International Journal of Advances in Applied Sciences (IJAAS)
Vol. 15, No. 2, June 2026, pp. 687~695
ISSN: 2252-8814, DOI: 10.11591/ijaas.v15.i2.pp687-695

Assessment of deep learning based Hindi Odia bidirectional machine translation system

Subhashree Satpathy[1], Smitaprava Mishra[2], Ajit Kumar Nayak[2]
[1]Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India
[2]Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India

Article Info

Article history:

Received Oct 31, 2025
Revised May 1, 2026
Accepted May 22, 2026

Keywords:

Deep learning
Evaluation metrics
Indic languages
Machine translation
Natural language processing

ABSTRACT

India is a vast nation with a diverse range of cultures and languages. Most Indians choose to use their native languages when communicating with machines. To integrate smart technologies into every facet of Indian society, efficient systems that can positively identify Indian languages must be established. Machine translation (MT) studies comprise most of the natural language processing (NLP) in the era of multilingual computer-human interaction. Till now, less emphasis has been placed to develop MT systems among Indian languages. Yet again, building a qualitative and quantitative corpus in these languages is challenging. This work focuses on two Indic languages for the development of a Hindi to Odia bidirectional machine translation system (HOBMT). Bilingual evaluation understudy (BLEU), word error rate (WER), character error rate (CER), and metric for evaluation of translation with explicit ordering (METEOR) evaluation metrics are used to assess the accuracy of the translation. The most advanced sequential deep learning (DL) models, such as recurrent neural network (RNN), long short-term memory (LSTM), and gated recurrent unit (GRU), are used in this study. In this research, RNN is observed with improved translation results due to its sequential data handling with context preservation.

Corresponding Author:

Subhashree Satpathy
Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed to be University)
Bhubaneswar, Odisha 751030, India
Email: subhashreesatpathy16@gmail.com

1. INTRODUCTION

Machine translation (MT) is known as a portion of natural language processing (NLP) which automatically converts text or speech between different languages using algorithms, statistical models, or neural networks to achieve accurate and context-aware translations. India has a wide variety of spoken languages, where more than a thousand dialects and at least 30 distinct languages coexist. MT systems are vital for overcoming language barriers and providing developments like Digital India. Sequence-to-sequence (seq2seq) models have considerably enriched the translation quality by capturing language data contextual dependencies in deep learning (DL) techniques [1], [2]. Indian languages that lack computational models, annotated datasets, linguistic tools, and other digital resources for tasks like NLP are referred to as low-resource Indic languages [3]. When compared to international languages with abundant resources like English or French, these low-resource languages frequently lack sufficient corpora, dictionaries, and research attention. Examples including Manipuri, Konkani, Santali, Bodo, and Odia [4], [5].

Developing resources for these languages is vital for preserving linguistic diversity and aiding access to technology for speakers of such languages. Hindi is an Indo-Aryan language spoken primarily in India and globally, with over 600 million speakers, including native and second-language speakers. India's

literature, culture, media, and education all depend on it [6]. It is written in the Devanagari script and has its roots in Sanskrit, with significant influences from Persian, Arabic, and English. Hindi is one of the official languages of the government of India. The Indo-Aryan language known as Odia is mainly spoken in the state of Odisha and its neighboring regions. It is one of India's 22 scheduled languages and was designated a 6th classical language of India due to its rich literary heritage and historical significance. It has its own script and has a history over more than 1,000 years. Odia is renowned for its contributions to Indian culture, devotional poetry, and ancient texts. This language, spoken by more than 45 million people, is an important part of the region's identity, education, and government [7]. Many Odisha citizens still have trouble grasping information written in Hindi, despite Hindi being the official language of India. In this case, a suitable MT system from Hindi to Odia will enable people to comprehend and utilize Hindi more creative way [8], [9].

Rule-based MT, statistical MT, neural MT, example-based MT, and hybrid MT are the categories of MT systems [2], [4]. Neural MT has demonstrated promise in handling sequential data and enhancing translation accuracy, especially with recurrent architectures like recurrent neural network (RNN) [10], long short-term memory (LSTM) [11], and gated recurrent unit (GRU) [12]. Although parallel corpora for Indic languages have been expanded by recent initiatives like the Samanantar dataset, noise and inconsistency problems still exist [13]. DL models for Hindi-Odia translation have not yet been thoroughly evaluated, though MT for Hindi-English or other Indic pairs has been the subject of numerous studies. In countries with multiple languages, like India, it is required to have a lot of high-quality parallel datasets for Indian languages to build low-resource Indic MT systems [14]. Research on bidirectional Hindi-Odia MT systems is simply lacking. So, comparative analysis of DL models on such datasets is rare. It is going to be difficult to identify the best architectures for low-resource Indic languages without trustful evaluation. However, researchers have developed the Samanantar dataset, a vast parallel dataset collection for Indian languages.

