Morphological Analysis and Disambiguation of Agglutinative Assamese through Rule-Based Synthesis and Neural Sentence Scoring

Anonymous EMNLP submission

Abstract

Assamese is a highly agglutinative language that uses suffix sequences to convey information regarding case, quantity, number, person, tense, certainty, and more. Assamese root words also undergo derivation that affects the succeeding suffixation. Surface form of some suffixes is multi-functional demonstrating different functions based on the context and meaning of the preceding morphemes. These ambiguities along with the lack of an annotated dataset renders morphological analysis difficult for Assamese. Our approach to automatic morphological analysis is based on the fact that there are only a few legal analyses for a word, and one of them is the most appropriate to the context it is used. Using a refined morpheme tag-set, derivational-rules and a tag-transition rule-set we produce the legal morphological analyses of the words in a sentence. We manually annotated around 8k context-based morphological analyses. By using the analyses as morphological feature vectors of corresponding tokens, we trained a sentence-level encoder to discriminate between correct and incorrect sequence of analyses. The trained model is evaluated on the task of assigning the best sentence-level $grammar\ score$ to the correct $analysis\ sequence$ among all the candidates. Accuracy of the hybrid pipeline on morphological analysis is up to 91.42% with disambiguation accuracy of 83.42%. The system is a new baseline for morphological analysis in Assamese on the novel dataset. Experiments further strengthen the idea that − analysis of the context is useful for resolving morphological ambiguities in agglutinative languages.

1 Introduction

Like many other Indian languages Assamese is agglutinative, form word-groups, but otherwise exhibit a free word-group ordering. In such scenarios position-based parsing algorithms are not sufficient for understanding the sentence structure, which is often required for tasks like shallow semantic parsing, information extraction, and other downstream applications. However, the deficiency in strict parse trees is compensated by the rich suffixation phenomena of such languages, thereby making automatic morphological analyzers essential (Seeker and Kuhn, 2013; Yildiz et al., 2016; Sharma et al., 2008). The suffix sequences in Assamese provide various information about the preceding word-groups, like case, quantity, number, person, tense, certainty, etc. and helps in sentence understanding (Roy and Purkayastha, 2017; Das et al., 2019). An example for such phenomena is given in Figure 2. Prefixation phenomenon in Assamese is comparatively less diverse and prefixes do not concatenate.

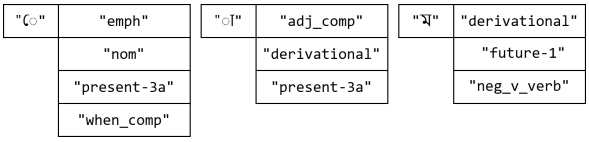

Figure 1: Example of multi-functional surface form of morphemes.

In Assamese and other agglutinative languages the suffixation on a root morpheme is guided by rules (Das et al., 2019; Yildiz et al., 2016). However, in Assamese the same surface realization of a suffix may belong to multiple classes and perform different functions based on the type of root it attaches to (Yildiz et al., 2016). In investigation of Assamese derivational suffixes, Gayan and Nath (Gayan and Kumar Nath, 2025) discovered that some derivational and inflectional suffixes have the same surface realization. Such a suffix might derive the root word into another form, subsequently affecting the succeeding suffixation. It is also common in Assamese for an inflectional suffix to serve multiple functions (see Figure 1) (Das et al., 2019; Shen et al., 2016). Besides multi-functional affixes, homonymy of root morphemes amplify the problem of producing multiple valid analyses of a single

word.

To address these problems we present a hybrid pipeline that uses a rule-based analyzer and linguistic characteristics to produce multiple legal analyses of a word. If more than one analysis is produced, one of them will be the most appropriate. Earlier, it has been shown that the context plays a crucial role in morphological disambiguation of agglutinative languages (Yildiz et al., 2016; Shen et al., 2016). Therefore, we consider the possibility that, to find the correct analyses, the morphological analyses of the context words in the sentence can be used. According to our formulation – the morphological features of the words in a sentence will follow some patterns of *morphological grammar*. To test this hypothesis, we conduct experiments on a manually annotated dataset and obtain good results. In the later sections we elaborate the details regarding linguistic knowledge, methodology, system, and results.

