

Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text

Taniya Seal ¹, , Debapratim Das Dawn ², Ankit Saha ¹, Subhasish Mukherjee ¹, Prajjal Mukherjee ¹, Abhinandan Khan ³, Sanjit Kumar Setua ¹, Rajat Kumar Pal ¹

1. Department of Computer Science and Engineering, University of Calcutta, Kolkata, IND

2. Department of Computer Science, Bundwan Polytechnic, Purulia, IND

3. Product Development and Diversification, ARP Engineering, Kolkata, IND

Received: June 30, 2025 | Review began: November 03, 2025 | Review ended: December 01, 2025 | Published: February 27, 2026

© Copyright 2026

This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

In multilingual machine translation systems, word sense disambiguation (WSD) plays a critical role in ensuring contextual accuracy, especially when translating between morphologically rich languages like Bengali and English. This paper focuses on the computation of machine translation errors arising from incorrect sense selection of polysemous Bengali words and proposes a hybrid approach to minimize such errors through implicit consideration of WSD. We explore and compare multiple word embedding techniques, including Bag of Words, Term Frequency-Inverse Document Frequency, Word2Vec, and FastText, ultimately identifying FastText as the most effective due to its subword-level handling of Bengali morphology. Furthermore, we incorporate a custom label of exact sense mappings to preserve the correct translation of named entities and culturally significant terms such as mythological names like “कर्ण” (Karṇa) being translated accurately as “Karna” rather than “a character of Mahabharata”. We implemented a hybrid approach consisting of FastText, bidirectional long short-term memory, and attention mechanisms. To evaluate the effectiveness of our approach, some machine translation evaluation metrics are measured. Our proposed methodology is compared with existing WSD algorithms. It improves the disambiguation issue by leveraging contextual embeddings and exact sense matching that leads to better translation outcomes. This work highlights the need for sense-aware translation models and presents a robust, hybrid strategy for mitigating WSD-related translation errors in Bengali-to-English machine translation systems.

Categories: AI applications, Machine Learning (ML), Natural Language Processing (NLP)

Keywords: bengali natural language processing, word sense disambiguation, deep learning, machine translation error, translation evaluation metrics

Introduction

Machine translation (MT) has progressed rapidly in recent years, mainly due to breakthroughs in neural networks and attention-based models such as transformers. Despite these technological improvements, capturing the word meaning accurately in a context remains a serious hurdle. Especially, it is difficult for under-resourced and linguistically complex languages like Bengali. Word sense disambiguation (WSD) is the process of identifying the correct meaning of a word that has multiple interpretations depending on context. In languages like Bengali, polysemous words are common, and their meanings vary widely based on syntactic and semantic clues. For instance, the word “চাল” (Cāla) may mean either “rice” or “to drive,” depending on its surrounding words. If a translation system selects the wrong interpretation, it can produce a sentence that is grammatically correct but contextually misleading. This is a recurring issue in mainstream MT

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

sentence structure. Therefore, there is a critical need for a context-aware translation approach that can effectively resolve WSD errors and enhance the semantic fidelity of Bengali-to-English MT. Classification of WSD errors in MT can be broadly categorized into three types mentioned in Figure 1.



FIGURE 1: Types of WSD errors in machine translation

WSD, Word Sense Disambiguation

I. Idiomatic expressions error occurs when idioms are translated word-for-word instead of capturing their intended meaning. For example:

Translation 1: "গাছে কাঁঠাল, গৌঁফে তেল" (Gāchē kām̃ṭhāla, gōm̃phē tēla)

Google Translator: "Jackfruit on the tree, oil on the mustache"

Correct Translation: "Expecting results without effort"

Reason: The system fails to recognize idioms and translates them literally.

II. Lexical ambiguity error occurs when a word has multiple meanings, and the translation system selects the incorrect one due to a lack of context.

Translation 2: "আমার কল খারাপ হয়ে গেছে।" (Āmāra kala khārāpa hayē gēchē.)

Google Translator: "My call went bad."

Correct Translation: "My tap is broken."

Reason: The word "কল" (kol) can mean both "call" and "tap," leading to confusion.

III. Contextual error occurs when words or phrases are translated without considering their actual intended meaning in a given situation.

Example Source: "আপনার কষ্ট হচ্ছে?" (Āpanāra kaṣṭa hacchē?).

Incorrect Translation: "Are you having difficulty?"

Correct Translation: "Are you in pain?"

Reason: The phrase "কষ্ট হচ্ছে" (Kaṣṭa hacchē) in Bengali often refers to physical pain or discomfort, but it was translated as "difficulty," which changes the meaning.

Translating Bengali into English presents several complex issues, despite the progress made through modern neural translation models. One of the biggest difficulties is tied to how Bengali words can change meaning based on their usage in a sentence. Many terms are highly context-dependent, so without correctly interpreting their intended sense, translation systems can generate outputs that sound correct grammatically but convey the wrong idea. A second major issue is the limited availability of aligned Bengali-to-English datasets. In contrast to languages that have extensive digital resources, Bengali suffers from a shortage of parallel sentence collections [1], which are crucial for training accurate

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

translation models. This gap in data limits the learning potential of modern translation tools. The structural difference between Bengali and English also creates complications. Bengali typically follows a sentence order where the verb comes last, unlike English, where the verb usually appears in the middle [2]. As a result, translation systems must reorganize sentence components while preserving meaning - something they often fail to do effectively. Bengali's rich morphological system adds another layer of difficulty. Words can take on many forms due to changes in tense, case, or grammatical role. These variations increase the complexity of identifying the base meaning of a word, which is essential for accurate translation. Cultural and idiomatic language in Bengali adds yet another challenge. Many common phrases or sayings in Bengali do not have straightforward equivalents in English. If translated word-for-word, their meanings are often lost or distorted. Additionally, proper nouns - such as those of people, places, or institutions - are frequently misunderstood by translation systems, which might treat them as ordinary vocabulary and translate them incorrectly. Lastly, Bengali relies heavily on context to communicate meaning. A word or phrase might make sense only when seen within the broader passage it appears in. Many translation systems do not account for this and treat words in isolation, which can result in output that lacks relevance or clarity in English.

WSD in the Bengali language is crucial for a wide range of natural language processing (NLP) tasks and practical applications. One significant area where WSD plays a vital role is in MT [3]. Translating text from Bengali to another language requires an accurate understanding of words that carry multiple meanings, as the correct translation often depends on how the word is used within its specific context. Part-of-speech tagging also benefits greatly from effective WSD [4]. In Bengali, a single word can serve different grammatical roles depending on the sentence. Identifying the precise role whether noun, verb, adjective, etc., requires a clear understanding of the surrounding context, which WSD helps provide. WSD is also useful in detecting subjectivity in text. This involves determining whether a piece of writing reflects opinion or objective information, which is a common task in fields like sentiment analysis and content classification. Accurate sense identification makes it easier to categorize text appropriately. Another complex application of WSD is in emotion detection, particularly in informal digital content such as blogs or social media updates. Since many emotional expressions in Bengali rely on words with multiple interpretations, disambiguation is essential to correctly identify underlying emotions like happiness, anger, or fear. In addition, WSD contributes to opinion polarity analysis. This is particularly important for understanding online reviews or user feedback, where it is necessary to determine whether the expressed sentiment is positive, negative, or neutral. By clarifying the intended meaning of ambiguous terms, WSD helps systems make more precise judgments about user sentiment.

This paper addresses the problem by exploring how contextual information can be better used during translation from Bengali to English. We aim to reduce errors caused by incorrect word sense choices through a strategy that combines both computational techniques and human-in-the-loop insights. Our work tests multiple text representation approaches - such as Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), Word2Vec, and FastText - to determine which best supports disambiguation in Bengali sentences. To handle culturally significant words and named entities like "কর্ণা" (Karna) or "বৃহস্পতি" (Bṛhaspati), we have expanded our dataset to include a column called "Exact Sense English Word", ensuring accurate and culturally appropriate translations without confusion or unnecessary simplification. By integrating context-aware models and semantic mapping into the translation process, we aim to produce outputs in English language that are not only fluent but also true to the source language's meaning. This research ultimately contributes to developing a more reliable, hybrid translation system that better handles ambiguity and preserves the integrity of the original text. This work is driven by the need to bridge this semantic gap by developing a context-aware, linguistically informed translation framework that can handle WSD implicitly. By leveraging FastText embeddings, bidirectional long short-term memory (BiLSTM) models, and multi-head attention mechanisms, our goal is to enhance the quality and accuracy of Bengali-to-English MT, ensuring that cultural, idiomatic, and contextual integrity is preserved. Translation error is calculated using different machine translational error metrics described later in this study. This research aims to contribute to the development of more intelligent, inclusive, and effective translation tools for low-resource languages.

While existing translation systems often rely on surface-level or statistical patterns, our work delves into the semantic structure of Bengali, offering several noteworthy contributions:

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

1. Development of a context-aware WSD pipeline: We designed a custom deep learning pipeline that integrates FastText embeddings, BiLSTM sequence modeling, and multi-head self-attention mechanisms to accurately predict the intended meaning of ambiguous Bengali words within a context. This approach goes beyond traditional translation models by learning both semantic and syntactic cues for improved disambiguation.
2. Construction and enhancement of a WSD dataset: The project enhances a publicly available Bengali WSD dataset by manually labelling ambiguous words with their exact English equivalents. This includes the creation of a specialized attribute named Replaced Word to handle named entities, mythological terms, and culturally rooted expressions, ensuring their integrity in translation.
3. Comparative study of embedding techniques: The research systematically compares various word vectorization techniques - BoW, TF-IDF, Word2Vec, and FastText. The results confirm FastText's superiority for morphologically rich languages like Bengali due to its ability to capture subword information.
4. Construction of hybrid model: This hybrid model consists of BiLSTM models, and multi-head attention mechanisms, to enhance the quality and accuracy of Bengali-to-English MT.
5. Introduction of a post-prediction replacement mechanism: We implemented a custom function that dynamically replaces polysemous words in the original Bengali sentence with the model's predicted sense, thereby generating an improved, disambiguated sentence ready for translation.
6. Hybrid translation evaluation strategy: By integrating a pre-translation sense substitution phase with Google Translate, we demonstrate how modifying input text based on WSD significantly improves translation quality. A side-by-side comparison of outputs, with and without disambiguation, validates the impact of our approach.
7. Evaluation using state-of-the-art translation metrics: To measure the machine translational error, a contextual and supervised evaluation metric like BERT-MTE, as well as lexical and embedding-based metrics like bilingual evaluation understudy (BLEU), metric for evaluation of translation with explicit ordering (METEOR), and BERT scores, is used. Our proposed model consistently showed improved semantic alignment with human translations.
8. Performance benchmarking against Google Translate: Extensive experimentation revealed that our proposed system outperforms Google Translate in both controlled single-word tests and larger evaluation sets. Specifically, it achieved a 1.24% improvement in translation quality on selected words and a 0.66% overall gain on a broader test set of 100 ambiguous Bengali terms.

Materials And Methods

Literature review

Over the years, various strategies have been proposed to tackle WSD. These include rule-based systems, statistical models, and, more recently, approaches leveraging neural networks. However, leading translation platforms like Google and Bing translator continue to struggle with WSD in Bengali, especially in cases involving idioms, multiple-meaning words, and context-dependent phrases. To better understand and categorize these WSD-related issues, researchers have examined them in terms of idiomatic misinterpretation, lexical ambiguity, and contextual misunderstanding. These categories help pinpoint where current models fall short and guide the development of more robust solutions. These review works are broadly classified in two groups that are explained below.

Review of WSD Algorithms for Bengali

Several attempts have been made to address WSD in Bengali texts through dedicated algorithms. However, the outcomes have varied widely in terms of accuracy and reliability.

In Bengali, the WSD task is accelerated by a combined strategy that includes all required adjustments [5]. The results have been assessed for every combination produced by knowledge-based, supervised, and unsupervised methods, including knowledge-based with unsupervised, supervised with unsupervised, and supervised with knowledge-based. Using Lesk's technique with context expansion features, knowledge-based approaches and supervised methodologies are combined in lemmatized scenarios. The greatest result generated by these algorithms was the 92% accuracy.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

Finding emotions in both raw and romanized Bengali texts was the subject of another study [6]. The military, medical, religious, and general context datasets are gathered and tested using BERT, in order to perform sentiment analysis across many classes on Bengali social media comments that are categorized as acceptable, political, religious, and sexual.

In recent years, WSD has seen major progress through neural and hybrid models. In [7], a weakly supervised approach that integrates WSD into neural MT (NMT) systems is introduced using adaptive clustering and sense embeddings. Their model showed up to 1 BLEU point and 4% WSD improvement across multiple language pairs. This technique highlights the advantage of soft sense integration in translation tasks, especially in ambiguous scenarios.

Following that, SensPick, a gloss-enhanced BiLSTM architecture, is introduced that boosts WSD accuracy through context-aware gloss expansion, which has shown a 3.5% F1-score improvement over baseline models on standard WSD benchmarks [8].