In this work, Hindi and Odia language datasets are considered to build a Hindi to Odia bidirectional machine translation system (HOBMT). The first step of this work to create a Hindi-Odia parallel corpus of over 60,000 sentence pairs. It investigates how different DL methods like RNN, LSTM, and GRU perform on Hindi-Odia bidirectional translation. All the translations are evaluated using standard evaluation metrics, i.e., bilingual evaluation understudy (BLEU) [15], metric for evaluation of translation with explicit ordering (METEOR) [16], word error rate (WER) [17], and character error rate (CER) [18]. Translation quality is evaluated with varying parameter settings, like learning rate, batch size, and epochs, on the translation performance. It categorizes the benefits and challenges of each model in processing simple and short sentences of a low-resource dataset. This work focuses on building Hindi-Odia MT system along with dataset creation. It emphasizes the importance of applied sciences and information technologies by tackling issues in low resource languages and advancing cross-lingual communication technologies in the digital ecosystem. Moreover, it promotes digital inclusion by providing speakers of underrepresented languages with improved access to government services, digital content, engineering, information technology, and different educational materials by concentrating on Hindi-Odia MT system.

The rest of the article is elaborated as follows. Section 2 is the primary part of the method, comprised with the dataset collection and preparation, different techniques, including hyperparameter settings with flow of work. Section 3 unfolds the experimental results and discussions. Section 4 concludes the work and highlights its future perspectives.

2. METHOD

2.1. Dataset preparation

It is easy to develop a bilingual corpus for resource-rich language pairs, like English, Chinese, Japanese, German, and so forth, because there are plenty of resources available for corpora collection. There are so many websites, books, government, and institutional datasets available for data collection [19]. For data preparation, it is important to have an extensive dataset. Preprocessing and cleaning of these data are essential to maintain that the models can train effectively. The raw data from the language includes not only a lot of data from other languages but also other random characters. The length of the sentences also varies significantly. Thus, it needs to be standardized to train a sound model. Downloading contents from websites, aligning of documents and sentences, and post-processing are the steps need to be followed to prepare the dataset. Besides this, depending on the traits of each language pair, a few appropriate processing steps can be added [20], [21].

Developing an MT system in Indian language is a very tough task. Because it has multiple issues like script complexity, morphological richness, lack of standardization, speech recognition issues, data scarcity, and limited research support. Even though there are numerous major recent advancements in corpus development, more efforts should be made to create large-scale diversified dataset for all Indic languages [13].

Indian institutions, researchers, and organizations have begun developing MT systems for Indian languages and have achieved satisfactory results [22]. But Hindi-Odia is a low-resource language pair, and developing a bilingual corpus is difficult. Very less bilingual electronic documents are present in online platform. The work is to build a Hindi to Odia bilingual MT system using different DL methods. The authors collected more than one lakh Hindi to Odia sentence pairs. The collection process is quite tedious, and the verification process makes it even more difficult. Human validation is needed for each source to target language alignment. Proposed datasets are collected from the website "https://www.manythings.org/anki/". The datasets are in international language form [23]. The English sentences have been translated manually into two different Indic languages, Hindi and Odia. Table 1 represents one example of proposed dataset.

Table 1. Example of proposed dataset

English	Hindi	Odia
You need to grow up	तुम्हें बड़ा होने की जरूरत है (*Tumhean baḍaā hone kī jarūrat hai*)	ତୁମେ ବଡ ହେବା ଆବଶ୍ୟକ (*Tume bada heba abashyaka*)
You need to help me	तुम्हें मेरी मदद करनी होगी (*Tumhean merī madad karanī hogī*)	ତୁମେ ମୋତେ ସାହାଯ୍ୟ କରିବା ଆବଶ୍ୟକ (*Tume mote sahajya kariba abashyaka*)
You need to wake up	आपको जागने की जरूरत है (*Āpako jāgane kī jarūrat hai*)	ତୁମେ ଜାଗ୍ରତ ହେବା ଆବଶ୍ୟକ (*Tume jagrata heba abashyaka*)
You never helped me	तुमने कभी मेरी मदद नहीं की (*Tumane kabhī merī madad nahīan kī*)	ତୁମେ ମୋତେ କେବେ ସାହାଯ୍ୟ କରି ନାହଁ (*Tume mote kebe sahajya kari nahan*)