2 Related Works

Morphological disambiguation has been extensively studied for morphologically rich and highly agglutinative languages. Shen et al. (Shen et al., 2016) proposed a language-agnostic neural approach for morphological disambiguation that leverages contextual information, and evaluated their method on Turkish, Russian, and Arabic. Their approach, however, relies on large semi-automatically annotated datasets, which are currently unavailable for Assamese. Similar context-based approaches have also been explored for highly agglutinative languages such as Turkish, German, French, and Kazakh (Yildiz et al., 2016; Kessikbayeva and Cicekli, 2014). Nevertheless, adapting such approaches directly to Assamese remains challenging due to substantial linguistic differences and the difficulty of transferring language-specific morphological knowledge across languages.

For Indian languages, context-aware morphological disambiguation remains largely underexplored. Dasari et al. (Dasari et al., 2023) created a morphologically analyzed sentence-level dataset for Telugu and evaluated pretrained language models for morpheme-tag classification. Their experiments demonstrated that monolingually pretrained models outperform multilingual models (Dasari et al., 2023), a phenomenon that becomes particularly relevant for Assamese due to the severe imbalance in multilingual pretraining corpora and the comparatively limited availability of Assamese textual resources.

Previous research on Assamese morphology has primarily focused on identifying morphological rules and linguistic characteristics (Roy and Purkayastha, 2017; Das et al., 2019). Sharma et al. (Sharma et al., 2008) investigated morphology acquisition from corpus data by decomposing words into a root followed by a sequence of suffixes, demonstrating that a single word can admit multiple valid decompositions. However, even when a word has a unique decomposition, determining the correct morphological tag often requires sentence-level contextual understanding because of the multifunctional nature of both roots and affixes in Assamese.

To the best of our knowledge, a context-aware sentence-level morphological disambiguation framework for Assamese, or for other Indian languages with similar morphological complexity, has not been previously explored.

3 Assamese Morphology

Affixation is a major morphological phenomenon in Assamese and plays a crucial role in computational language processing tasks (Sharma et al., 2008). Disambiguation of suffixes is particularly important for determining the part-of-speech (PoS) of Assamese words, as derivational processes are highly prevalent in the language (Roy and Purkayastha, 2017). In surface forms, affixes frequently undergo phonological and orthographic mutations depending on the boundary characters of adjacent morphemes (see Figure 3). These transformations result in multiple surface realizations of the same suffix. In our approach, all such surface variants are mapped to a unified morphological tag category.

Beyond surface-level variations, we observe Assamese morphology to largely follow the Item-and-Arrangement model, where words are formed through structured sequences of morphemes. The arrangement of these morphemes is governed by morphotactic constraints, many of which have not yet been comprehensively formalized. Some of these rules have been discussed in prior work on the computational morphology and syntax of Assamese (Saharia, 2014). In this work, we further emphasize the importance of finer-grained morpheme subcategorization and additional morpho-

Assamese raw sentence	এই	ৰাজ্যৰ	নিৰ্বাচনতে	দৃষ্টি	ৰাখিছে	সমগ্ৰ	দেশে	।
Surface level morphology	**এই**	**ৰাজ্য**+ৰ	**নিৰ্বাচন**+ত+ে	**দৃষ্টি**	**ৰাখ**+িছে	**সমগ্ৰ**	**দেশ**+ে	।
Morphological features	PN	NNA;gen	NNA;dat;emph	NNA	Verb;present;3p;a	ADJ	NNA;nom	।
English form	**This**	**state**+gen	**election**+dat+emph	**observation**	**keep**+present+3p+a	**entire**	**country**+nom	punct
English translation	***The entire country is keeping an eye on this state's election.***							
Alternative ordering	সমগ্ৰ দেশে এই ৰাজ্যৰ নিৰ্বাচনতে দৃষ্টি ৰাখিছে।							

Figure 2: Morphological analysis of a complete Assamese sentence.

সংস্থাটোৱে	['সংস্থা', 'Abstract Noun'], ['টো', 'number'], ['ৱে', 'nom']
উপায়ুক্তই	['উপায়ুক্ত', 'Common Noun'], ['ই', 'nom']
গৱেষকসকলে	['গৱেষক', 'Common Noun'], ['সকল', 'number'], ['ে', 'nom']

Figure 3: Multiple surface level realization of the nominative suffix in Assamese

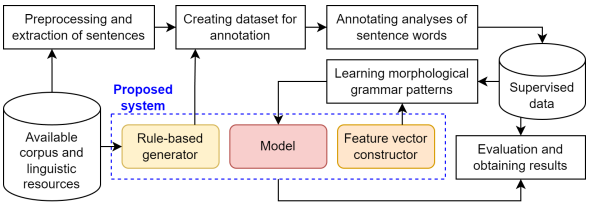

Figure 4: Overview of the methodology with various components of the proposed system.

tactic constraints beyond those previously documented. For example, among case suffixes, only the absolutive case can follow another case suffix, whereas other case suffixes generally do not concatenate with each other.

