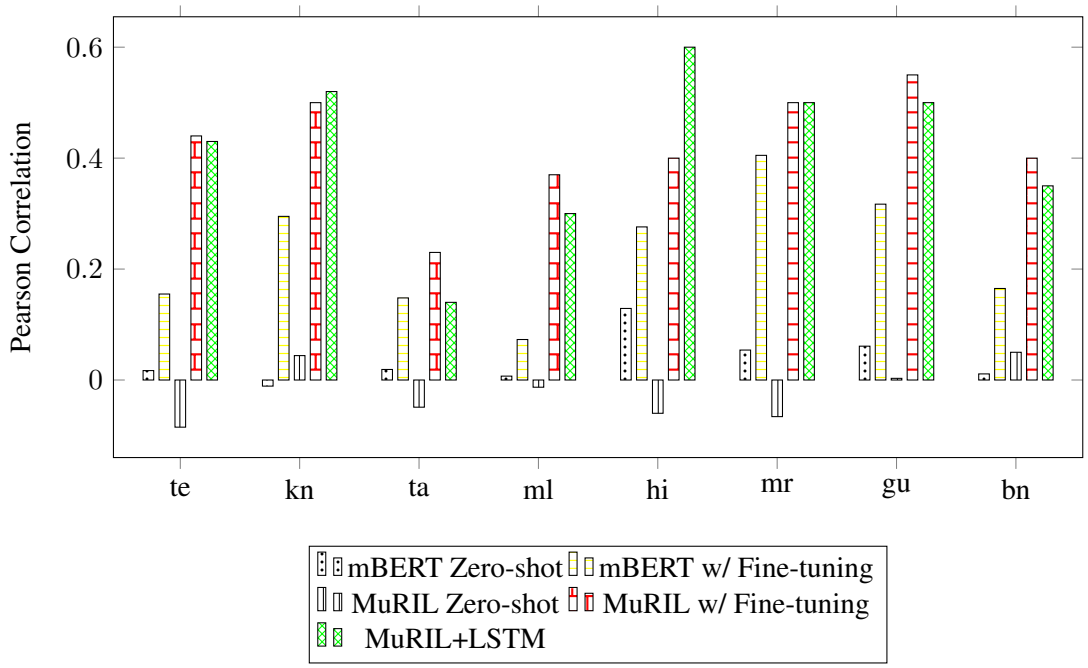

Figure 3: Comparative analysis of correlation scores of various models. (The scores of Oriya and Sinhala are not reported)

tectures are often desirable for achieving higher performance, we have made the most of the available resources to train these models effectively. Also, due to human resource limitations, we could not explore other Indic languages in creating a benchmark test set for fluency. However, we recognize the significance of expanding the scope to encompass additional Indic languages in future research endeavors.

7 Ethical Statement

We scrape regional news websites for source articles, under a fair usage policy. The copyright of the original news articles remains with the original authors/publishers of the articles.

8 Conclusion and Future Work

This paper presents our experiments on *unsupervised reference-free fluency evaluation at the sentence level*. Text fluency is the ability to read words