2.2. Training dataset

Data preparation is a fundamental step in any machine learning task, comprising cleaning, transforming, and organizing raw data. To train HOBMT model, it is ensured that Hindi and Odia data are properly pre-processed (tokenized, padded, sentence lengths, and vocabulary size). The tokenizer object can convert all the words of a dataset into unique numerical index. Parallel sentence pair of Hindi-Odia and Odia-Hindi are tabulated in Table 2. The total dataset consists of 60,000 parallel sentences, of which 72% (43,200) is allocated for training, 8% (4,800) for validation, and remaining 20% (12,000) are used for testing. Hindi has 10078, and Odia has 11321 tokens for this particular system. Figure 1 compares the distribution of a particular feature like sentence length, of the Hindi and Odia datasets by displaying two histograms side by side. Particularly in contrast to the Hindi sentence lengths, the Odia sentences are in maximum frequency. So that the sentence length is taken up to 8. Sentence sequences are converted into integer sequences. To make the sentences of the same length, padding is applied so that zeros are added to the end of the sequence.

Table 2. Dataset splits

Category	Value
Total parallel sentences	60,000
Training + validation	48,000 (43,200 + 4,800)
Testing	12,000
Hindi vocabulary size	10078
Odia vocabulary size	11321

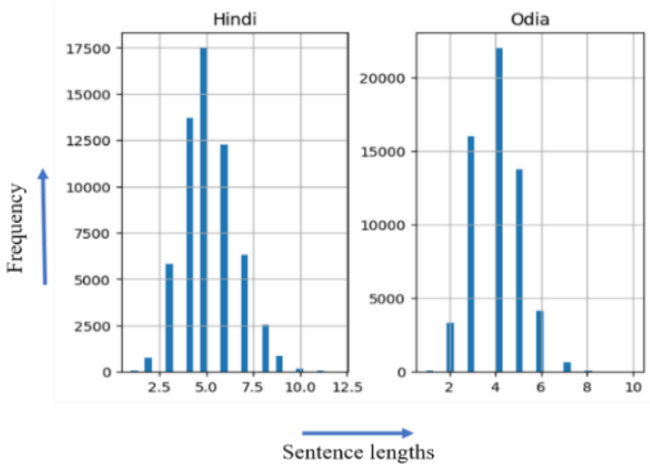

Figure 1. Histogram of Hindi and Odia dataset

2.3. Deep learning models

By enhancing neural network architecture, many complex problems in DL can be resolved. In seq2seq learning, the RNN and its variations are very helpful. The most popular seq2seq techniques, like LSTM and GRU [10]–[12] are described in the sections.

2.3.1. Recurrent neural networks

The shortcomings of conventional neural networks, such as feedforward neural networks (FNNs), in processing sequential data were addressed by the introduction of RNNs. FNNs are inefficient at capturing dependencies because they process inputs independently, disregarding context and order. By adding recurrent connections, RNNs get around this and enable data to move between time steps. In order to learn temporal dependencies and manage variable-length inputs, RNNs can retain internal memory, where each output is fed back as input to the subsequent step [12], [24]. RNNs primarily use recent data when processing lengthy sentences because the vanishing gradient problem during backpropagation causes earlier information to fade. The exploding gradient problem, on the other hand, can also result from gradients growing exponentially. GRUs and LSTMs were developed to overcome these issues, enhancing long-term memory retention and stabilizing training [25].

2.3.2. Long short-term memory

This model was developed in 1997 to identify long-term dependencies in sentences [26]. Using a cell state and three gates, "forget, update, and output" to regulate information flow. It overcomes the drawbacks of RNNs. These three gates employ sigmoid activations, with values ranging from 0 to 1, retain relevant data, and discard the rest. The output gate regulates the subsequent hidden state, the forget gate chooses what should be kept, and the update gate decides when to update the memory [27].