Pronouns in Assamese do not typically occur in explicit oblique forms. Instead, they inherently encode case, person, number, and honorific information. Similarly, verbal suffixes express a diverse range of grammatical functions. Certain suffixes encode tense, person, and honorific agreement, while others express relational or compositional functions through tags such as `adj_comp`, `reason_comp`, and `after_comp`. These properties give rise to phenomena related to government and agreement, which are prevalent in Assamese sentence structures. In particular, verb morphology exhibits agreement with nominal or pronominal subjects in terms of person, case, and honorificity. Additionally, factors such as nominal animacy and verb category (e.g., transitive or intransitive) influence the selection of case suffixes (Bhattacharjya, 2012).

These linguistic characteristics are advantageous for our proposed approach because they can be systematically modeled computationally and provide strong contextual cues for resolving ambiguity among multiple valid morphological analyses. Additional computationally relevant morphological characteristics are discussed in Appendix B.

4 Rule-based Analyzer

Our rule-based analyzer utilizes *XOBDO* (XOBDO, 2026), a collaboratively developed lexical resource containing static PoS-tagged base forms along with their semantic information. From this resource, we construct a lexical dictionary (Dictionary-A) consisting of words and their associated PoS categories. To improve morphological coverage, we augment this dictionary with multiple surface realizations of 35 suffix categories. These suffix variants were extracted from the Assamese Wikipedia corpus using a suffix-stripping approach combined with manual linguistic analysis (Sharma et al., 2008).

In addition to suffixes, we incorporate a limited set of prefixes belonging primarily to two categories: (i) negation prefixes, which occur with a restricted set of verbs, and (ii) numerical prefixes, which typically attach to determiners. Due to the multifunctional nature of Assamese morphology, several morphemes belong to multiple grammatical categories, including PoS classes. A detailed description of the morpheme tags and associated linguistic features is provided in Appendix A.

The primary objective of the analyzer is to generate all valid morphological analyses for a given word. To achieve this, we incorporate both derivational and inflectional morphotactic constraints within the analysis process. The major components of the analyzer are described in the following subsections.

4.1 Derivational Rules

We observed the presence of derived words in Dictionary-A. Therefore, we apply the suffix-stripping method in order to investigate the presence of derivational morphemes (Sharma et al., 2008). We take a very large rule frequency (50)

to exclude spurious matches. We obtained 54 derivational suffixes that convert at least 50 base words into derived lexical forms. Many of these suffixes match with those discussed by Roy and Purkayastha (2017) and Gayan and Kumar Nath (2025). For each derivational suffix s, we estimate the probability of a part-of-speech transition from tag T_A to tag T_B using dictionary-level transition frequencies:

$$P(T_A \rightarrow T_B \mid s) = \frac{Count(T_A \rightarrow T_B \mid s)}{\sum_i \sum_j Count(T_i \rightarrow T_j \mid s)}$$

where $Count(T_A \rightarrow T_B \mid s)$ denotes the number of observed derivational transformations in which suffix s converts a base word of category T_A into a derived form of category T_B. During inference, the most probable derivational transformation for any tag T induced by suffix s is selected, to limit the number of legal analyses.

$$T_{derived} = \underset{i}{\operatorname{argmax}}\, P(T \rightarrow T_i \mid s)$$

4.2 Bigram Transition Rule-set

To constrain the generation of morphologically valid analyses, we manually constructed a morphotactic transition rule-set describing legal category-level transitions between adjacent morphemes belonging to tags T_i and T_j. The rule-set was developed using linguistic analysis of Assamese suffixation patterns and with the help of previous studies (Saharia, 2014; Das et al., 2019). Special boundary states corresponding to word-start (ws) and word-end (we) were additionally introduced to model valid word initialization and termination conditions. The rule-set is converted into a transition matrix $\mathcal{R} \in \{0,1\}^{n \times n}$ that is used by the generator, where:

$$\mathcal{R}(T_i, T_j) = \begin{cases} 1 & \text{if transition } T_i \rightarrow T_j \text{ is legal} \\ 0 & \text{otherwise} \end{cases}$$

The rule-set can definitely be improved to capture longer sequence of patterns with the advent of superior linguistic knowledge for the language.