More recently, WSD is tackled in translation using document-level context, constructing pseudo-documents to enhance semantic scope during translation. Their approach achieved disambiguation gains without requiring manual annotations [9].

A model named Word Sense Pretraining for Neural Machine Translation incorporates word sense pretraining into multilingual translation tasks. Using sense-aware pretraining derived from structured semantic databases, the model outperformed standard NMT baselines on the DiBiMT benchmark and further confirmed the role of WSD in cross-lingual accuracy [10].

These newer models mark a shift from traditional lexical and gloss-based disambiguation toward context-rich, embedding-driven WSD systems, especially in NMT pipelines. Their relevance becomes particularly important for Bengali - a morphologically rich and low-resource language where word-level ambiguity frequently affects translation quality.

Review of Evaluation Metrics

Evaluating the performance of MT systems remains a challenging and evolving area in NLP. Several automatic metrics have been proposed to approximate human judgment by comparing machine-generated translations with reference translations. This subsection reviews key evaluation metrics and studies that assess their effectiveness and correlation with human judgments.

In MT evaluation, BLEU score has become the de facto standard. BLEU measures the overlap of n-grams between the candidate and reference translations. While efficient and language-independent, BLEU is limited in its ability to capture synonyms, semantic variation, and sentence-level fluency. It is particularly weak for evaluating translations involving morphologically rich or low-resource languages like Bengali, where surface-form mismatches may still preserve semantic equivalence [11].

To address some of BLEU's limitations, METEOR was proposed [12]. METEOR incorporates stemming, synonym matching (via WordNet), and a penalty for word order variation, leading to better correlation with human judgment compared to BLEU. This work is further extended with m-BLEU, m-METEOR, and m-TER [13], which are more robust on multilingual datasets. These metrics were shown to outperform BLEU in both sentence-level and system-level evaluations, particularly for tasks with rich lexical variation. Their study remains influential in multilingual MT benchmarking.

Another recent trend involves embedding-based evaluation metrics, which assess translation quality in the semantic space rather than relying on exact token matches. BERTScore, based on contextual embeddings from BERT, has shown a strong correlation with human judgments across diverse languages and domains. A comprehensive survey on evaluation metrics has been done, and it has been concluded that embedding-based metrics such as BERTScore, MoverScore, and BLEURT significantly outperform traditional metrics like BLEU and translation edit rate (TER) in semantic evaluation tasks [14]. However, these methods are computationally more expensive and require pre-trained transformer models, which may not be feasible in all production settings.

Classification models with feature extraction

This section provides a detailed description of different classification models and feature extraction techniques used in this work. Feature extraction is the process of turning unprocessed input into numerical features that machine learning algorithms can use efficiently. An algorithm that uses the extracted characteristics to assign labels to input data is called a

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

classification model.

Analysis of Feature Representation Techniques for WSD

To build an efficient MT model with accurate WSD, various word embedding techniques are experimented to convert Bengali textual data into a numeric form that machine learning models can process.

BoW is a simple vectorization method where each word is represented as a one-hot encoded vector. Although this approach is computationally inexpensive, it fails to capture the contextual meaning or semantics of words. The advantage of BoW is that it is simple and fast, but it ignores context and word order.

TF-IDF improves on BoW by assigning importance to words based on their frequency and rarity across the dataset. While more expressive, it still lacks semantic understanding. The advantages of TF-IDF include its ability to highlight important terms, but it has limitations, such as its inability to capture synonymy or semantic relationships.

Word2Vec is a neural embedding model that maps words into dense vectors by considering their surrounding context. It captures semantic similarity and is more effective in detecting the correct sense of polysemous words. It can capture semantic meaning and context, but there are some limitations: out-of-vocabulary (OOV) words may be ignored.

FastText, an extension of Word2Vec, breaks words into subword units, allowing the model to handle OOV words more efficiently. In our experiments, FastText outperformed other techniques in Bengali text due to its ability to capture morphological variations. FastText can handle OOV words and affixes better, but it has slightly more computational cost.

Machine Learning-Based Classifier

Based on Bayes' theorem, Naïve Bayes (NB) is a probabilistic machine learning algorithm that is mostly employed for classification tasks. It functions under the premise where the features are conditionally independent having class label. It streamlines computation while maintaining high practical effectiveness. In contrast to intricate deep learning models, NB is lightweight, interpretable, and useful for sentiment analysis, spam filtering, and text classification. The algorithm calculates the probabilities of individual features and combines them to estimate the likelihood that a data point belongs to a specific class. NB performs well even with small training samples and is especially robust for high-dimensional data, despite its oversimplified assumptions. It is a dependable option for real-world applications due to its effectiveness and capacity to manage noisy data, particularly in natural language. Bayes' theorem is expressed by equation given below.

$$P(y/x_1, x_2, \dots, x_n) = \frac{P(y) \cdot \prod_{i=1}^n P(x_i/y)}{P(x_1, x_2, \dots, x_n)}$$

For classification, the denominator can be ignored as it is constant across classes, and the model simplifies to:

$$\hat{y} = \arg \max_y P(y) \cdot \prod_{i=1}^n P(x_i|y)$$

where, y is a class label (e.g., a particular word sense), x_1, x_2, \dots, x_n are the features (e.g., context words), $P(y)$ is the prior probability of class y , and $P(x_i|y)$ is the likelihood of feature x_i given in class y .

Random forest (RF) is an ensemble learning algorithm that builds multiple decision trees and combines their outputs to improve predictive accuracy and robustness. Unlike a single decision tree, which may be overfit to training data, RF reduces variance by averaging the predictions of multiple trees, making it more resilient to noise and outliers. Each tree is trained on a randomly sampled subset of the data using a technique called bagging, while feature selection at each split further enhances diversity among trees. This approach ensures strong performance across various domains, including classification, regression, and anomaly detection. RF is particularly effective when dealing with high-dimensional datasets and missing values, as it naturally handles feature interactions and does not require extensive preprocessing. Its ability to balance interpretability with predictive power makes it a popular choice in fields such as finance, healthcare, and recommendation systems. The classification decision is made by majority voting across all trees mathematically expressed below:

$$\hat{y} = \text{mode} (T_1(x), T_2(x), \dots, T_k(x))$$

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

where, $T_k(x)$ is the class prediction by the k-th decision tree.

Conventional classifiers such as NB, support vector machines, and decision trees often struggle with contextual dependencies and sparse high-dimensional data. Our motivation for proposing K^* arises from needing a more context-sensitive, interpretable, and data-efficient model that can adapt to domain-specific challenges. Let x be an input instance and $y \in Y$ be the target label. The K^* classifier computes a weighted probability distribution:

$$P(y/x) = \sum_{i=1}^k w_i \cdot P(y_i/x_i)$$

where, x_i represents the i-th neighbour of x , y_i its label, and w_i is the weight assigned to that neighbour based on a contextual similarity function.

Deep Learning-Based Classifier

To address the limitations of classical machine learning models in capturing long-term dependencies and complex linguistic structures, a gated recurrent unit (GRU)-based deep learning classifier is employed. GRUs are a type of recurrent neural network (RNN) designed to efficiently model sequential data, making them well suited for NLP tasks such as WSD in Bengali. A GRU unit uses two primary gates. Update gate z_t controls how much of the past information needs to be passed along to the future. Reset gate r_t decides how much of the previous information to forget. The hidden state h_t at time step t is updated as:

$$\begin{aligned} z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\ r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W \cdot [r_t \cdot h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \end{aligned}$$

where, x_t is the input at time step t ; σ is the sigmoid activation; and W_z , W_r , and W are trainable weight matrices.

RNNs are a class of neural networks designed for sequential data processing, where the order of inputs plays a crucial role. Unlike traditional feedforward networks, RNNs have internal memory that enables them to model dependencies between elements in a sequence. The RNN classifier designed for this research contains the following layers. At each time step t , an RNN receives an input, updates its hidden state h_t and produces an output y_t . The hidden state is a function of the current input and the previous hidden state:

$$\begin{aligned} h_t &= \tanh(W_h \cdot h_{t-1} + W_x \cdot x_t + b_h) \\ y_t &= W_y \cdot h_t + b_y \end{aligned}$$

where, h_t is the hidden state at time t ; x_t is the input at time t ; W_h , W_x and W_y are trainable weight matrices; and \tanh is the activation function.

Convolutional neural networks (CNNs), though originally developed for computer vision tasks, have demonstrated strong performance in various text classification problems. In this research, CNNs are explored as an alternative deep learning approach for Bengali WSD. The model exploits local word patterns and n-gram-like features to identify the correct sense of an ambiguous word based on its context.

Formally, for an input sentence of length n , represented as a matrix $X \in R^{n \times d}$ (where d is the embedding dimension), a filter W is applied to a window of h words to produce a feature:

$$c_i = f(W \cdot X(i : i + h - 1) + b)$$

where, f is a non-linear activation, $X(i : i + h - 1)$ is the sub-matrix of words from position i to $i + h - 1$, and c_i is a resulting feature at position i .

How to cite this article:

BiLSTM networks represent a significant advancement in sequential modeling, especially for language tasks like WSD. BiLSTMs extend the standard LSTM architecture by processing the input sequence in both forward and backward directions, thereby capturing both past and future context for each word. This bidirectional flow makes BiLSTMs especially powerful in morphologically rich and syntactically flexible languages like Bengali. A BiLSTM consists of two LSTM layers: i) A forward LSTM processes the sequence from left to right. ii) A backward LSTM processes it from right to left. At each time step t , the final hidden representation is formed by concatenating the forward and backward hidden states:

$$\begin{aligned}\vec{h}_t &= LSTM_{forward}(x_t) \\ \vec{h}_t &= LSTM_{backward}(x_t) \\ h_t &= [\vec{h}_t, \vec{h}_t]\end{aligned}$$

Transformer-based architectures have revolutionized NLP tasks by relying entirely on self-attention mechanisms, removing the sequential bottleneck of RNNs. In this research, a transformer-inspired model is integrated with a BiLSTM backbone and multi-head self-attention for the task of Bengali WSD.

The multi-head attention mechanism computes attention across different subspaces and concatenates the results:

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \dots, head_h)W^o$$

Each head computes:

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

where, Q , K , and V are query, key, and value matrices.

Analysis of MT Evaluation Metrics

The translation quality of neural network-based MT transcends that of translations derived using statistical methods. This growth in MT research has entailed the development of accurate automatic evaluation metrics that allow us to track the performance of MT. However, automatically evaluating inaccuracy and comparing MT systems is a challenging task. The evaluation metrics that are used in MT are classified into lexical-based metrics, unsupervised metrics, and supervised metrics depending on the machine learning technique. The classification tree in details is shown in Figure 2.

How to cite this article:

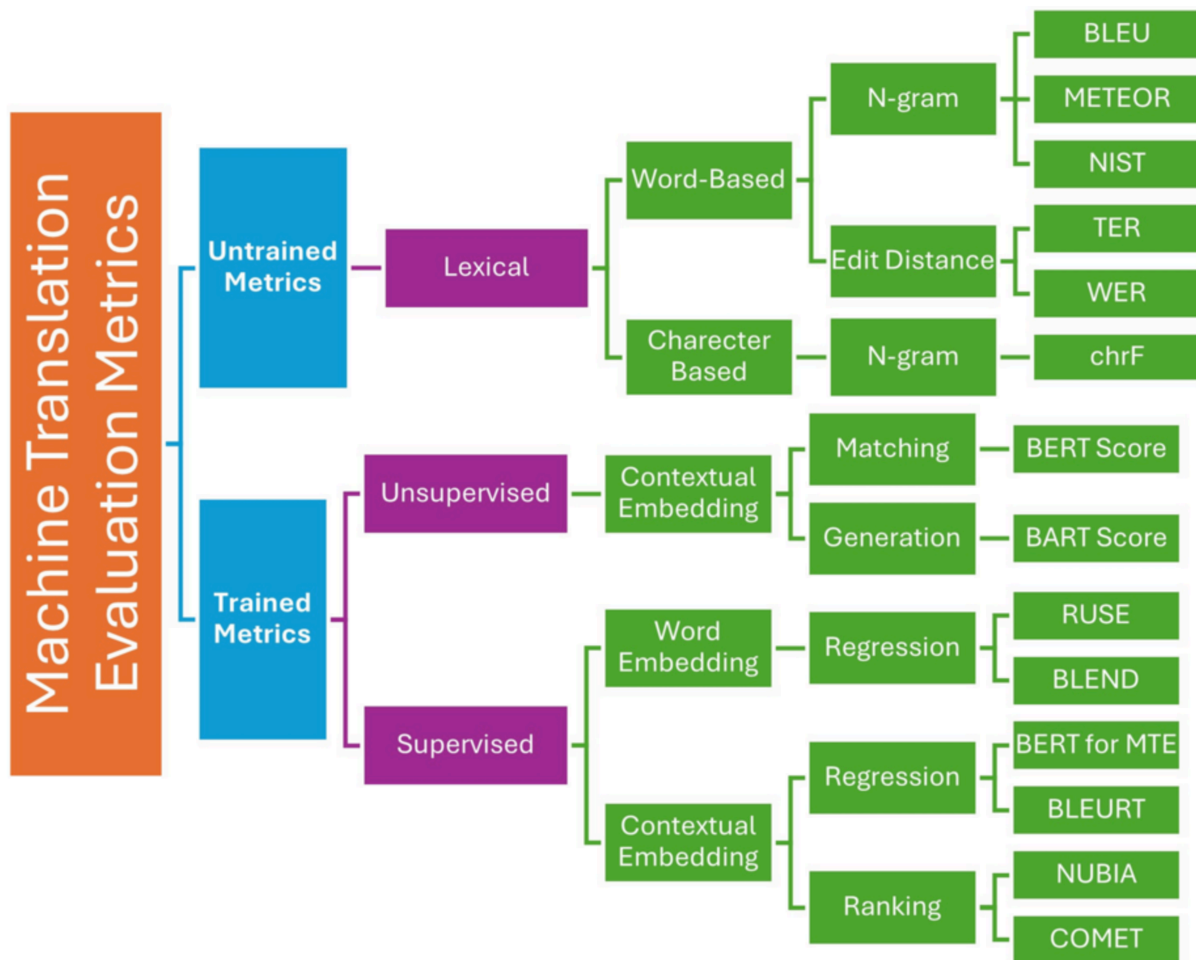


FIGURE 2: Machine translation evaluation metrics

BLEU with 3-gram matching (BLEU-3) is a method used to evaluate the quality of translations produced by machines. It works by checking how many sequences of one, two, and three words (known as unigrams, bigrams, and trigrams) in the machine-generated text also appear in a reference translation written by a human. The higher the overlap, the better the translation is. To make the evaluation more robust, BLEU-3 often uses smoothing techniques, which help prevent the score from dropping sharply when some 3-grams are missing. It also typically ignores differences in capitalization by converting all words to lowercase. The final score is expressed as a percentage, where 0% means no similarity to the reference and 100% means an exact match.

METEOR is an evaluation method, developed to better capture the quality of machine-translated text. Unlike BLEU, which relies mostly on exact word matches, METEOR considers more flexible factors such as word variations, similar meanings, and word order. It uses techniques like stemming to match related word forms and synonym matching to recognize different words with similar meanings. Additionally, METEOR evaluates both how many correct words are present (precision) and how many relevant words are covered (recall) and then aligns them to compare structure and flow. Scores are given on a scale from 0 to 1 (or 0% to 100%), where a higher score indicates that the translation is closer to human-like quality.

The National Institute of Standards and Technology (NIST) score is an MT evaluation metric like BLEU but with an important difference. It assigns more weight to rare words since they often carry more meaning in a sentence. Unlike BLEU, which focuses solely on word overlap, NIST prioritizes informativeness and context-sensitive evaluation, making it

How to cite this article:

more reliable for assessing translation quality. It incorporates weighted n-grams, giving higher importance to rare words and avoiding BLEU's tendency to penalize them. The NIST score typically ranges from 0 to 10, with higher values indicating better translation quality.

TER is a metric that measures the amount of editing required to make MT output match a reference translation. Unlike BLEU, which rewards similarity, TER focuses on how much effort is needed to correct errors, making it particularly useful for error analysis. It calculates the number of insertions, deletions, substitutions, and shifts required to transform the MT output into the reference translation, normalizing the score based on sentence length. TER values range from 0 to 100%, where lower scores indicate better translations.

Character F-score (ChrF) is an MT evaluation metric that assesses translation quality at the character level rather than the word level. This makes it particularly useful for morphologically rich languages like Bengali, where words can have complex inflections. Unlike BLEU, which relies on word-level matching, ChrF is more effective in handling spelling variations and small word changes. It calculates translation quality using character-based precision and recall, making it more robust for languages with flexible word structures. The ChrF score ranges from 0 to 100, with higher values indicating better translations.

Word error rate is an evaluation metric used to determine how accurate a translation is by comparing it word by word with a reference sentence. It calculates the number of edits, such as inserting, deleting, or replacing words, needed to transform the system output into the correct version. These edits are based on the Levenshtein Distance, which measures how different two sequences are. The total number of required changes is then divided by the number of words in the reference to normalize the score. Word error rate is expressed as a percentage from 0% to 100%, where a lower score means higher accuracy. It is commonly applied in both MT and speech recognition systems to assess performance based on exact word matching.

BERT score is an MT evaluation metric that measures the similarity between a machine-generated translation and a reference translation using contextual word embeddings from the BERT model. Unlike traditional n-gram-based metrics, BERT score evaluates translations at the semantic level by computing cosine similarity between token embeddings. This allows it to capture meaning beyond exact word matches, making it particularly useful for morphologically rich languages like Bengali. BERTScore_F1 is widely used in translation quality assessment due to its ability to handle paraphrasing and variations in word choice. It incorporates precision, recall, and F1-score to provide a balanced evaluation.

BART score is another transformer-based evaluation metric that leverages the BART model to assess translation quality based on fluency, adequacy, and contextual coherence. Unlike BERT score, which focuses on token similarity, BART score predicts how well a given translation aligns with a human reference by considering sentence-level structure and meaning. BART score has gained popularity in evaluating text generation tasks due to its strong correlation with human judgments. A higher scaled score generally indicates better fluency and meaning retention.

BERT-MTE score is an advanced MT evaluation metric that enhances BERT score by incorporating deeper contextual embeddings to assess translation quality. Unlike traditional n-gram-based metrics, it evaluates semantic similarity between machine-generated and human reference translations using transformer-based representations. This makes BERT-MTE score particularly effective for morphologically rich languages like Bengali, where exact word matches may not fully capture meaning variations. By focusing on contextual alignment, it handles paraphrasing, synonym usage, and structural differences in translations more effectively. The metric leverages BERT embeddings to provide a more accurate assessment of meaning similarity rather than just word overlap, making it robust against word reordering and minor textual variations. A higher BERT-MTE score indicates better alignment with human translations in terms of meaning and fluency, whereas lower scores suggest semantic mismatches or structural inconsistencies. Overall, BERT-MTE score offers a more refined evaluation compared to traditional metrics like BLEU by prioritizing contextual accuracy over lexical similarity, making it highly valuable for evaluating translations in complex languages like Bengali. Mathematical formula of BERT-MTE score is given below.

$$BERT - MTE = \cos(\vec{e}_{MT}, \vec{e}_{Ref}) = \frac{\vec{e}_{MT} \cdot \vec{e}_{Ref}}{\|\vec{e}_{MT}\| \cdot \|\vec{e}_{Ref}\|}$$

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

where, \vec{e}_{MT} = Embedding vector of the machine-translated sentence and \vec{e}_{Ref} = Embedding vector of the reference, i.e., human-translated sentence

NUBIA score is an MT evaluation metric that measures the semantic similarity between a machine-generated translation and a reference translation using deep learning-based natural language understanding. Unlike traditional metrics such as BLEU, which rely on n-gram overlaps, NUBIA score focuses on assessing meaning by leveraging transformer-based models. This allows it to capture nuances such as paraphrasing, contextual shifts, and logical consistency in translations. NUBIA score is particularly useful for evaluating translations in morphologically rich languages like Bengali, where word inflections and variations can lead to differences in expression while retaining the same meaning. By using pre-trained language models, NUBIA score effectively determines how well a translated sentence aligns with the intended meaning of the reference. A higher score indicates better semantic preservation, while a lower score suggests a lack of contextual accuracy or fluency.

COMAT score is an MT evaluation metric designed to assess the coherence, meaning preservation, and adequacy of machine-generated translations. Unlike traditional metrics such as BLEU, which rely on n-gram overlaps, COMAT score leverages advanced neural network models to capture contextual meaning and fluency. This makes it particularly effective for evaluating translations in morphologically rich languages like Bengali, where word variations and sentence structures can differ significantly from the reference translation while still conveying the same meaning. COMAT score focuses on semantic alignment rather than just surface-level word matches, making it more robust against paraphrasing, synonym usage, and word order variations. A higher COMAT score indicates that the translation closely follows the intended meaning of the reference, while a lower score suggests discrepancies in fluency, coherence, or adequacy.

BLEURT score is an MT evaluation metric that leverages deep learning models to assess the quality of translations based on semantic similarity rather than mere word overlap. Unlike traditional n-gram-based metrics such as BLEU, the BLEURT score is trained on large datasets of human evaluations, making it more aligned with human judgment. It utilizes BERT-based embeddings to compare machine-generated translations with reference texts, capturing nuances such as paraphrasing, fluency, and contextual accuracy. This metric is particularly useful for evaluating translations in morphologically rich languages like Bengali, where word inflections and variations can differ while retaining the same meaning. BLEURT score effectively measures how well a translated sentence preserves the intent and coherence of the original reference. A higher score signifies better alignment with human translation quality, whereas a lower score suggests inconsistencies in meaning, grammar, or fluency.

RUSES score is an MT evaluation metric that focuses on assessing the fluency, coherence, and semantic accuracy of machine-generated translations. Unlike traditional n-gram-based metrics such as BLEU, RUSES score leverages deep learning models and contextual embeddings to better align with human judgment. It evaluates translations based on how well they preserve the original meaning while maintaining natural language flow. This metric is particularly effective for morphologically rich languages like Bengali, where direct word-to-word comparisons often fail to capture nuances such as inflections, word order variations, and paraphrasing. By analysing semantic similarity instead of relying solely on lexical overlap, the RUSES score provides a more accurate representation of translation quality. Higher scores indicate better alignment with human translations, while lower scores suggest issues with fluency, coherence, or context.

BLEND score is an MT evaluation metric that combines multiple assessment techniques to provide a comprehensive measure of translation quality. Unlike traditional metrics such as BLEU, which rely on surface-level word overlap, BLEND score integrates aspects of semantic similarity, fluency, adequacy, and coherence to better align with human judgments. By leveraging deep learning models and contextual embeddings, it captures nuances in translation quality that word-based metrics may overlook. This metric is particularly useful for morphologically rich languages like Bengali, where variations in word order, inflections, and paraphrasing can significantly impact translation assessment. BLEND score ensures that translations are evaluated not just based on lexical similarity but also on how well they preserve the meaning and readability of the original text. A higher score indicates a well-formed, accurate translation, while a lower score suggests deficiencies in fluency, coherence, or semantic preservation.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

Word	Bengali Sentence	Sense	Replaced Word
জল	খালি পেটে জল, ভরা পেটে ফল।	এক প্রকার পানীয় বিশেষ	—
জল	এই পদক্ষেপ পিছুটান রেখেছে রক্ত জল করা পরিশ্রমের কাহিনী।	অত্যধিক পরিশ্রম	toil
জল	সারারাত খুম খুম বৃষ্টি	বৃষ্টি	—
জল	কফোটাও চোখে জল ফেলেছে।	অশ্রু	tears
আগুন	আগুন পুড়ে যায় শুদামের সমস্ত মালপত্র।	অগ্নি	—
আগুন	আগুন সেইক কোথায় আমার খুঁজছ?	উহে	—
বর্ণ	উপস্থাপন, ব্যবহার করার সময় পরিশীলনের আগেই প্রচারিত হবে।	অগ্রভাগ	tip
বর্ণ	বর্ণ বৈষম্য শ্রেণি বিভাজনের সৃষ্টি।	হিন্দুধর্মের বিভাগ	caste
বর্ণ	ছাপানো হচ্ছে বিশ্বয় একটি বাংলা বর্ণ; এটি কোনো চিঠি নয়।	অক্ষর	letter
বর্ণ	ওয়া ময়ম কে দেখি দুজনের বর্ণ আলাদা।	রঙ	—
চাল	ওই চালের পেছনেই খরচ হয় ১০০ টাকা	খাদ্যদ্রব্য বিশেষ	—
চাল	অনেক চাল গেছে ছাদের দেখা গেছে।	ঘরের ছাউনির	roof
চাল	দারাব চাল দেবার জন্যে	দারাব দান	move

TABLE 1: Snapshot of modified dataset

Feature Representation

To build an efficient MT model that can handle WSD properly, multiple word embeddings are experimented to convert Bengali text into a numerical format. It can be further processed by deep learning algorithms. After comparative analysis, FastText is selected as the optimal method for the proposed system, due to its superior handling of morphologically rich languages like Bengali. Unlike traditional word embedding models that treat words as atomic units, e.g., Word2Vec, FastText represents words as the sum of character n-gram embeddings. This allows the library to understand subword-level patterns, handle OOV words, and generalise better in Bengali languages with inflexions, compounds, and agglutinative structure. By integrating FastText embeddings into our proposed model, richer contextual information can be provided, which significantly improve sense classification and, ultimately, the quality of Bengali-to-English translations.