2.3.3. Gated recurrent unit

GRU has two gates rather than three gates and fewer parameters, which simplifies LSTM. By combining the input and forget gates into a single "update gate," it ensures that crucial information is preserved over time. While the reset gate regulates the amount of historical data considered, the update gate establishes the final cell state. Previous data is ignored if the reset gate is close to zero and information is maintained if the update gate is close to one [11], [28]. Some key differences have been found from three models, which are discussed as follows: i) RNNs are vulnerable to vanishing and exploding gradients due to its basic structure; ii) LSTMs employ distinct cell states and gates for more precise control over the information flow; and iii) GRU uses fewer parameters and is an alternative to LSTM, commonly generating comparable performances.

Proper parameter setting is crucial for successful outcomes because adjusting the values of parameters can affect how quickly, accurately, and efficiently the program runs. Some common factors used to define the discussed model, like size of input and output vocabulary, length of input and output sequences as well as the number of hidden units. The discussed models are compiled with different parameters, which is given in Table 3.

Table 3. Parameter settings

Parameter	Value
Optimizer	Root mean square propagation (RMSProp)
Loss function	Sparse categorical cross entropy
Learning rate	0.001, 0.01
Hidden states	512
Batch size	128, 256, 512
Epochs	30, 60, 100
Maximum length of both sentences	8

2.4. Evaluation metrics

For any comparative study, evaluation must be rapid, precise, economical, and time-efficient. The system is trained using the same corpus of language pairs to guarantee that the evaluation is consistent across all models. In this case, BLEU [15], METEOR [16], WER [17], and CER [18] scores have been used to guide the evaluation and comparison of the systems.

BLEU metric measures the similarity between a machine-translated text to a reference translation. This score range lies between 0 to 1. Where 0 indicates no match and 1 indicates perfect match between the predicted sentence and reference sentence. It is more completely dependent on precision. It ignores similar

words and searches for exact words. The METEOR score computes the harmonic mean using weighted precision and recall. Since it concentrates on correlation with human translations at the segment level, whereas the BLEU score focuses on correlation at the corpus level, this score resolves a number of issues with the BLEU score. WER computes the ratio of errors on substitutions, insertions, and deletions in the system's output to the total number of words in the reference text. CER calculates the distance between the machine-translated text, which is known as the candidate text and the correct, human-translated text, which is known as the reference text at the character level. For training purposes, graphics processing unit (GPU) is an excellent substitute for the central processing unit (CPU) in terms of handling and speedy computation. It solves the problem of complex computation and leads the system towards good performance, supporting massive parallel computation. Almost one epoch takes less than one minute using the system configuration depicted in Table 4.

Table 4. System configuration

Component	Specification
Python 3 Google Compute Engine backend (GPU)	T4GPU
System RAM	12.7 GB
GPU RAM	15.0 GB
Disk	112.6 GB
Processor	Intel Core i5

2.5. Workflow diagram

The choice of selecting between these models is primarily based on the need of task including the computational resources that are available, the complexity of the dependencies, and the sequence length. Sequential models are simple method of building neural networks by stacking layers linearly, and the output of each layer feeding directly into the next layer [29]. Figure 2 represents the workflow to establish HOBMT system, including several distinct steps like parameter setting, training dataset, and model evaluation, and finally to save the best models.

Figure 3 is an illustration of the proposed models using an encoder-decoder mechanism. The model receives input sequences with one word for each time step. To match the vocabulary in the target dataset, each word is encoded as a distinct integer. Every word is converted into a vector using embedding and the vector size depends on the level of vocabulary complexity. In encoder layer, the current word vector is given the context from word vectors in earlier time steps. Dense layer, where the encoded input is decoded into the appropriate translation sequence using these fully connected layers. The outputs can be mapped to the target dataset vocabulary after being returned as a series of integers or encoded vectors [25], [30]. The RepeatVector layer repeats this context vector over time, feeding it to the decoder. Then the second layer of discussed models acts as the decoder. The dense layer with SoftMax activation outputs the probability distribution over the output vocabulary [31].