4.3 Generator

The generator produces all valid morphological analyses $A_w = \{a_0, a_1, ..., a_n\}$ of a word w. Each of the analysis a_i is a morphological segmentation of the surface form with the sequence of tags

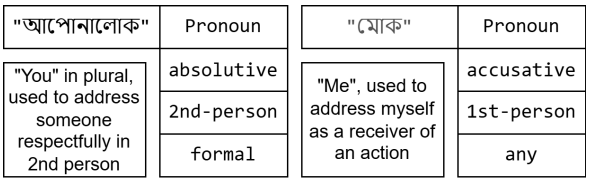

Figure 5: Example: case and other information integrated with pronouns.

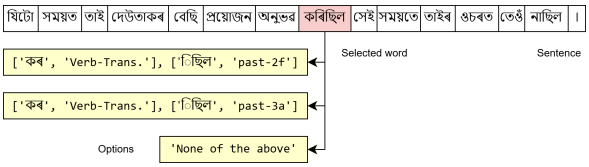

Figure 6: A sample from the annotation set. An annotator is supposed to select the most correct analysis or mark a word as 'None of the above' for manual investigation.

$T_{i,1}T_{i,2}...T_{i,k}$. The generator incorporates the possibility of PoS-tag transition due to derivation and the presence of integrated case markers in Assamese Pronouns (see Figure 5). During traversal, only transitions permitted by $\mathcal{R}$ are explored, thereby pruning structurally invalid morpheme sequences and restricting the search space to linguistically valid analyses (algorithm 2 in appendix D). The rule-based analyzer contributes highly to the accuracy in morphological analysis. The performance of this generator can be known through some entries in Table 1.

5 Annotation Process

We sampled the sentences for annotation from the 'Sangraha' corpus (Khan et al., 2024), which is a good representation of Assamese texts available on the internet. Because the dataset contains a lot of noise, transliterations and spelling errors, only a clean part was selected.

To make the annotation process simpler, we chose only those sentences that contain words with 6 or less legal morphological analyses. On the interface, for each word w in a sentence the available legal analyses $A_w = \{a_0, a_1, ..., a_k\}$ generated by the rule-based analyzer are displayed with an additional option to choose 'none of the above' (see Figure 6). The task is to select the most appropriate analysis $a_w \in A_w$ based on the context as understood by the annotators. For word where none of the analyses is correct, the sample is marked for manual analysis.

After annotation process is done, it was found

that around 300 words required manual analysis. On investigation, it was found that to incorporate these words, the bigram-transition rules have to be updated, which exponentially increases the search space. It was finally decided to drop the sentences containing these words.

6 Neural Disambiguation

Multiple studies (Shen et al., 2016; Dasari et al., 2023; Yildiz et al., 2016) have emphasized the role of context in morphological disambiguation task for agglutinative languages. This motivates us to use context to disambiguate between the legal analyses of Assamese words. We formulate the task as analyzing the complete sentence at once by giving it a *grammar score*.

6.1 Morphological Vector

In contrast to other approaches, we make use of a morphological feature vector instead of character or morpheme embeddings. The reason is a lack of large datasets that could accommodate a rich contextual distribution of base words. Moreover, it allows us to test our hypothesis that, for assigning a *morphological grammar score*, whether the analysis of the words in a sentence is sufficient.

We represent the morphology of a word using a custom 64 dimensional feature vector that captures the PoS tags observed in the word, presence of derivation, prefix types and suffix types. Let a_w be a legal morphological analysis of the word w that can be represented as a sequence of tags $T_1T_2..T_k$, then the feature vector f_w can be obtained using algorithm 3 (appendix D).

6.2 Transformer Encoder

We use a custom transformer encoder block because of its superior pattern recognition capabilities across sequence classification tasks (Vaswani et al., 2017). A sequence of morphological feature vectors is the raw input to the encoder illustrated in Figure 8. The input is projected onto a 16 dimensional plane and positional encoding is added.

6.3 Training Objective

As the rule-based generator can produce multiple legal morphological analyses of a word, a sentence can can be represented by multiple sequences of morphological features (see Figure 2) and morphological vectors. However, out of them only one is correct (correct and incorrect samples are illustrated in Figure 7). Therefore, the proposed model is trained as a binary sequence classification model that assigns a *grammar score* to an input sequence. The model is optimized using Binary Cross Entropy (BCE) loss.