Description of the Proposed Model

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

Figure 4 illustrates the architecture of proposed deep learning model for sense prediction of a polysemous word used Bengali sentences. The model is designed to capture semantic context, focus on relevant tokens, and accurately classify ambiguous words into their correct senses. The components of the architecture are described below.

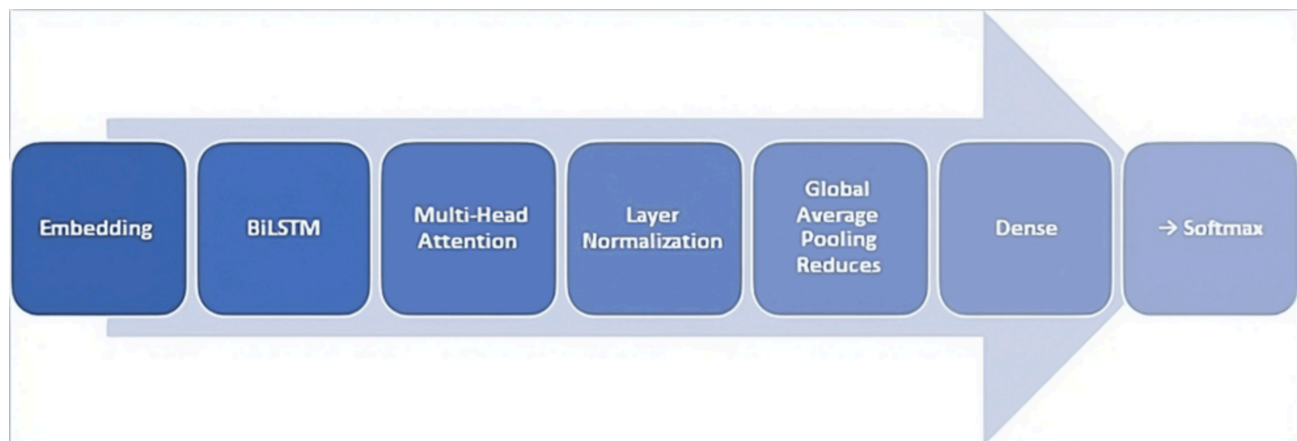


FIGURE 4: Architecture of the proposed hybrid model

BiLSTM, Bidirectional Long Short-Term Memory

In the embedding layer, pre-trained FastText Bengali word embedding is utilized, which is available in the FastText library developed by Facebook AI Research. At this stage, the Bengali input sentence is passed through a custom-defined function `PreprocessText`, which takes a Bengali sentence and removes non-Bengali characters and tokenizes it using `word_tokenize` function from natural language toolkit (NLTK) library. The output is a clean, space-separated Bengali string. Next, `TokenizerTextsToSequences` function is applied using Keras library. It converts the cleaned sentence into a list of integer tokens representing word indices. Then, `PadSequences` function from Keras pads the sequences to ensure uniform input length. These padded sequences are embedded using a custom FastText embedding function, resulting in an embedding matrix where each word token is replaced by its corresponding 300-dimensional FastText vector.

In the BiLSTM layer, the embedded sentence E is passed into BiLSTM function, a Keras layer that reads input in both directions. This outputs H , a sequence of hidden state vectors containing contextual information from both the left and right of each word. The output H is then passed through MultiHeadAttention function, implemented using Keras' MultiHeadAttention layer, which computes attention scores to determine how much focus should be paid to other tokens in the sequence. It produces A , a context-enhanced vector. The addition $A + H$ is passed through LayerNormalization function, a Keras layer, which standardizes inputs to stabilize training and is more suitable than batch normalization for sequence data. It outputs A_norm . GlobalAveragePooling1D function is then used to reduce the variable-length sequence to a fixed-size vector. This is a Keras built-in layer that computes the average of each feature across all time steps. Dropout function by Keras applies a dropout rate of 0.2-0.3, which randomly disables neurons during training to prevent overfitting. Then, Dense function provides a fully connected Keras layer that learns higher-level feature transformations. This is followed by Softmax function, which turns the output into a probability distribution over sense classes. Argmax function selects the class with the highest probability. Then, the integer class is mapped back to its original label with InverseTransform function from Scikit-learn library to get the sense string.

The original sentence S is scanned for a polysemous word w for first match. If matched, the system replaces w with a corresponding mapped term from a custom-built dictionary function created during dataset preparation. This produces $T_modified$, the modified Bengali sentence with more accurate word sense. Both the original and $T_modified$ sentences are translated using the `googletrans` library function, and then compared using custom-defined function `calculate_bert_similarity` function, which uses BERT from `huggingface transformers` library to compute semantic similarity between the translations. Table 2 shows the algorithm of WSD-aware Bengali to English Translation

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

	Input: Bengali sentence S	
	Output: Translated English sentence T_modified, the predicted sense sense_label, and the translation similarity Score	
1	Apply PreprocessText(S) to clean and tokenize the sentence. Store the result in S_pre.	
2	Apply TokenizerTextsToSequences(S_pre) to convert the cleaned sentence into a list of integer tokens. Store the result in X.	
3	Apply PadSequences(X) to pad the input to a fixed length. Store as X_padded.	
4	Generate the embedding matrix from X_padded using FastText and store as E.	
5	Pass the embedded sequence E through a BiLSTM layer. Store the output as H.	
6	Apply multi-head Attention on H, store the result as A.	
7	Apply layer normalization to the sum of A and H, and store the result as A_norm.	
8	Apply global average pooling to A_norm, and store the pooled vector as V.	
9	Pass V through a dense layer with Dropout, followed by a Softmax activation. Store the output as y.	
10	Use argmax(y) to find the class with the highest probability. Store it as predicted_class.	
11	Decode predicted_class to its actual label using LabelEncoder.inverse_transform(). Store this as sense_label.	
12	Identify the first polysemous word w in the original sentence S.	
13	If w exists in the sentence:	
14		Replace w with the dictionary-mapped value of sense_label.
15		Store the updated sentence as T_modified.
16	End if	
17	Translate the original sentence S and store the result as T_original.	
18	Translate the modified sentence T_modified.	
19	Use BERT-MTE similarity scoring to compare T_original and T_modified, and store the result as Score.	
20	Return the modified sentence T_modified, the predicted sense_label, and the translation similarity Score.	

TABLE 2: Algorithm of WSD-aware Bengali to English translation

BiLSTM, Bidirectional Long Short-Term Memory; BERT-MTE, Bidirectional Encoder Representations from Transformers – Machine Translation Evaluation; WSD, Word Sense Disambiguation

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

Dataset

A publicly available benchmark dataset named Bengali Word Sense Disambiguation (WSD) dataset [15] is used for this purpose. In this dataset, a collection of 100 polysemous Bengali words has been selected, and a text corpus is created with those words. The dataset contains 45 polysemous words with four senses and 55 polysemous terms with three senses. There are 165 and 180 sense definitions in the dataset, respectively. The snapshots of the dataset have been covered below.

Snapshot 1:

- Index: W_1S_1
- Sample: খালি পেটে জল, ভরা পেটে ফল।
- Transliteration: Khāli pētē jala, bharā pētē phala.
- Translation: Water on an empty stomach, fruits on a full one.
- Sense(S1): এক প্রকার পানীয় বিশেষ (A type of drink)
- Replaced Word: drinking water
- Word(W_1): জল

Snapshot 2:

- Index: W_1S_2
- Sample: এই পদকে পিঠে রয়ে গেছে রক্ত জল করা পরিশ্রমের কাহিনি।
- Transliteration: Ēi padakē piṭhē rayē gēchē rakta jala karā pariśramēra kāhini.
- Translation: This medal carries the tale of blood-sweating hard work.
- Sense(S2): অত্যাধিক পরিশ্রম (Exhaustion)
- Replaced Word: toil
- Word(W_1): জল

Snapshot 3:

- Index: W_1S_4
- Sample: কফোটাও চোখের জল ফেলেছে।
- Transliteration: Kaphōṭā'ō cōm̄khēra jala phēlēchē.
- Translation: Even a single drop of tear has fallen from the eye.
- Sense(S4): অক্ষ (Tear)
- Replaced Word: tears
- Word(W_1): জল

Snapshot 4:

- Index: W_2S_1
- Sample: আগ পড়ে যায় গুণদামের সময় মালপত্র।
- Transliteration: Āga paṛē yāy guṇadāmera samay mālapatra.
- Translation: Goods often fall during warehouse fires.
- Sense(S1):: আগুন (Fire)
- Replaced Word: অগ্নি
- Word(W_2): আগ

Snapshot 5:

- Index: W_2S_2

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

- Sample: আগ সেক কোথায় আমরা খুঁজছ?
- Transliteration: Āga sēka kōthāy āmārē khun̄jacha?
- Translation: Why are you looking for me so early?
- Sense(S₂): আগে (Before)
- Replaced Word: ওহে
- Word(W₂): আগ

Snapshot 6:

- Index: W₂S₃
- Sample: যাত্রার সময় পরিস্কারভাবে আগ প্রসারিত হবে।
- Transliteration: Yātrāra, byabahāra karāra samaȳ pariṣkārahābē āga prasārita habē.
- Translation: During travel, the tip will be extended clearly.
- Sense(S₃): অগ্রভাগ (Tip)
- Replaced Word: tip
- Word(W₂): আগ

Snapshot 7:

- Index: W₃S₁
- Sample: বর্ণ অনুযায়ী শ্রেণি বিভাগ সৃষ্টি।
- Transliteration: Barṇa anuyāyī śreṇi bibhāga sṛṣṭi.
- Translation: Division is created according to caste.
- Sense(S₁): হিন্দুসমাজের বিভাগ (Caste)
- Replaced Word: caste
- Word(W₃): বর্ণ

Snapshot 8:

- Index: W₃S₂
- Sample: চিন্তামণি হচ্ছে বিস্ময় একটি বাংলা বর্ণ; এটি কোনো চিহ্ন নয়।
- Transliteration: Cintāmaṇi hacchē bismaya ēkaṭi bāṅlā barṇa; ēṭi kōnō cihna naḷa.
- Translation: Chintamani is a surprising Bengali letter; it's not any mark.
- Sense(S₂): অক্ষর (Letter)
- Replaced Word: letter
- Word(W₃): বর্ণ

Snapshot 9:

- Index: W₃S₃
- Sample: সময় ফুরিয়ে যাওয়ার পর তারা রঙগুলি নিয়ে আলোচনা করল।
- Transliteration: Ōrā samaȳ phuriyē yā'ōyāra barṇa ālāpanā.
- Translation: They discussed the colors after time ran out.
- Sense(S₃): রং (Color)
- Replaced Word: রঙ
- Word(W₃): বর্ণ

As part of exploratory data analysis, it is examined how frequently each distinct sense appears in the dataset. Figure 5 shows the frequency of each sense represented on the x-axis and their respective counts on the y-axis.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

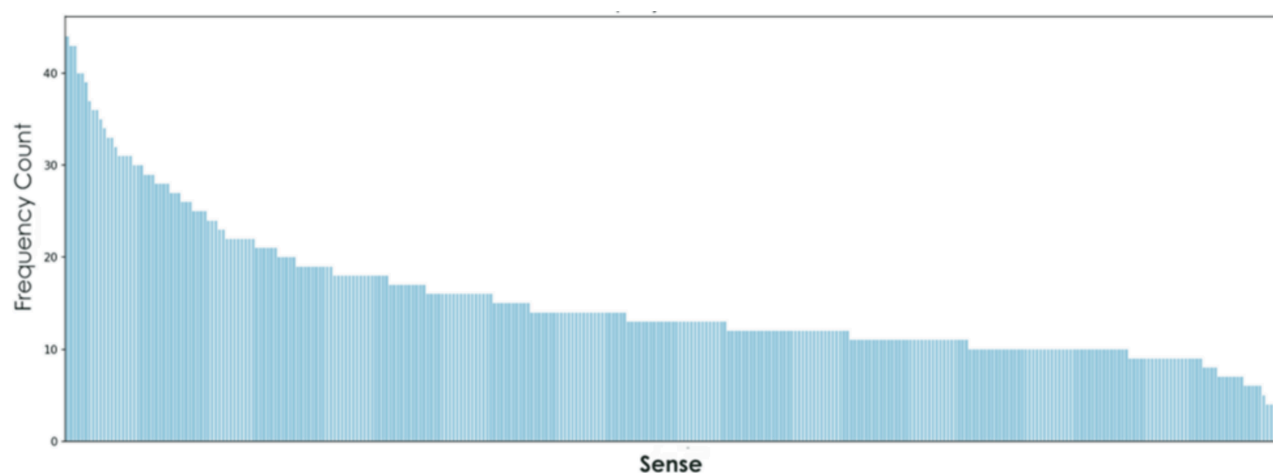


FIGURE 5: Frequency distribution of senses

It is shown that a few senses occur very frequently, i.e., more than 40 times, forming a long head. Most of the senses appear far less frequently, some as few as 10 to 18 times, forming a long tail and the lowest sense occurs 4 times only. This distribution indicates a highly imbalanced dataset. Such class imbalance is common in NLP tasks, especially in WSD, where many rare senses exist alongside a few dominant ones. This imbalance can severely affect the performance of supervised learning models, as they tend to overfit to the majority classes and ignore the minority ones. The most frequent sense appears 44 times, while many senses occur only once. The average frequency per sense is just over 14, but the standard deviation of value 7.31 shows wide variability with 348 unique senses. The top 10 senses already account for a significant portion of the dataset, which skews learning.