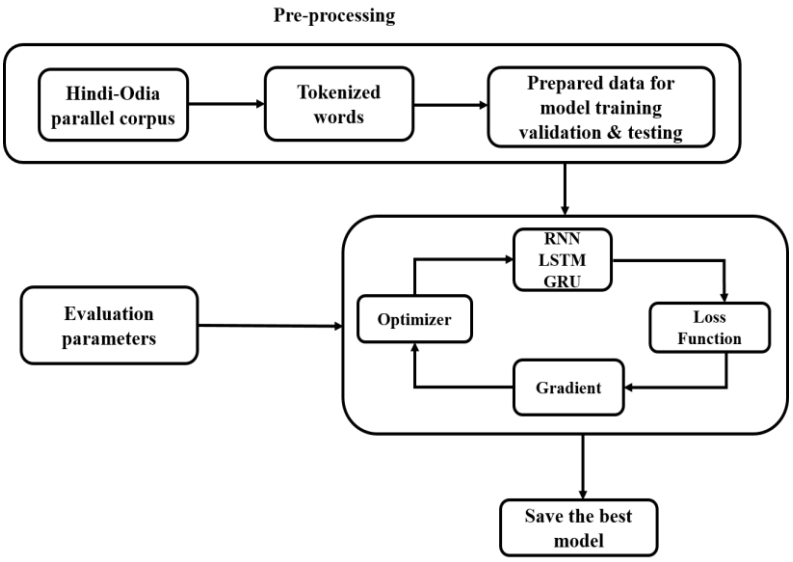

Figure 2. HOBMT system model

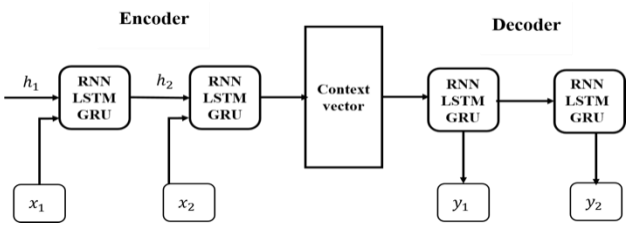

Figure 3. Proposed MT system using an encoder-decoder mechanism

Mathematically, it takes the current input (x_t) and the memory of the past or the previous hidden state $h_{(t-1)}$ to calculate a new memory or new hidden state (h_t). Then the new memory (h_t) is used to produce an output (y_t). The weight matrix $W_{hh}$is connecting from previous hidden state to the current hidden state and W_{xh} is connecting the current input to the current hidden state. (b_h) is known as the bias vector of hidden states. $tanh$ is the hyperbolic tangent activation function, which squashes the values between -1 and 1. The (1) and (2) can be used to represent the fundamental structure of an RNN.

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \quad (1)$$

The output (y_t) at the current time step "t" is known as probability distribution or prediction. The weight matrix (W_{hy}) is matrix connecting the hidden state to the output and the output bias vector is (b_y) which is calculated as in (2).

$$y_t = W_{hy}h_t + b_y \quad (2)$$

3. RESULTS AND DISCUSSION

Reliability, robustness, and effectiveness on a variety of tasks are used to evaluate the overall performance of the MT system. A detailed evaluation not only helps monitor a model's potential but also identifies areas that need some work. Figure 4 signifies the 30-epoch results of HOBMT system. Typically, BLEU score is accepted to automatically evaluate the quality of translation. Score values are observed at 30-epochs. This indicates that the simple RNN's BLEU score outperforms LSTM and GRU for the HOBMT translation tasks for the current dataset and training setup. Some analysis and key problems which is defined that RNN has achieved good results in comparison to LSTM and GRU.

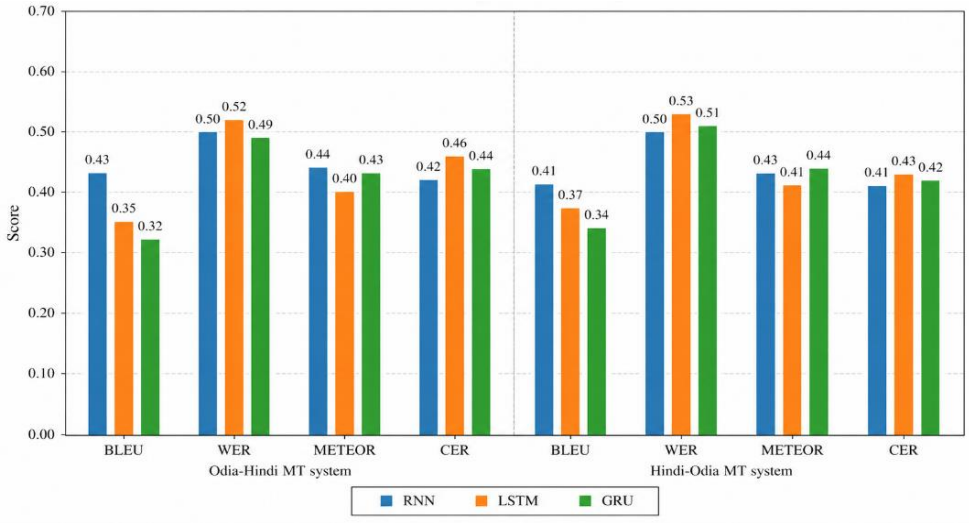

Figure 4. Performance comparison of HOBMT systems

3.1. Sentence lengths

The sentence lengths are small and simple within this experiment. RNN generates text sequentially by predicting the next character or word based on their prior sequences. LSTM and GRU cells are designed