Category	Value
Total sentences	792
Total word tokens	7962
Unique tokens	3011
Ambiguous tokens	4125
Unambiguous tokens	3837
Average candidate analyses	1.91
Maximum candidate analyses	6
Total morphological tags	59
Total suffix categories	35
Average sentence length	10.05 tokens
Longest sentence length	29 tokens
Shortest sentence length	4 tokens
Training set size	673 sentences
Test set size	119 sentences

Table 1: Summary statistics of the annotated corpus used for morphological analysis experiments. Contribution from rule-based analyzer is in **bold**.

7 Experimental Setup

The hybrid pipeline has been depicted in Figure 4 containing the methodological process of annotation and updation of the rule-based component.

7.1 Annotated Dataset

The dataset contains index of the correct analysis for each word of the sentences. This means that there exist a *gold* sequence of morphological vectors for a sentence, while others are negative. The tailored dataset statistics is provided in the Table 1. For each sentence we select up to K_n negative samples by swapping a correct analysis from the gold sequence. We also consider augmenting the dataset with upto K_a random-offset padding for each sample because the sentences are of uneven length and do not fill the entire context of the encoder. For $K_n = 2$ and $K_a = 3$ there are a total of 3 positive and 3 negative samples generated from one sentence.

7.2 Training Configuration

The model is trained using the binary sequence classification objective described in Section 6. Sentence-level train-test splitting is performed using unique sentence identifiers to prevent augmented variants of the same sentence from appearing in both partitions. Experiments are conducted using different configurations of model dimension, negative sampling (K_n) and augmentation (K_a)

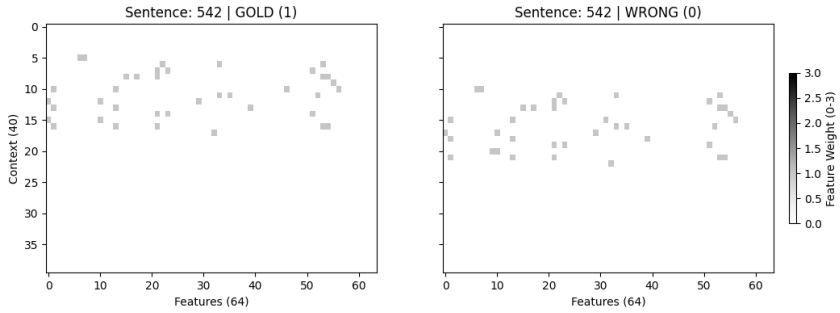

Figure 7: Example of two input representation for encoder. Sequence of feature vectors of a positive and negative sample respectively for the same sentence. The vectors with all feature weights as zero indicate padding vector.

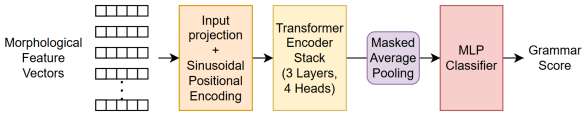

Figure 8: Transformer encoder for discriminating between correct and incorrect sequence of analyses.

Hyperparameter	Value(s)
Context window size	40
Feature dimension	64
Model dimension	{32, 64}
Transformer layers	3
Attention heads	4
Feed-forward dimension	64
Classifier hidden dimension	32
Dropout	0.2
Training epochs	50
Batch size	1024
Learning Rate	1×10^{-4}
Random state	10
K_n	{1, 2, 4, 16}
K_a	{1, 2, 4, 16}

Table 2: Values for neural model hyperparameter and dataset configuration parameters.

parameters to analyze their effect on sequence discrimination and morphological disambiguation performance (see Table 2).

7.3 Evaluation Metrics

The model is evaluated using both sequence-level and token-level metrics. Sequence-level performance is measured using ranking accuracy, which evaluates whether the gold morphological sequence receives a higher grammaticality score than all corresponding negative candidate sequences for a sentence.

Let S_g denote the gold morphological sequence for a sentence, and let $\{S_1^-, S_2^-, \ldots, S_n^-\}$ denote the corresponding negative candidate sequences. If $f(\cdot)$ represents the grammaticality scoring function learned by the model, then a prediction is considered correct when $f(S_g) > f(S_i^-), \quad \forall i \in \{1, 2, \ldots, n\}$. The sequence-level ranking accuracy is computed as:

$$\text{Ranking Acc.} = \frac{1}{N} \sum_{k=1}^{N} \mathbb{I}\left(f(S_g^{(k)}) > \max_i f(S_i^{-(k)}) \right)$$

where N denotes the total number of evaluation sentences and $\mathbb{I}(\cdot)$ is the indicator function.