Results And Discussion

This section presents a comprehensive evaluation of the proposed WSD-enhanced Bengali-to-English translation model. Here, the effectiveness of both traditional machine learning classifiers and our deep learning architecture has been shown for disambiguating the senses of Bengali polysemous words. Multiple feature representation techniques, including BoW, TF-IDF, and FastText embeddings, are employed to transform Bengali textual data into a numerical form suitable for classification. Additionally, translation quality improvements have been made using semantic evaluation metrics such as BERT-MTE, enabling both quantitative and qualitative comparisons between original and modified translations. Under the machine learning approaches, three classifiers - NB, K*, and RF - are selected. This selection was guided by the results of a benchmark study [16] on Bengali WSD using the same dataset. Their findings indicated that these classifiers consistently performed better than other traditional models. By focusing on NB, K*, and RF, we ensured a representative comparison across probabilistic, instance-based, and ensemble learning paradigms. This focused evaluation allows us to directly contrast the performance of our proposed deep learning model against the strongest conventional baselines while maintaining relevance to established research.

Experimental setup

Every experiment was carried out with Google Colaboratory (Colab), which offers a cloud-based execution environment equipped with an Intel Xeon CPU, 12 GB system RAM, and an NVIDIA Tesla T4 GPU (16 GB VRAM). A learning rate of 0.0001 and a batch size of 32 were used during the 10 epochs of training. Python 3.1, pandas 2.0, and scikit-learn 1.3 were utilized in the environment to preprocess and evaluate the data. The models are evaluated over 100 ambiguous words, collectively representing 348 distinct senses. Each model's performance was assessed using the accuracy function, defined as the proportion of correctly predicted senses over the total number of test instances.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

The average scores in percentage achieved by the classifiers are shown in Figure 6. These results indicate that RF outperforms the other classifiers, achieving the highest average score. The K* classifier performs moderately, while NB yields the lowest performance among the three.

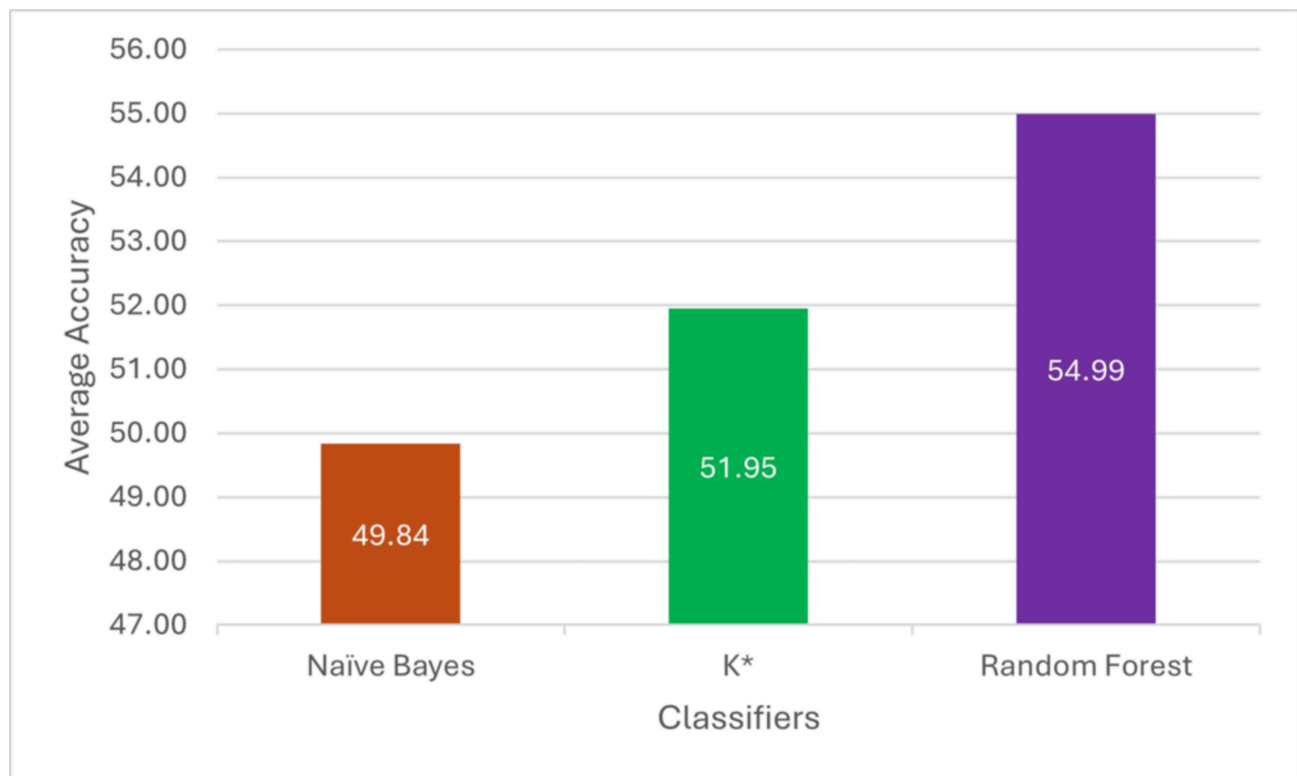


FIGURE 6: Comparison of average accuracy with classifiers

For converting text data into numerical features suitable for machine learning models, both BoW and TF-IDF are used as traditional methods. Figure 7 compares the performance of three machine learning classifiers - K*, RF, and NB - using the BoW model, measured in terms of the average accuracy across 100 polysemous words. NB achieves the highest performance, with an average of 60.79%, indicating its effectiveness in handling sparse, high-dimensional data typical of BoW models. This aligns well with NB for its known efficiency in text classification tasks due to its probabilistic foundation and assumption of feature independence. RF performs moderately well, with an average score of 55.75%. K*, a variant of k-nearest neighbours, demonstrates the lowest performance, scoring only 41.65% on average.

How to cite this article:

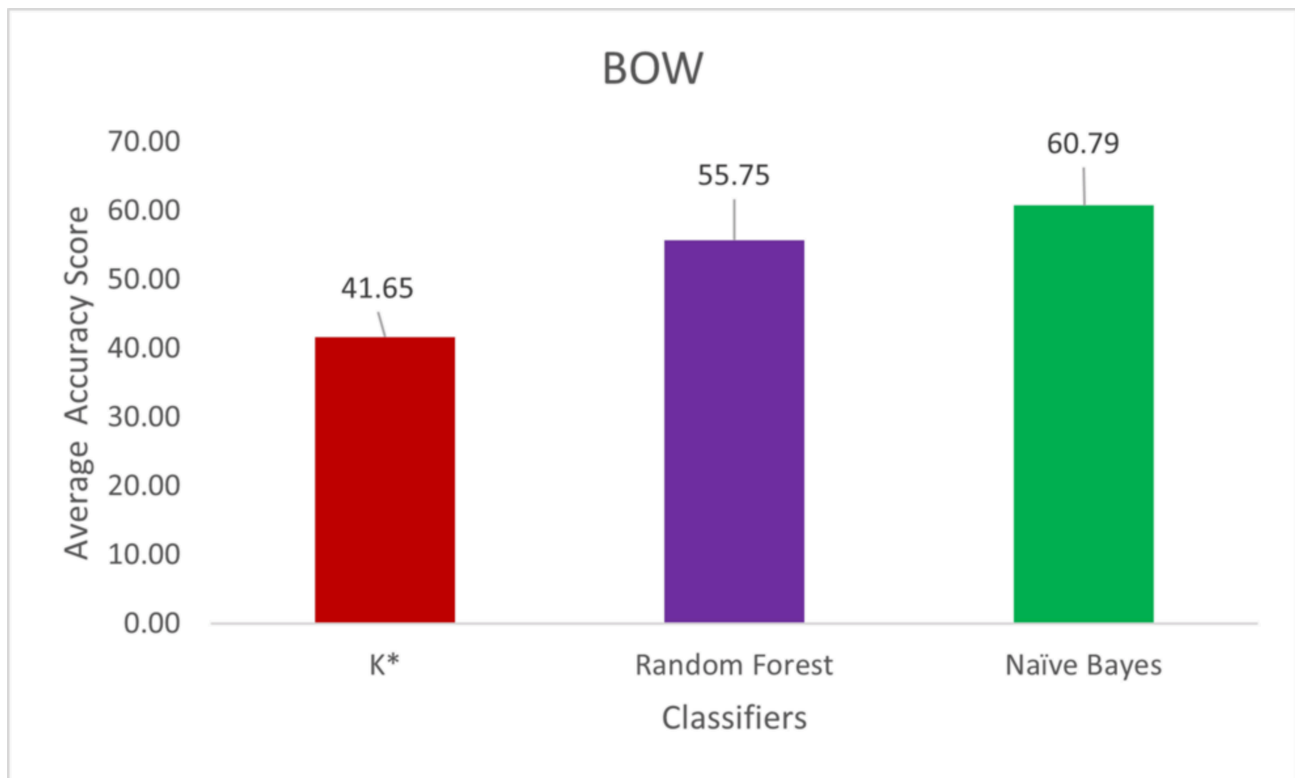


FIGURE 7: Comparison of average accuracy of classifiers with Bag of Words (BoW)

Figure 8 illustrates the performance comparison of NB, K*, and RF based on their average accuracy output over 100 words. The performance metric appears to reflect classification accuracy or effectiveness when utilizing the TF-IDF feature extraction technique. It is evident that the RF classifier outperforms the other two models, achieving the highest average value of 54.99. The K* classifier ranks second with an average of 51.95, demonstrating moderate performance. The NB classifier records the lowest average score of 47.25, which is consistent with its simplistic probabilistic assumptions. In summary, the RF model appears to be the most suitable choice for Bengali text classification tasks involving TF-IDF features in this context, followed by K* and NB.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

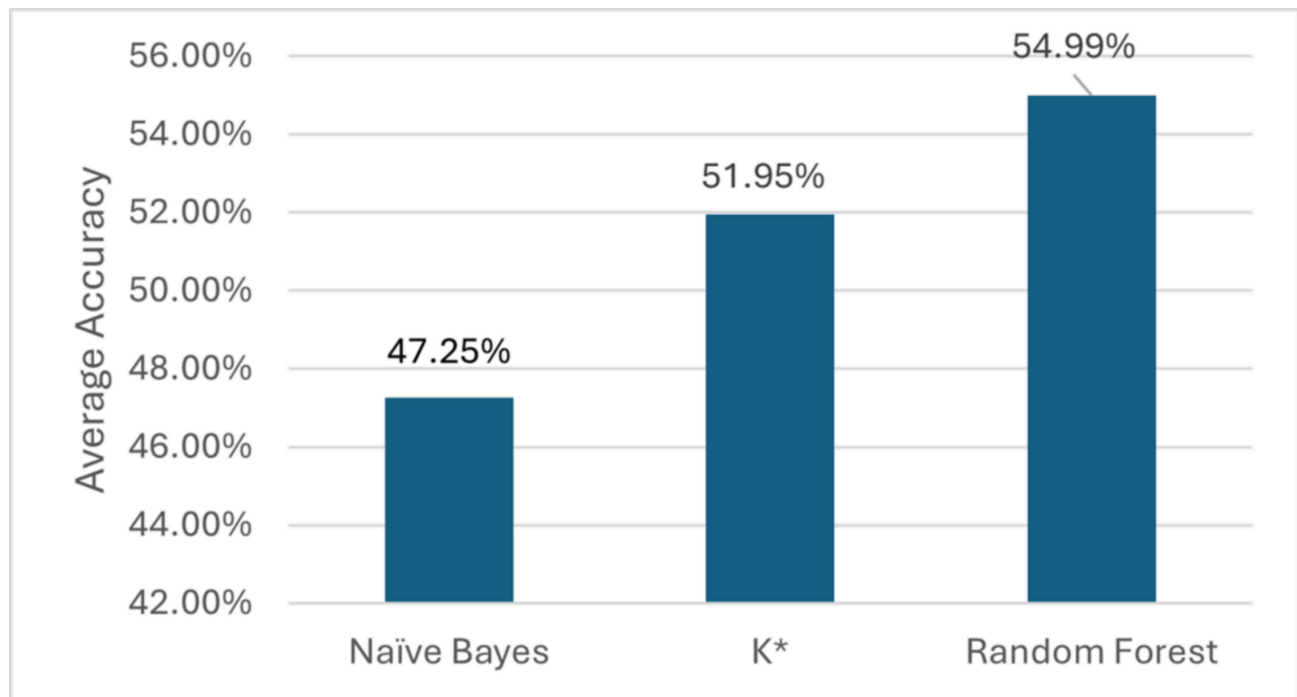


FIGURE 8: Average accuracy of classifier with term frequency-inverse document frequency (TF-IDF)