to capture long-range dependencies in sequences, which is not suitable for short and simple sentences. RNN approach has given rise to a new optimism, producing better results for shorter sentences instead of longer ones. Evaluation metrics score of RNN network is more than 40% as compare to LSTM and GRU.

3.2. Remembering and reasoning

The continuous vector representation for all these three networks is incapable to recall the entire source sentence. Obtaining the correct target translation through decoding from this representation is quite challenging. Additionally, MT is a more complex problem that calls for rich reasoning operations, unlike other seq2seq NLP tasks.

3.3. Computational complexity

Deep neural network model training is a time-consuming task. It specifies the mathematical growth rate of a particular algorithmic resource. Training neural machine translation (NMT) system can take much time or memory, even with powerful GPUs.

3.4. Error analysis

This analysis reveals the specific linguistic patterns that the model is not capturing and the reason for the low score. It is challenging to comprehend why these NMTs improve translation performance or why it fails because it works with variables in the real-valued continuous space. Error analysis contributes to better model performance by highlighting flaws, model tuning, boosting overall accuracy, and addressing data cleaning. Test case results are observed at 30th epoch of this dataset is displayed in Table 5. One mistranslation example from Table 5 is actual- ମୁଁ ଯିବାକୁ ଚିନ୍ତା କରୁଛି (*mun jibaku cinta karuchi*) = predicted- ମୁଁ ଯିବାକୁ (*mu jibaku*), where the source and target word orders are incorrect. Other criteria for changing parameters are taken into consideration to determine whether improvement is necessary like (no of epochs, learning rate, batch size). The optimum performance is observed, with epochs =30, learning rate =0.01, batch size =128. The source and target test cases words matching of RNN is equal as compare to LSTM and GRU.

One of the reasons for India's low level of digital literacy is language barriers. This work can be used to create web content translation applications, which will lessen the language barrier. It would encourage more people to use digital platforms, government portals, and educational apps, which would aid the government's Digital India initiative. Techniques like attention mechanisms, large language models, regularization techniques, and transformer-based models like bidirectional encoder representations from transformers (BERT), generative pre-trained transformer (GPT), and multilingual bidirectional and auto-regressive transformers (MBART) may also include as part of the future scope of this work. Besides this, processes to improve quality checking, as well as using data generation for parallel corpora strategies, can be built to further uplift translation efficiency.

Table 5. Test cases results of HOBMT systems

Type	HI-OD			OD-HI		
	RNN	LSTM	GRU	RNN	LSTM	GRU
Actual	वह मैं हूं (*vah maian hūan*)	वह मैं हूं (*vah maian hūan*)	वह मैं हूं (*vah maian hūan*)	ମୁଁ ଯିବାକୁ ଚିନ୍ତା କରୁଛି (*mun jibaku cinta karuchi*)	ମୁଁ ଯିବାକୁ ଚିନ୍ତା କରୁଛି (*mun jibaku cinta karuchi*)	ମୁଁ ଯିବାକୁ ଚିନ୍ତା କରୁଛି (*mun jibaku cinta karuchi*)
Predicted	वह मैं हूं (*vah maian hūan*)	वह हो है (*vah ho hai*)	वह हूं (*vah hūan*)	ମୁଁ ଯିବାକୁ ଚିନ୍ତା କରୁଛି (*mun jibaku cinta karuchi*)	ମୁଁ ଯିବାକୁ (*mu jibaku*)	ମୁଁ ଯିବାକୁ କରିବି (*mun jibaku karibi*)

4. CONCLUSION

Standard data for Indian languages is insufficient and not developed to implement on several DL methods effectively. Less work has been accounted for Hindi to Odia or Odia Hindi MT system, which is a challenge to create one HOBMT system. It emphasizes on bidirectional evaluation on low-resource Indic languages. These discussed seq2seq models performs decent at the same time, it confuses to grasp the important words in several cases. To determine the best model performs best for MT, this work compares performances of various models using different evaluation metrics. This analysis allows us to validate and compare not only the accuracy but also the final performance of the models with respect to their dataset, training time, and parameters. In contrast to LSTM and GRU networks, the comparison demonstrates that RNN outperforms them on shorter sentences with less computation time.