Additionally, word-level disambiguation accuracy is computed by selecting, for each token, the candidate analysis associated with the highest-scoring sentence sequence for the i-th sentence. Let $a_{i,j}$ denote the gold analysis of the j-th token in the i-th sentence, and $\hat{a}_{i,j}$ denote the predicted analysis of that same token within the top-scoring sequence. The token-level accuracy across a corpus of N sentences is defined as

$$\text{Word Acc.} = \frac{1}{\sum_{i=1}^{N} L_i} \sum_{i=1}^{N} \sum_{j=1}^{L_i} \mathbb{I}(\hat{a}_{i,j} = a_{i,j})$$

We evaluate this metric across two distinct granularities depending on the selection of L_i:

- **Ambiguous Word Accuracy:** L_i denotes the number of ambiguous tokens within the i-th sentence. This captures the effectiveness of the proposed framework in resolving contextual morphological ambiguity at the token level.
- **Overall Word Accuracy:** L_i denotes the total number of all valid, non-padded tokens within the i-th sentence. This metric incorporates the deterministic performance of the rule-based components alongside the neural ranking framework, capturing the end-to-end pipeline accuracy.

K_n	K_a	Rank Acc. (%)	Overall Word (%)	Ambigious Word (%)	Resolved (nos./579)
1	1	13.39	80.97	63.21	366
1	2	27.68	86.15	73.23	424
1	4	19.64	83.11	67.36	390
1	16	33.93	88.20	77.20	447
2	1	28.57	87.22	75.30	436
2	2	23.21	82.84	66.84	387
2	4	37.50	89.63	79.97	463
2	16	29.46	84.90	70.81	410
4	1	13.39	76.85	55.27	320
4	2	19.64	81.86	64.94	376
4	4	35.71	88.29	77.37	448
4	16	34.82	85.61	72.19	418
16	1	16.07	77.12	55.79	323
16	2	23.21	82.22	65.63	380
16	4	36.61	88.47	77.72	450
16	**16**	**44.64**	**91.42**	**83.42**	**483**

Table 3: Effect of negative sampling and random-offset augmentation on dataset scale, sequence ranking, and morphological disambiguation performance with **model dimension = 32**.

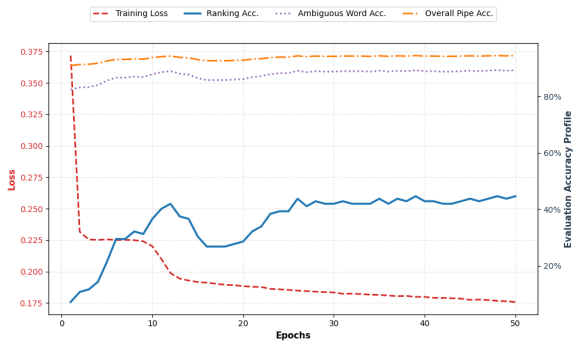

Figure 9: Comparison of direct model (model dim = 32) loss and other metrics during model training with limited negative sampling $K_n = 16$ and augmentation factor $K_a = 16$.

The model is tested for word accuracy during the training process, which is not representative of the inference-time performance because, the training is conducted on a limited number of negative samples that contain one wrong analysis, thereby only attracting a small penalty for incorrect ranking. But during inference all combinations of analysis sequence are ranked which increases the possibility of incorrect prediction for multiple tokens.

8 Results and Discussion

8.1 Quantitative Results

As the model improves on the discriminative sequence classification objective, its ability to rank correct morphological sequences above incorrect alternatives also improves (Figure 9). Consequently, improvements in ranking accuracy lead to corresponding improvements in token-level morphological disambiguation accuracy. The results indicate that sentence-level contextual modeling substantially resolves morphological ambiguity, particularly in cases involving multifunctional morphemes and structurally similar candidate analyses.

The effect of augmentation and negative sampling on the model's performance on the test set has been illustrated in Table 3. Performance seems to improve uniformly with increasing values of K_n and K_a. Moreover, the loss convergence is more stable for higher values of $K_n \times K_a$. It was observed that the model may not produce the same accuracy on every run for a particular value of K_n and K_a. Therefore, we took the best out of 3 training runs for each row in the Table 3.

Overall, the model with projection dimension 32 produced better results than the model with 64 dimensional projection. This can be attributed to appropriate compression of morphological information required for scoring the grammar of a sequence.