Figure 9 illustrates the average performance of five deep learning models, i.e., GRU, BiLSTM, RNN, CNN, and transformer on 100 polysemous words, when used in conjunction with FastText embeddings. Transformer outperforms all other models significantly with an average accuracy of 72%. CNN, RNN, and BiLSTM show comparable performance, with scores of 42.77%, 42.55%, and 42.27%, respectively. GRU achieves the lowest performance at 38.45%.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

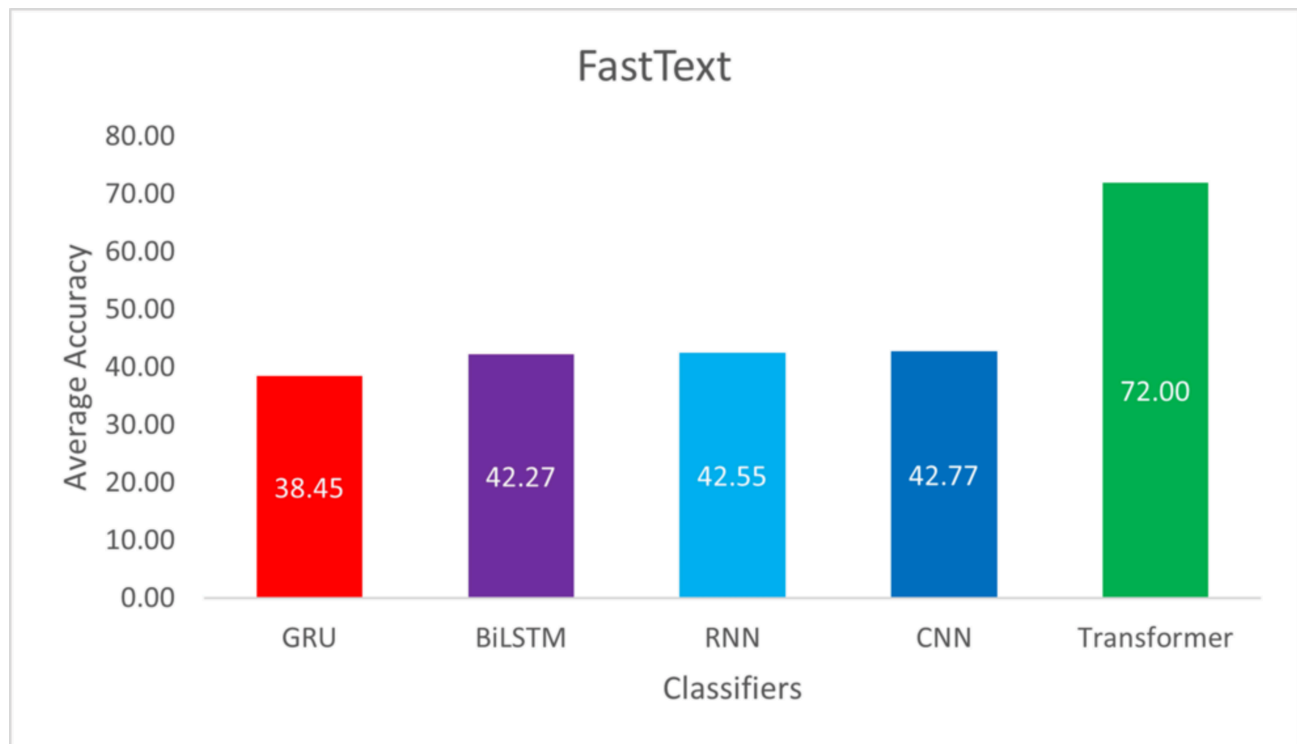


FIGURE 9: Comparison of average accuracy of classifiers with FastText

BiLSTM, Bidirectional Long Short-Term Memory; CNN, Convolutional Neural Network; GRU, Gated Recurrent Unit; RNN, Recurrent Neural Network

To evaluate the performance of the proposed model, which consists of a BiLSTM + multi-head attention architecture with pre-trained FastText Bengali embeddings, a comparative study is presented in Figure 10 against several baseline models. The results demonstrate a significant improvement achieved through context-aware and attention-based approaches. The FastText + BiLSTM + attention model outperforms the others, achieving an accuracy of 65%. Although BERT is a transformer-based model, its accuracy is nearly 40% in this case. Traditional models such as BoW + NB and TF-IDF + NB perform poorly, with accuracies of 13% and 15%, respectively. Word2Vec performs better than BoW.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

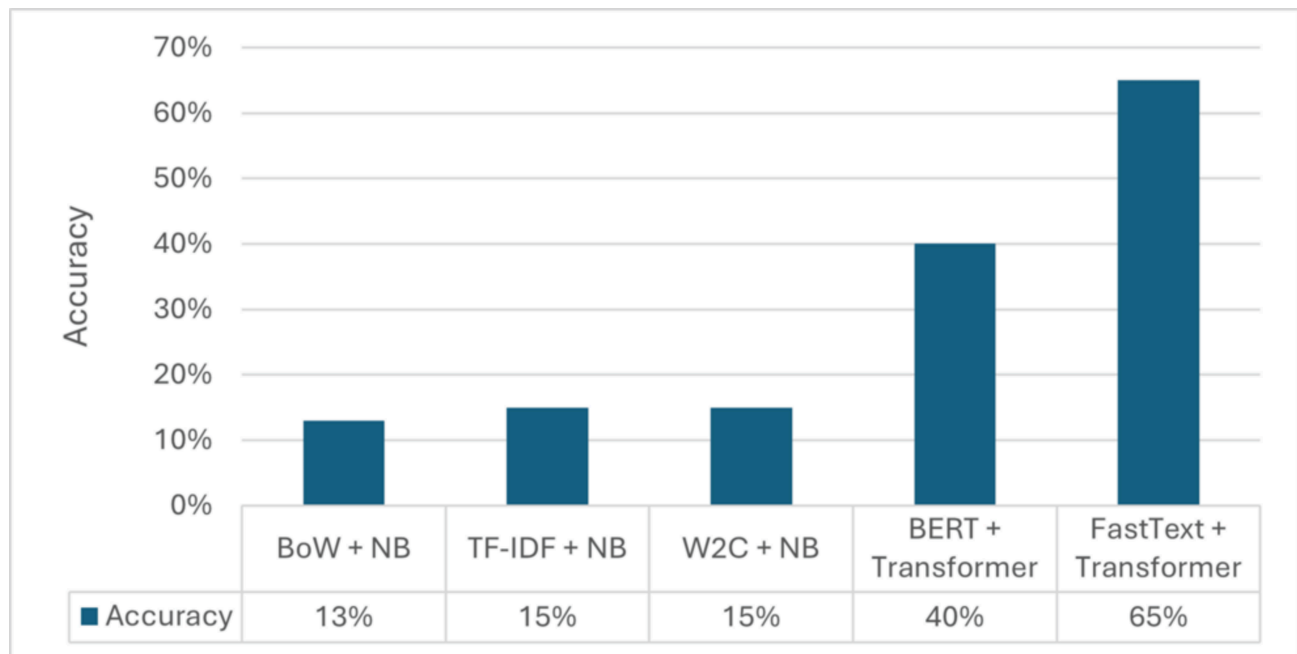


FIGURE 10: Comparison of classifiers

BERT, Bidirectional Encoder Representations from Transformers; BoW, Bag of Words; NB, Naive Bayes; TF-IDF, Term Frequency–Inverse Document Frequency; W2C, Word2Context

Figure 17 shows that lexical-based evaluation metrics focus on measuring the degree of similarity between the MT output and reference translations using n-gram overlaps and string-based comparisons. BLEU-3 and WER show high error rates of nearly 64%. NIST exhibits the highest error rate at 78.58%. METEOR performs the best in this category, achieving the lowest error rate of 40.65%.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

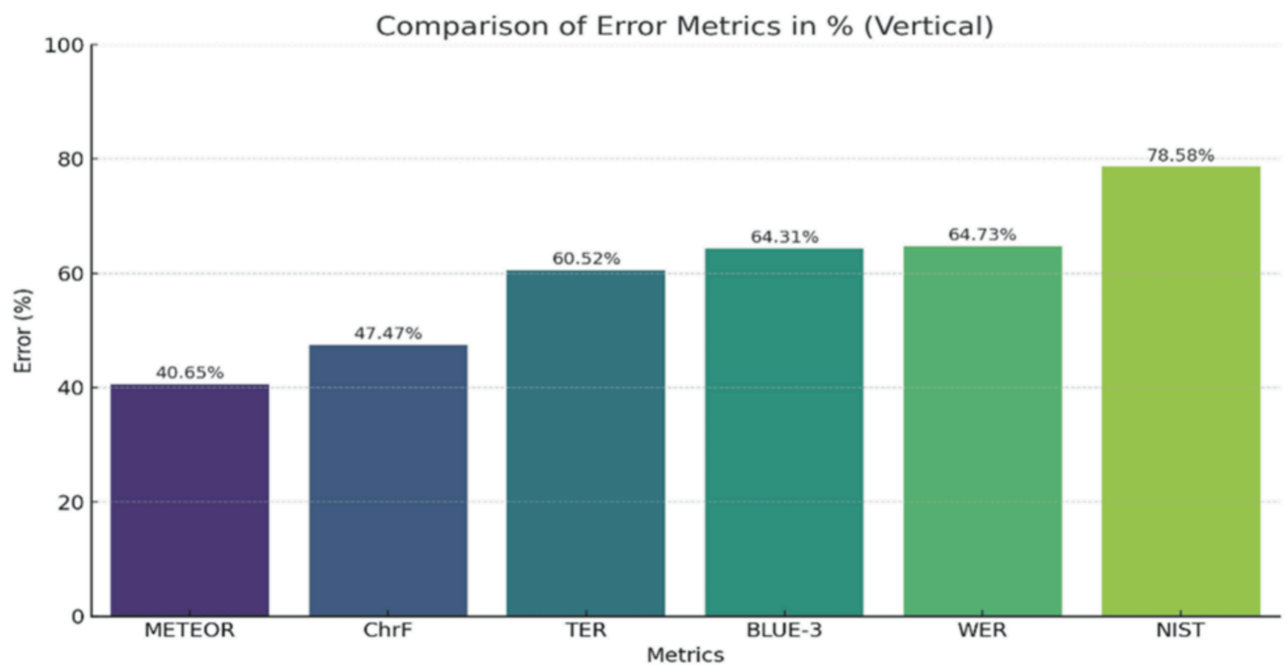


FIGURE 11: Lexical-based metrics

BLEU-3, Bilingual Evaluation Understudy (3-gram); ChrF, Character F-score; METEOR, Metric for Evaluation of Translation with Explicit ORdering; NIST, National Institute of Standards and Technology; TER, Translation Edit Rate; WER, Word Error Rate

Figure 12 illustrates that unsupervised evaluation metrics use deep learning models, such as BERT and BART, to assess the semantic similarity between human and machine translations. BERT score has the lowest error rate at 22.09%. The BART score error rate of 37.75% is higher than that of BERT score. Therefore, BERT score appears to be the most accurate method for evaluating translation accuracy.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