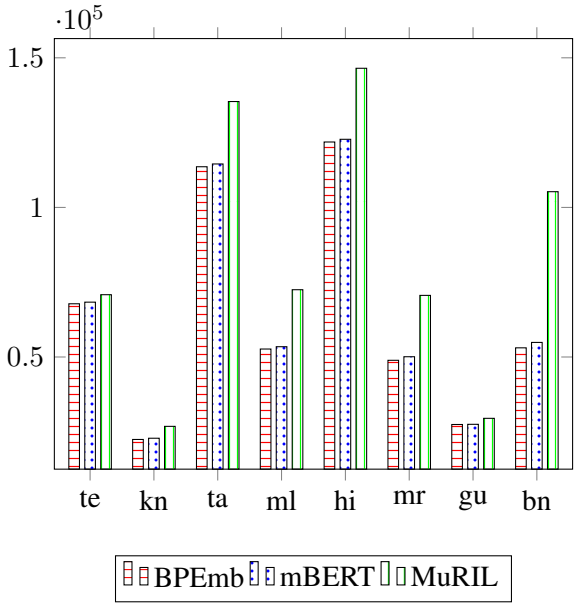

Figure 4: Language-wise count of Wikipedia articles used in training the models

without any apparent cognitive effort and is a critical gateway to comprehension. Fluency evaluation is very important and a fundamental step to be taken for evaluating the outputs of NLG systems. Fluency evaluation of generated text further helps to improve the models or filter unacceptable generations. Despite of text fluency evaluation being a very significant task, it is often ignored. Kann et al. 2018 made an initiative in this direction by proposing reference-free and reference-based approaches to measure sentence-level fluency. This paper builds on their reference-free approach by investigating on various languages of Indian subcontinent. We use *SLOR* to compute the fluency scores. We explored several models by a) training RNN-based LMs leveraging various embeddings, b) using mBERT and MuRIL for zero-shot inferences and by further c) fine-tuning mBERT and MuRIL. We performed 100+ experiments for 10 Indic languages and identified the best models. Due to the non-availability of test sets for sentence-level fluency evaluation, we collected 5K human-annotated benchmark data for 10 languages (500 samples per language). We evaluated our test results by correlating them with human judgments. Our exploration reveals that RNN-based language models trained with MuRIL embeddings stood as the winner in computing fluency scores. Also, fine-tuned MuRIL models performed better in terms of correlation with humans. It is observed that, even with lesser training data and model parameters, our model (MuRIL+LSTM) produced significant results compared to other models. Our experiments indicate that assessing the fluency of machine-generated text can be accomplished solely by the language model, without the need for references.

We believe that our pioneering work will act as a stepping stone for further research in this direction. This approach can be applied to other downstream tasks of NLG, such as Summarization, Paraphrase Generation, Translation, Dialogue Generation, Image Captioning, etc.

Further, we plan to extend our work to perform fluency evaluation at the discourse level and aim to release a larger human-annotated corpus as benchmark data for fluency evaluation.

References

Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. 2009. *Pearson Correlation Coefficient*, pages 1–4. Springer Berlin Heidelberg, Berlin, Heidelberg.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Nick Chater, Joshua Tenenbaum, and Alan Yuille. 2006. Probabilistic models of cognition: Conceptual foundations. *Trends in cognitive sciences*, 10:287–91.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Noam Chomsky. 1957. *Syntactic Structures*. De Gruyter Mouton.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William H DuBay. 2004. The principles of readability. *Online Submission*.

Robert Gunning. 1952. The technique of clear writing. McGraw-Hill.

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.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the nlp world.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. 2020. Indicnlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961.

Katharina Kann, Sascha Rothe, and Katja Filippova. 2018. Sentence-level fluency evaluation: References help, but can be spared!

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, et al. 2021. Muril: Multilingual representations for indian languages. *arXiv preprint arXiv:2103.10730*.

J. Peter Kincaid, Robert P. Fishburne, R L Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2017. Grammaticality, acceptability, and probability: A probabilistic view of linguistic knowledge. *Cognitive Science*, 41(5):1202–1241.

Chin-Yew Lin and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, Barcelona, Spain.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization.

Christopher Manning. 2003. Probabilistic syntax.

Marianna Martindale and Marine Carpuat. 2018. Fluency over adequacy: A pilot study in measuring user trust in imperfect MT. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 13–25, Boston, MA. Association for Machine Translation in the Americas.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Joss Moorkens, Sheila Castilho, Federico Gaspari, and Stephen Doherty. 2018. *Translation Quality Assessment From Principles to Practice*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Manjira Sinha and Basu Anupam. 2014. A study of readability of texts in bangla through machine learning approaches. *Education and Information Technologies*, 21.

Jon Sprouse. 2007. Continuous acceptability, categorical grammaticality, and experimental syntax. *Biolinguistics*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation.

Chunmei Zheng, Guomei He, and Zuojie Peng. 2015. A study of web information extraction technology based on beautiful soup. *J. Comput.*, 10:381–387.