8.2 Error Analysis

On investigating the system outputs on the test set, it is observed that the system performs better in sentences with fewer combination of analysis sequence. However, there are cases where due to a single misclassifications in the sequence the ranking accuracy has gone down. Some mistakes, committed by the best performing model are given in Figure 10.

It is important to note here that the model only sees morphological vectors, which only depicts the structure of the sentences. It does not capture the words, characters or any other surface realization, and is blind to word and sentence semantics. Therefore, the model is not able to correctly classify detailed PoS tags in the provided examples.

8.3 Discussion

This work does not expect to establish a linguistic theory for Assamese morphology. It only tries to establish a framework for realizing automatic morphological analysis, for which we assumed that there exists a *morphological grammar* across Assamese sentences. The results of the experiments suggests that the assumption might indeed be correct. The following characteristics of our experiment support this claim:

- The contextual encoder is blind to semantics and only takes the structure of words as input. Even then, the ranking accuracy improved

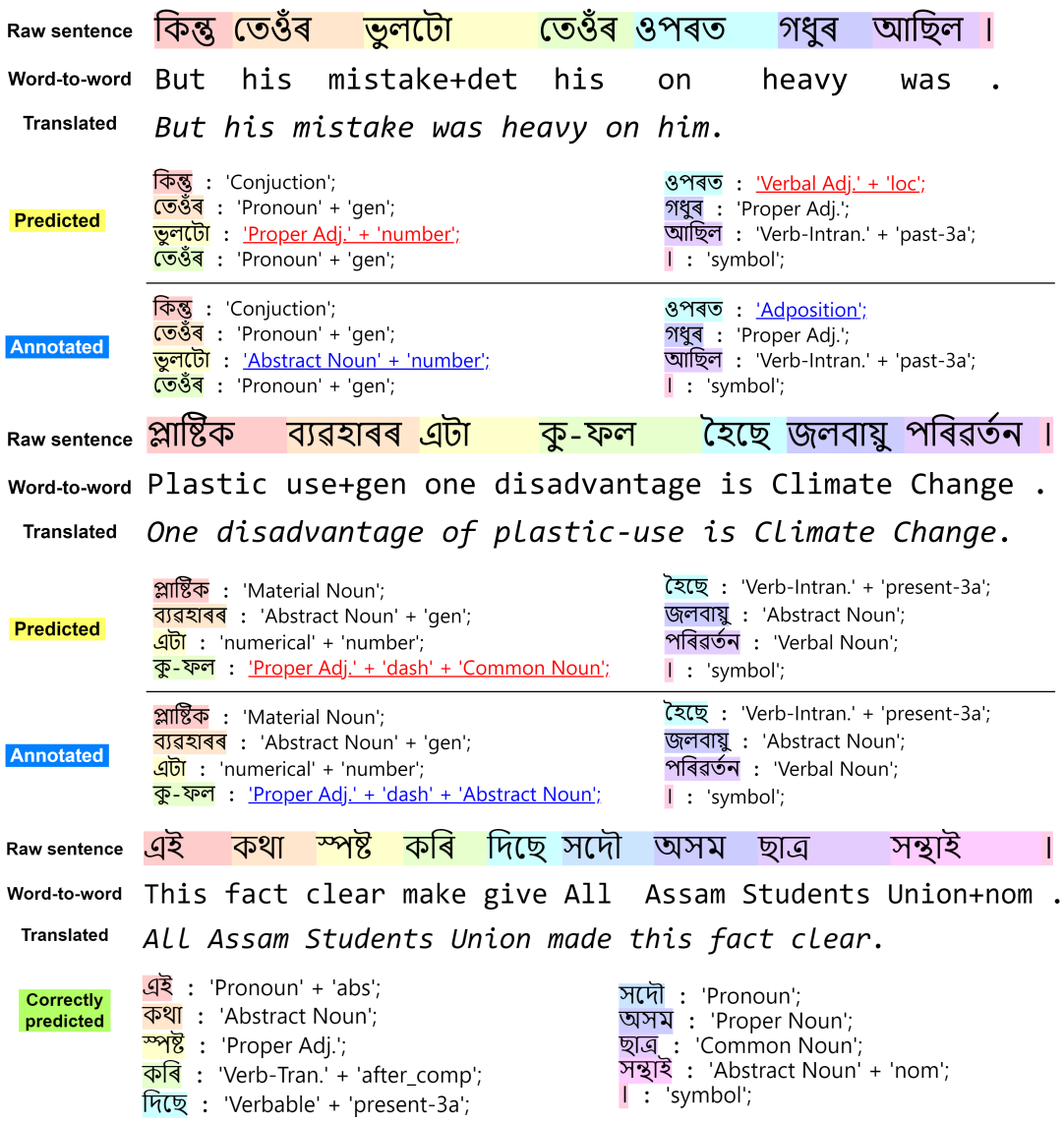

Figure 10: Difference in PoS-tagging of predicted and annotated values. The first two illustrations depict errors and the bottom one depicts correct prediction.

while the model learned to discriminate between correct and incorrect samples.