FUNDING INFORMATION

The authors state no funding is involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Subhashree Satpathy			✓		✓	✓	✓	✓	✓	✓	✓			
Smitaprava Mishra	✓	✓		✓						✓	✓	✓		
Ajit Kumar Nayak	✓	✓		✓	✓	✓		✓		✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The author states no conflict of interest.

DATA AVAILABILITY

Data collected from the open source at https://www.manythings.org/anki/, reference number [23]. After that manually converted into Hindi-Odia bilingual form.

REFERENCES

[1] S. B. Das, D. Panda, T. K. Mishra, and B. K. Patra, "Statistical machine translation for Indic languages," *Natural Language Processing*, vol. 31, no. 2, pp. 328–345, Mar. 2025, doi: 10.1017/nlp.2024.26.

[2] Sitender, S. Bawa, M. Kumar, and Sangeeta, "A comprehensive survey on machine translation for English, Hindi and Sanskrit languages," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 4, pp. 3441–3474, Apr. 2023, doi: 10.1007/s12652-021-03479-0.

[3] B. S. Harish and R. K. Rangan, "A comprehensive survey on Indian regional language processing," *SN Applied Sciences*, vol. 2, no. 7, Jul. 2020, doi: 10.1007/s42452-020-2983-x.

[4] A. Kalyani and P. S. Sajja, "A review of machine translation systems in India and different translation evaluation methodologies," *International Journal of Computer Applications*, vol. 121, no. 23, pp. 16–23, Jul. 2015, doi: 10.5120/21840-4917.

[5] O. Bojar *et al.*, "Findings of the 2015 workshop on statistical machine translation," in *Proceedings of the Tenth Workshop on Statistical Machine Translation, Stroudsburg, USA*, 2015, pp. 1–46, doi: 10.18653/v1/W15-3001.

[6] N. Telrandhe and R. Kadu, "Machine translation models for low-resource Indo-Aryan languages: a comprehensive review of techniques, challenges and future scope," in *2025 3rd DMIHER International Conference on Artificial Intelligence in Healthcare, Education and Industry (IDICAIHEI)*, Nov. 2025, pp. 1–6, doi: 10.1109/IDICAIHEI65991.2025.11378214.

[7] J. Rautaray and P. Mishra, "A naive approach: translation of natural language to structured query language," in *ICST Transactions on Scalable Information Systems*, 2023, doi: 10.4108/eetsis.4344.

[8] U. M. Mohapatra, "Machine learning based translation from ODIA to HINDI," *Journal of Advanced Multidisciplinary Research Studies and Development*, vol. 4, no. 2, pp. 118–125, 20225.

[9] P. M. Subhash, C. R. Kavitha, D. Gupta, and V. Kanjirangat, "Indo-Aryan dialect identification using deep learning ensemble model," *Procedia Computer Science*, vol. 235, no. C, pp. 2886–2896, 2024, doi: 10.1016/j.procs.2024.04.273.

[10] A. Naik, M. Karani, K. Chheda, and M. Gada, "NMT for English-Marathi using RNNs & attention," in *2022 6th International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, Aug. 2022, pp. 1–6, doi: 10.1109/ICCUBEA54992.2022.10010782.

[11] S. Yang, X. Yu, and Y. Zhou, "LSTM and GRU neural network performance comparison study: taking yelp review dataset as an example," in *2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI)*, Jun. 2020, pp. 98–101. doi: 10.1109/IWECAI50956.2020.00027.

[12] D. Datta, P. E. David, D. Mittal, and A. Jain, "Neural machine translation using recurrent neural network," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 4, pp. 1395–1400, 2020, doi: 10.35940/ijeat.D7637.049420.

[13] G. Ramesh *et al.*, "Samanantar: the largest publicly available parallel corpora collection for 11 Indic languages," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 145–162, 2022, doi: 10.1162/tacl_a_00452.