A Appendix

Sentence	Human Rating	Predicted Score	Description
Viśākhapaṭnanlō konasāgutunna nirudyōga saṅghāla āmaraṇa nirāhāra dīkṣa *The ongoing hunger strike of the unemployed unions in Visakhapatnam.*	3	3	Perfectly fluent sentence
Gāya kāraṇaṅgā bhuvanēśvar kumār **Aipi'el** nuṇḍi tappukunnāḍu *Bhuvneshwar Kumar ruled out of **IPL** due to injury.*	2	2	One missing word (Noun)
Prastuta paristhitullō prajalu kōviḍ 19 sūcanalu **pāṭicaḍa** tappanisari ani rāṣṭra mukhyamantri telipāru *The Chief Minister of the state said that it is mandatory for people to **follow** the instructions of Covid-19 in the current situation*	2	2	One misspelt word (Verb)
Pradhāni nirṇaya paṭla santōṣaṅgā unna strīlu varṣa kāraṇaṅgā āṭa raddu *Women are happy with PM's decision game canceled due to rain*	1	1	Coherence missing between word-groups
Mumbai iṇḍiyans pai arṣdīp siṅg 2 vikeṭs tīsi manci spelvēsāḍu **ayitē kaligi yuva pēsar unnāḍu sthiraṅgā uṇḍāli** *Arshdeep Singh bowled a good spell with 2 wickets against Mumbai Indians **but a young pacer is there should be consistent.***	1	1	Half of the sentence is fluent
Anēka atipedda cainā prapancanlōnē sanvatsarālugā vastuvula undi sarapharādārugā Many biggest China in the world for years of goods there is as a supplier	0	0	Non-fluent sentence
Rahīm **1950** nuṇḍi **1963**lō maraṇincē varaku bhārata jātīya phuṭbāl jaṭṭuku mēnējargā unnāru *Rahim was the manager of the Indian national football team from 1950 until his death in 1963*	3	2	Failed to perform well for sentences having digits
Enī'ī'ār'ai cē abhivrd'dhi cēyabaḍina ī sākētita ī kaṣṭa samayāllō mukhyamainadi	2	1	Failed to perform well for sentences having abbreviations

Table 5: Comparision of model output with humans for Telugu

Language Family	*Language*	Data Split	*Total tokens (thousands)*	*Unique tokens (thousands)*	*Avg tokens per sentence*
Dravidian	Kannada	*Train*	1387	125	13.87
		Test	13.7	5.7	13.76
		Validation	14	5.7	14.03
	Malayalam	*Train*	1173	233	11.73
		Test	11.7	7	11.76
		Validation	12	7	11.97
	Tamil	*Train*	1348	127	13.49
		Test	13.6	6.3	13.65
		Validation	13	5.9	13.45
	Telugu	*Train*	1224	142	12.24
		Test	*12*	*6*	12.24
		Validation	*12*	*6*	12.2
Indo-Aryan	Bengali	*Train*	1308	69	12.96
		Test	13	4.5	13.02
		Validation	13	4.5	12.93
	Gujarati	*Train*	1461	128	14.61
		Test	14	5.3	14.43
		Validation	14.7	5.3	14.8
	Hindi	*Train*	1701	60	17.01
		Test	16.7	4	16.72
		Validation	16.8	3.8	16.85
	Marathi	*Train*	1353	104	13.53
		Test	13	5	13.27
		Validation	13.6	4.8	13.69
	Odia	*Train*	1427	80	13.37
		Test	14	4.5	13.11
		Validation	14.5	4.5	13.47
	Sinhala	*Train*	1591	75	16.31
		Test	16	5	16.29
		Validation	16	5	16.42

Table 6: Data Statistics

SNo	**Language**	**Website**
1	Telugu	https://www.vaartha.com/
2	Hindi	https://www.indiatv.in/
3	Tamil	https://www.updatenews360.com/
4	Kannada	https://kannadanewsnow.com/kannada/
5	Malayalam	https://dailyindianherald.com/
6	Bengali	https://www.abplive.com/
7	Gujarati	https://www.gujaratsamachar.com/
8	Marathi	https://www.abplive.com/
9	Odia	https://www.dharitri.com/
10	Sinhala	https://www.newsfirst.lk/sinhala/

Table 7: News articles crawling sources

	LSTM + Muril	**Muril w/ finetuning**	**mBERT w/ finetuning**
Sequence Length	128	128	128
Embedding dimension	768	768	768
Learning Rate	0.001	$2e^{-5}$	$2e^{-5}$
Batch size	256	8	8
Epochs	5	5	5
hidden units	512	-	-
# Layers	2	12	12

Table 8: Model Hyper Parameters