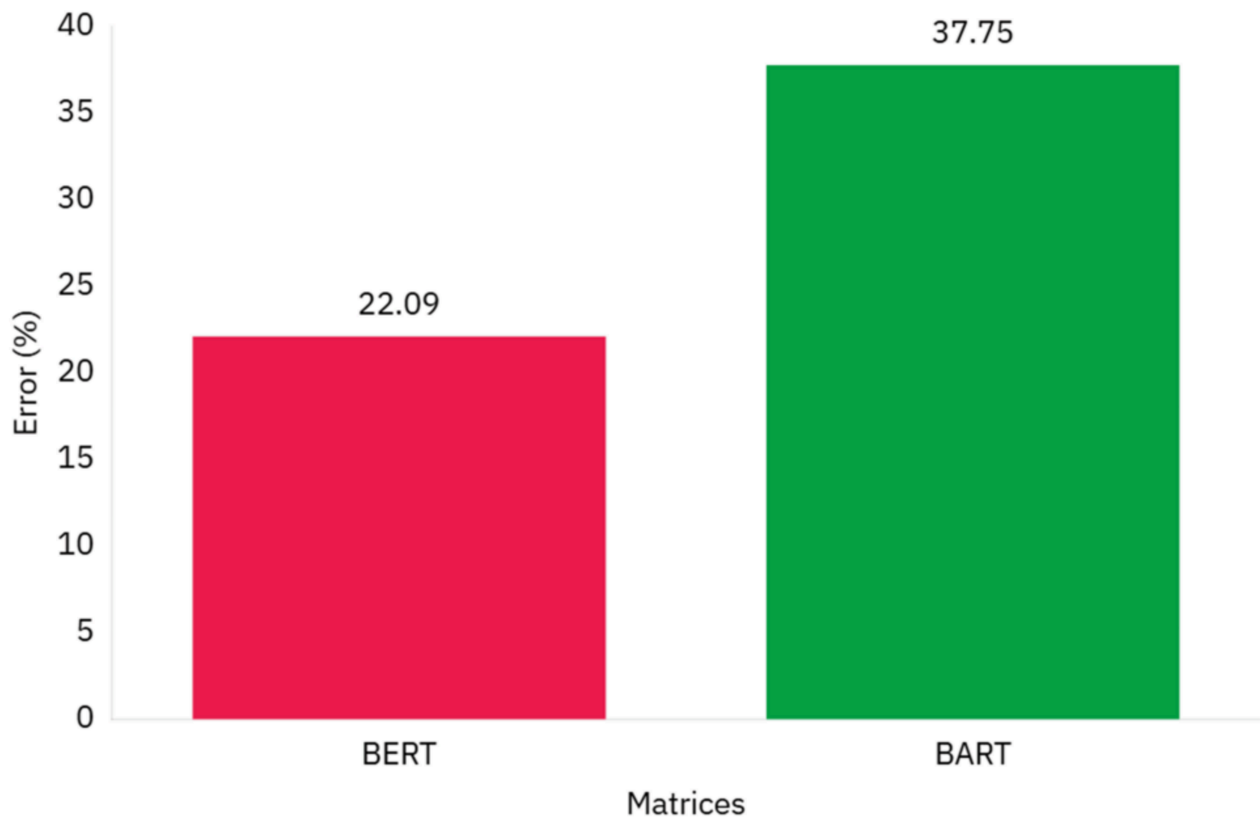


FIGURE 12: Unsupervised evaluation matrices

BART, Bidirectional and Auto-Regressive Transformer; BERT, Bidirectional Encoder Representations from Transformers

Figure 13 shows that supervised evaluation metrics leverage advanced neural network models trained on large datasets to provide more precise translation quality assessments. The BERT-MTE score has the lowest error rate at 12.41%, indicating its superior performance in capturing contextual meaning in Bengali translations. NUBIA and COMET, with scores of 39.31% and 42.17%, respectively, demonstrate moderate performance, suggesting that these models can capture meaning but still struggle with fluency and coherence. BLEURT, RUSE, and BLEND show higher error rates, indicating that these models are less effective at assessing translation errors in morphologically rich languages such as Bengali.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

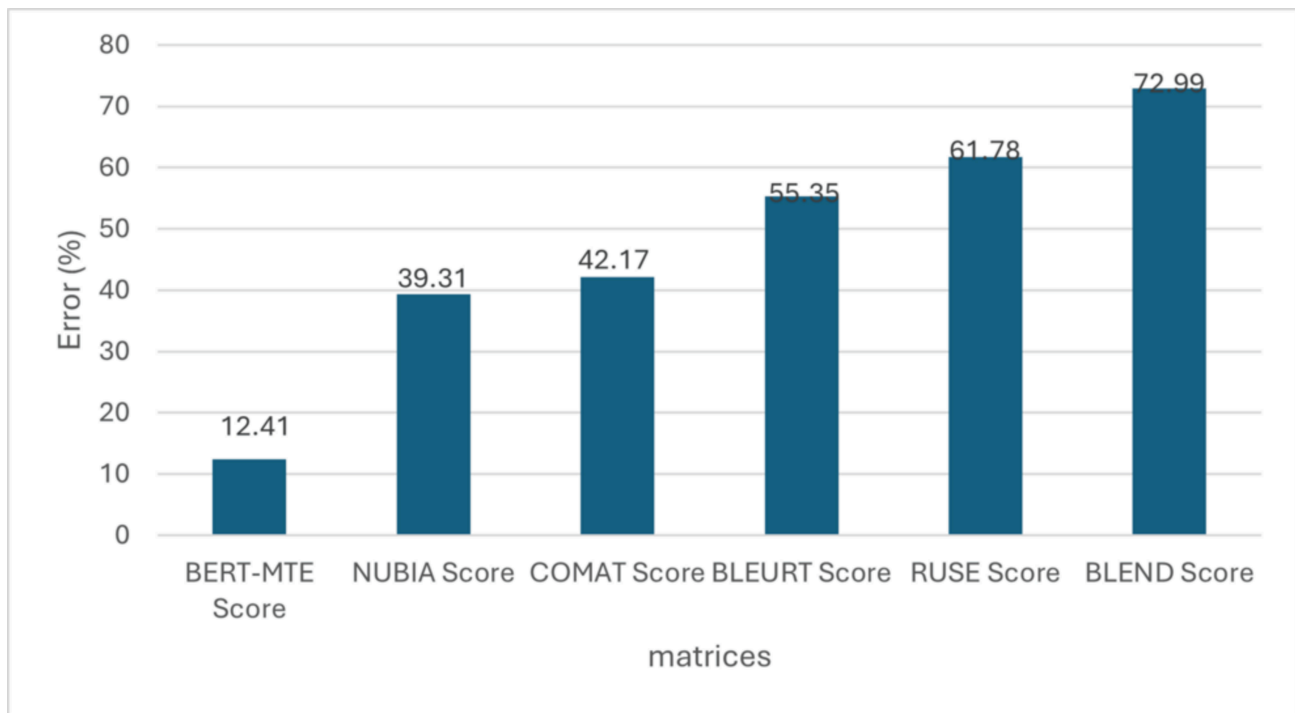


FIGURE 13: Supervised evaluation matrices

Figure 14 illustrates the percentage of error for each ambiguous word in the dataset when evaluated using our proposed model, which integrates a BiLSTM with transformer-based attention and FastText embedding-based architecture. The x-axis represents the 100 polysemous Bengali words (Word_1 to Word_100), while the y-axis indicates the corresponding prediction error expressed as a percentage (%). The error percentage is calculated as the proportion of incorrect sense predictions for each word relative to the total number of test instances involving that word.

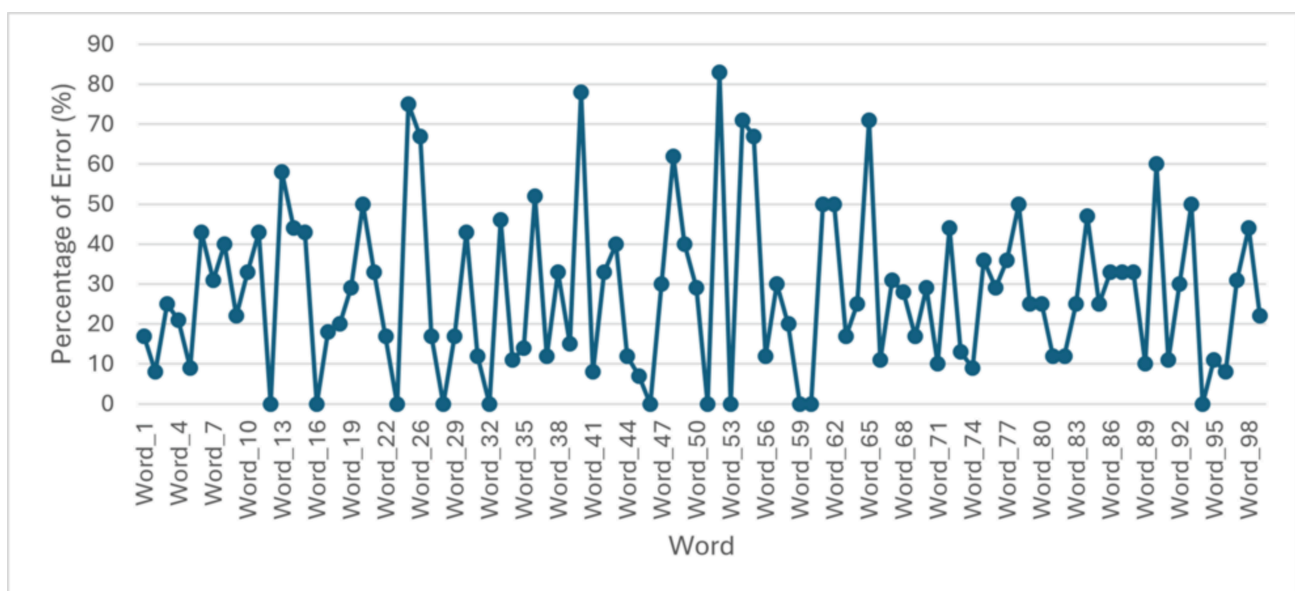


FIGURE 14: Error percentage of 100 words

How to cite this article:

The proposed model, which combines BiLSTM with transformer-based attention and FastText embeddings, achieves an error percentage of 14.89%. This outperforms Google Translate's error rate of 15.55% on the same dataset, which is shown in Figure 15.

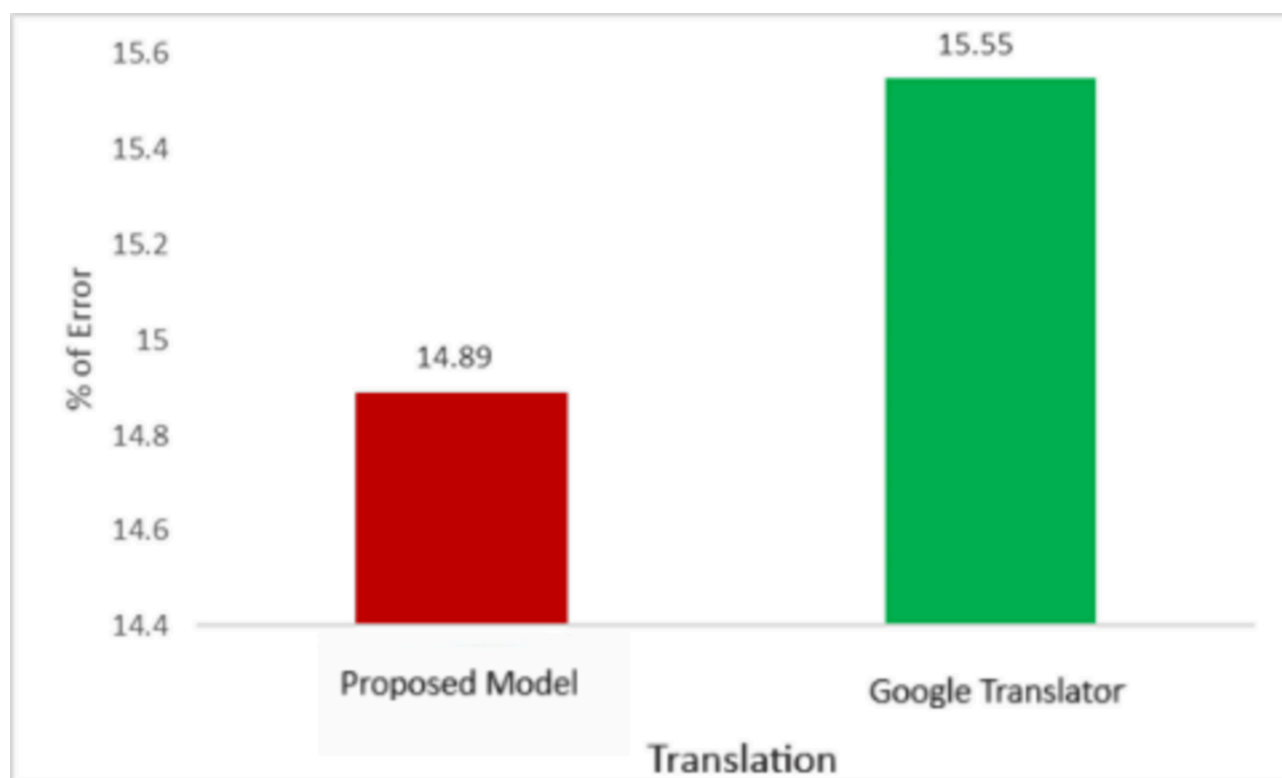


FIGURE 15: Error percentage of combine 100 words

The technical aspects of the proposed methodology are demonstrated by the example situations that are provided below.

Example 1:

Polysemous word: কাপ

Bengali Sentence: ধর্মীয় কাপ দেখে আপনি কেমন প্রতিক্রিয়া দেখান?

Sense: কপটতা

Transliteration: Dharmīya kâp dēkhē āpani kēmana pratikriyā dēkhān?

Human Translation: How do you react to religious hypocrisy?

Proposed Model Translation: How do you respond to religious hypocrisy?

BERT-MTE Score (Proposed vs Human Translation): 98.87%

Explanation: Here, our proposed model accurately detected the sense of the polysemous word "কাপ" as "কপটতা" (hypocrisy). It correctly replaced the ambiguous word with a contextually appropriate term, "hypocrisy", resulting in a very high similarity with the human translation.