[14] N. S. Dash and B. B. Chaudhuri, "Why do we need to develop corpora in Indian languages," *International Working Confereence on Sharing Capability in Localisation and Human Language Technologies (SCALLA-2001)*, 2001, pp. 1-18.

[15] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *40th Annual Meeting on Association for Computational Linguistics (ACL)*, 2001, pp. 311–318, doi: 10.3115/1073083.1073135.

[16] S. Banerjee and A. Lavie, "METEOR: an automatic metric for MT evaluation with improved correlation with human judgments," in *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005, pp. 65–72, doi: 10.5555/1626355.1626389.

[17] C. Park, M. Chen, and T. Hain, "Automatic speech recognition system-independent word error rate estimation," in *2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024, pp. 1979–1987.

[18] D. K. Thennal, J. James, D. P. Gopinath, and M. K. Ashraf, "Advocating character error rate for multilingual ASR evaluation," in *Annual Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics*, 2025, pp. 4926–4935, doi: 10.18653/v1/2025.findings-naacl.277.

[19] R. Raja and A. Vats, "Parallel corpora for machine translation in low-resource Indic languages: a comprehensive review," in *Proceedings of the Eighth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2025)*, Stroudsburg, USA, 2025, pp. 129–143, doi: 10.18653/v1/2025.loresmt-1.12.
[20] S. B. Das, A. Biradar, T. K. Mishra, and B. K. Patra, "Improving multilingual neural machine translation system for Indic languages," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 6, pp. 1–24, Jun. 2023, doi: 10.1145/3587932.
[21] B. Haddow, R. Bawden, A. V. M. Barone, J. Helcl, and A. Birch, "Survey of low-resource machine translation," *Computational Linguistics*, vol. 48, no. 3, pp. 673–732, 2022, doi: 10.1162/coli_a_00446.
[22] A. Godase and S. Govilkar, "Machine translation development for Indian languages and its approaches," *International Journal on Natural Language Computing*, vol. 4, no. 2, pp. 55–74, 2015, doi: 10.5121/ijnlc.2015.4205.
[23] Tatoeba Project, "Tab-delimited bilingual sentence pairs," ManyThings.org. Accessed: Jan. 20, 2025. [Online]. Available: https://www.manythings.org/anki/
[24] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2014, pp. 1724–1734, doi: 10.3115/v1/d14-1179.
[25] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, Mar. 1994, doi: 10.1109/72.279181.
[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[27] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling," in *Interspeech 2012*, Sep. 2012, pp. 194–197. doi: 10.21437/Interspeech.2012-65.
[28] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*.
[29] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *28th International Conference on Neural Information Processing Systems*, 2014, pp. 3104–3112, doi: 10.5555/2969033.2969173.
[30] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," in *International Conference on Learning Representations, Banff, Canada*, 2014.
[31] K. Revanuru, K. Turlapaty, and S. Rao, "Neural machine translation of Indian languages," in *10th Annual ACM India Compute Conference*, New York, United States, Nov. 2017, pp. 11–20, doi: 10.1145/3140107.3140111.

BIOGRAPHIES OF AUTHORS

Subhashree Satpathy received the degree in MCA from CMR College of Engineering and Technology, Hyderabad (CMRCET/CMRK), India in 2016. M.Tech. in Computer Science from Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India in 2018. She is currently pursuing his Ph.D. in Computer Science and Engineering at Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India. She has published 5 papers in various international conferences. She can be contacted at email: subhashreesatpathy16@gmail.com.

Prof. Dr. Smitaprava Mishra is currently working as professor in Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India. She has 18 years of teaching and research experience in the current organization. She has published 30 numbers of research papers in various reputed journals and conferences. 15 M.Tech. scholars produced under her supervision in the research areas of data mining, machine learning, and natural language processing. 4 Ph.D. scholars are pursuing research under her supervision. She has contributed several academic activities at organization level. She can be contacted at email: smitamishra@soa.ac.in.

Prof. Dr. Ajit Kumar Nayak is the professor and head of the Department of Computer Science and Information Technology, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India. He received degree Electrical in Engineering from the Institution of Engineers, India in the year 1994, M.Tech. and Ph.D. degrees in Computer Science from Utkal University in 2001 and 2010 respectively. He has published about 70 research papers in various journals and conferences. Also co-authored a book 'Computer network simulation using NS2'. 10 Ph.D. scholars have been awarded Ph.D. under his supervision. He can be contacted at email: ajitnayak@soa.ac.in.