Example2:

Polysemous word: Polysemous word: বেল

Bengali Sentence: বেলের জন্ম ভারত বর্ষে।

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

Sense: এক প্রকার ফল

Transliteration: Bêlêr janma bhārata barṣê.

Human Translation: Bael fruit originates from India.

Proposed Model Translation: Bell was born in India.

BERT MTE score (Proposed vs Human Translation): 55.21%

Explanation: Here, our proposed model fails to detect the correct sense of the polysemous word “বেল”, as “এক প্রকার ফল”, rather it senses it as a Surname. For this reason, the translation became inaccurate, resulting in a poor BERT-MTE score.

Discussion

The experimental analysis explains the results in this section, elucidating their significance, ramifications, and the reasons behind their occurrence. Figure 6 demonstrates that probabilistic models like RF may be better suited for WSD tasks, particularly in resource-constrained languages such as Bengali. The experimental results suggest that NB is the most effective classifier for BoW-based text classification in this context, as shown in Figure 7. Its simplicity, speed, and strong performance make it suitable for tasks involving large vocabularies and high-dimensional feature spaces. On the other hand, K* may not be a suitable choice for BoW representation due to its relatively poor performance. This may be due to the algorithm's sensitivity to high-dimensional data, where distance-based learning becomes less effective. Figure 8 shows that RF is more effective in capturing the underlying patterns of the TF-IDF features, likely due to its ensemble nature and the ability to handle high-dimensional data well. K* classifier, being an instance-based learner, may benefit from TF-IDF representations but could be limited by sensitivity to noise and computational cost. NB is generally efficient and performs well on text classification tasks, and its lower average in this case suggests that it may not fully leverage the discriminative power of TF-IDF features compared to the other models. It is seen from Figure 9 that transformer outperforms all other models significantly, indicating that it captures contextual relationships in word embeddings more effectively. This is consistent with current literature, as transformers are known for their attention mechanisms that excel at handling sequence data and long-range dependencies. CNN, RNN, and BiLSTM have close scores, suggesting that FastText provides a strong base representation, but these models may not fully leverage its subword information without additional tuning. GRU may not be as well-suited for this embedding setup or the task context, potentially due to its reduced complexity compared to other recurrent architectures. In summary, the results clearly highlight the superiority of the transformer model when using FastText embeddings, with a notable performance margin over traditional RNN-based and CNN-based architectures. This reinforces the growing preference for transformer-based models in modern NLP pipelines. Further exploration with hyperparameter tuning or hybrid models could help close the performance gap among the lower-performing models. Figure 10 shows that FastText + BiLSTM + attention model outperforms by capturing richer semantic and contextual information through pretrained FastText embeddings and BiLSTM with attention. While BERT is a transformer-based model, its lower accuracy is likely due to a lack of fine-tuning on the small dataset. Traditional models like BOW+ NB and TF-IDF + NB perform poorly because they ignore word order, context, and semantic meaning, and they treat each word as an independent feature without capturing the surrounding sentence structure. W2C, although better than BoW, lacks sentence-level compositionality and attention focus that leads to weaker disambiguation on polysemous Bengali words.

In Figure 11, the high error rates of BLEU-3 and WER indicate that simple n-gram matching struggles to capture meaning variations in Bengali text. The highest error rate is observed for NIST, suggesting that rare words, which typically carry significant meaning, are poorly handled by MT models. METEOR performs better than BLEU-3, likely due to its ability to recognize synonyms and word variations. TER indicates that substantial editing is required to correct MT outputs, reinforcing the need for improvements in current Bengali MT systems. In Figure 12, the lower error rate of BERT score highlights its effectiveness in capturing contextual meaning. BART score shows a higher error rate than BERT score, suggesting that although BART emphasizes fluency and structural coherence, it struggles more with semantic retention. Overall, BERT score provides a more reliable assessment of translation accuracy, significantly outperforming traditional lexical-based methods by focusing on semantic similarity rather than surface-level matching. From Figure 13, it is evident that BERT-MTE is the most effective among the supervised evaluation metrics, as transformer-based contextual

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

embeddings substantially enhance evaluation accuracy. NUBIA and COMET outperform lexical-based methods but still require improvement in fluency and coherence assessment. BLEURT, RUSE, and BLEND struggle with semantic preservation, leading to high error rates.

The variation of error rates in Figure 14 highlights the challenges posed by certain words due to their complex contextual dependencies or overlapping senses. Some words exhibit lower error rates, suggesting effective sense disambiguation by the model, while others, particularly those above 60% error, reveal areas where contextual cues were insufficient or where semantic overlap between senses led to misclassification. This analysis provides insight into the model's performance on a per-word basis and helps identify candidate words for future refinement. Although the 0.66% improvement shown in Figure 15 is modest, it demonstrates the model's scalability and its continued effectiveness in disambiguating word senses in context. The results further validate the efficacy of using contextualized embeddings and attention mechanisms for low-resource, morphologically rich languages such as Bengali in applications critical for WSD.

Conclusions

The analysis of Bengali MT errors underscores the pivotal role that WSD plays in enhancing the output quality of modern MT systems. Bengali, as a morphologically rich and context-sensitive language, poses inherent challenges to traditional translation models due to its frequent use of polysemous words, complex syntactic structures, and highly inflected word forms. These linguistic features complicate the mapping of source to target language, often leading to semantically inaccurate or contextually inappropriate translations. The calculation of MT errors resulting from improper sense selection of polysemous Bengali words is the main topic of this research. A hybrid technique is suggested to reduce these errors by implicitly taking WSD into account. BoW, TF-IDF, Word2Vec, and FastText are among the word embedding methods that are examined. FastText is found to be the most successful since it handles Bengali morphology at the subword level. To maintain the accurate translation of named things and culturally significant phrases, a customized label is introduced for exact sense mappings. The experimental results demonstrated that conventional lexical-based evaluation metrics such as BLEU-3, WER, and NIST perform inadequately when applied to Bengali MT system, with observed error rates exceeding 60%. These metrics primarily rely on surface level word overlap and n-gram precision, which makes them poorly suited for capturing deeper semantic relationships and disambiguation challenges present in Bengali. In contrast, deep learning-based evaluation metrics, particularly BERT Score and BERT-MTE Score, yielded substantially lower error rates of 22.09% and 12.41%, respectively. These metrics leverage transformer-based language models and contextual embeddings, enabling them to assess semantic similarity at a deeper level. Their superior performance illustrates the growing importance of context-aware architectures in both evaluation and development of MT systems, especially for low-resource and morphologically complex languages. The proposed model achieves an error percentage of 14.89% by combining FastText embeddings, transformer-based attention, and BiLSTM. On the same dataset, this performs better than Google Translate's error rate of 15.55%. The outcomes further support the effectiveness of contextualized embeddings and attention methods for morphologically rich, low-resource languages like Bengali in WSD-critical applications. These findings affirm that future research in Bengali MT should prioritize contextual understanding, semantic disambiguation, and adaptive evaluation methods to bridge the gap between human-like translation and current system capabilities.

Future efforts may explore the following research directions to further enhance the quality and adaptability of Bengali MT systems. Our approach targets a single ambiguous word per sentence. However, real-world sentences frequently contain multiple polysemous terms whose meanings are interdependent. Future systems must employ multi-word disambiguation strategies. Future MT architectures should focus on real-time adaptation through interactive learning. By incorporating user feedback or contextual hints dynamically during the translation process, these systems can continuously refine their understanding of word senses, leading to more accurate and personalized translations in live settings. A promising direction involves the development of cross-linguistic alignment frameworks using multilingual embeddings (e.g., mBERT, XLM-R) and lexical resources like WordNet or IndoWordNet. These models can facilitate better semantic correspondence between languages, especially when transferring knowledge from resource-rich languages (e.g., English) to low-resource languages.

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. *Cureus J Comput Sci* 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Taniya Seal, Debapratim Das Dawn, Abhinandan Khan, Rajat Kumar Pal

Acquisition, analysis, or interpretation of data: Taniya Seal, Subhasish Mukherjee, Ankit Saha, Prajjal Mukherjee, Debapratim Das Dawn, Abhinandan Khan, Sanjit Kumar Setua

Drafting of the manuscript: Taniya Seal, Subhasish Mukherjee, Ankit Saha, Prajjal Mukherjee, Sanjit Kumar Setua, Rajat Kumar Pal

Supervision: Taniya Seal, Debapratim Das Dawn, Abhinandan Khan, Sanjit Kumar Setua, Rajat Kumar Pal

Critical review of the manuscript for important intellectual content: Debapratim Das Dawn, Abhinandan Khan, Sanjit Kumar Setua, Rajat Kumar Pal

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Acknowledgements

Data Availability: The datasets and code generated and/or analyzed during this study are included in this article or are available from the corresponding author upon reasonable request.

References

1. Singh M, Kumar R, Chana I: [Machine translation systems for Indian languages: Review of modelling techniques, challenges, open issues and future research directions](#). Archives of Computational Methods in Engineering. 2021, 28:2165-2193. [10.1007/s11831-020-09449-7](#)
2. Paul S: [Bilingual \(English to Bengali\) technical E dictionary for aviation OOV words](#). International Journal of Engineering and Advanced Technology. 2019, 9:3777-3781. [10.35940/ijeat.b3572.129219](#)
3. Vidhu Bhala RV, Abirami S: [Trends in word sense disambiguation](#). Artificial Intelligence Review. 2014, 42:159-171. [10.1007/s10462-012-9331-5](#)
4. Roberto N: [Word sense disambiguation: A survey](#). ACM Computing Surveys. 2009, 41:1-69. [10.1145/1459352.1459355](#)
5. Pal AR, Saha D, Naskar SK, Dash NS: [In search of a suitable method for disambiguation of word senses in Bengali](#). International Journal of Speech Technology. 2021, 24:439-454. [10.1007/s10772-020-09787-8](#)
6. Ahmed L, Polok IK, Islam MA, Akhtaruzzaman M, Mukta MSH, Rahman MM: [Context based emotion recognition from Bengali text using transformers](#). 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli. 2023, 1478-1484. [10.1109/ICSSIT55814.2023.10060927](#)
7. Pu X, Pappas N, Henderson J, Popescu-Belis A: [Integrating weakly supervised word sense disambiguation into neural machine translation](#). Transactions of the Association for Computational Linguistics. 2018, 6:635-649. [10.1162/tacl_a_00242](#)
8. Zobaed S, Haque ME, Rabby MF, Salehi MA: [SensePick: Sense picking for word sense disambiguation](#). 2021 IEEE 15th International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA. 2021, 318-324. [10.1109/ICSC50631.2021.00060](#)
9. Rippeth E, Carpuat M, Duh K, Post M: [Improving word sense disambiguation in neural machine translation with salient document context \[PREPRINT\]](#). arXiv. 2023, [10.48550/arXiv.2311.15507](#)

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. Cureus J Comput Sci 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>

10. Iyer V, Barba E, Birch A, Pan JZ, Navigli R: [Code-switching with word senses for pretraining in neural machine translation \[PREPRINT\]](#). arXiv. 2023, [10.48550/arXiv.2310.14050](#)
11. Papineni K, Roukos S, Ward T, Zhu WJ: [BLEU: A method for automatic evaluation of machine translation](#). Proceedings of the 40th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, USA; 2002. 311-318. [10.3115/1073083.1073135](#)
12. Banerjee S, Lavie A: [METEOR: An automatic metric for MT evaluation with improved correlation with human judgments](#). Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. Association for Computational Linguistics, Ann Arbor, MI; 2005. 65-72.
13. Agarwal A, Lavie A: [METEOR, M-BLEU and M-TER: Evaluation metrics for high-correlation with human rankings of machine translation output](#). StatMT '08: Proceedings of the Third Workshop on Statistical Machine Translation. Association for Computational Linguistics, USA; 2008. 115-118. [10.3115/1626394.1626406](#)
14. Lee S, Lee J, Moon H, et al.: [A survey on evaluation metrics for machine translation](#). Mathematics. 2023, 11:1006. [10.3390/math11041006](#)
15. [Bengali Word Sense Disambiguation \(WSD\) Dataset](#). (2025). Accessed: September 25, 2025: <https://www.kaggle.com/datasets/debapratimdasdawn/bengali-wsd-dataset>.
16. Das Dawn D, Khan A, Shaikh SH, Pal RK: [Likelihood corpus distribution: an efficient topic modelling scheme for Bengali document class identification](#). Sādhanā. 2024, 49:198. [10.1007/s12046-024-02470-7](#)

How to cite this article:

Seal T, Das Dawn D, Saha A, et al. (February 27, 2026) Computation of Machine Translation Errors Through Implicit Consideration of Word Sense Disambiguation in Bengali Text. Cureus J Comput Sci 3 : es44389-025-00026-2. DOI <https://doi.org/10.7759/s44389-025-00026-2>