- The performance improves with increasing number of augmented samples (see Table 3). This aligns with the interpretation that Assamese sentences exhibit free word-group ordering and augmentation with random offset helps the model generalize better.

The implication of the hypothesis being correct lies in promotion of NLP research in low-resource agglutinative languages. For example, one might suggest that if there really exist such a grammar then a deterministic modeling will perform better for most instances.

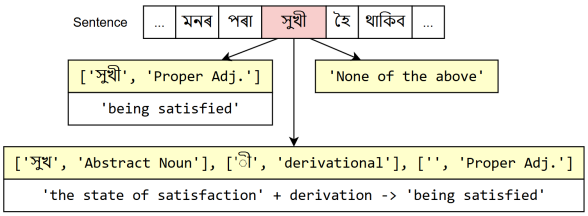

Figure 11: Both the options ending with 'Proper Adj.' are correct.

Limitations

The performance of the proposed system depends on several linguistic and methodological factors. Since the analyzer relies on a predefined morpheme inventory, handling out-of-vocabulary (OOV) and foreign words remains challenging. Nevertheless,

the system assigns a dedicated feature representation to words with no valid morphological analysis. During annotation, multiple candidate analyses were occasionally found to be linguistically plausible for the same token (see Figure 11), highlighting the importance of linguistic expertise in resource creation and annotation.

The bigram transition rule-set currently captures only the observed and linguistically validated transitions. Some missing transitions were identified during dataset annotation, indicating scope for further expansion. Additionally, to reduce ambiguity, derivational transformations were restricted to the most probable cases.

The current framework does not explicitly model semantic information, although semantic interpretation may influence suffix functionality in certain contexts. The feature vector constructor in our system is rather simple. More sophisticated feature construction methods and additional linguistic attributes, such as animacy, may further improve performance.

The number of valid morphological sequences can grow rapidly for morphologically complex sentences and OOV words. Although chunk-based ranking was used to handle large candidate sets efficiently, further constraints on candidate generation and heuristic methods may improve scalability.

Ethics Statement

Acknowledgements

References

Manmee Bhattacharjya. 2012. Case marking in Assamese: Morpho-Syntactic and Semantic Analysis. In Abhijit Debnath, Tariq Khan, Amalesh Gope, and Bindu Madhavi, editors, *Coalescence: A Post Conference Publication cum Students' Resource Book*, pages 195–204. Mudranik Technologies Pvt. Ltd., Bangalore.

Palash Das, Madhumita Barbora, and Utpal Sharma. 2019. Morphological rule formulation for nouns in assamese to develop morphological tools. *Language in India*, 19.

Priyanka Dasari, Abhijith Chelpuri, Nagaraju Vuppala, Mounika Marreddy, Parameshwari Krishnamurthy, and Radhika Mamidi. 2023. Transformer-based context aware morphological analyzer for Telugu. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 25–32, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Pinky Moni Gayan and Arup Kumar Nath. 2025. Productivity of assamese derivational suffixes. *Language Sciences*, 112:101746.

Jugalakshmi Gogoi. 2020. Word formation in Assamese. Master's thesis, Assam University, Silchar, India. Department of Linguistics.

Gulshat Kessikbayeva and Ilyas Cicekli. 2014. Rule based morphological analyzer of Kazakh language. In *Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM*, pages 46–54, Baltimore, Maryland. Association for Computational Linguistics.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad B, Varun G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh M. Khapra. 2024. IndicLLMSuite: A blueprint for creating pre-training and fine-tuning datasets for Indian languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15831–15879, Bangkok, Thailand. Association for Computational Linguistics.

Bipul Roy and Bipul Syam Purkayastha. 2017. A suffix based morphological analysis of assamese word formation. *International Journal on Recent and Innovation Trends in Computing and Communication*, 5(3):445–449.

Navanath Saharia. 2014. *Computational Morphology and Syntax for a Resource-Poor Inflectional Language*. Ph.D. thesis, Tezpur University, Napaam, Tezpur, Assam, India. Accessed via Shodhganga: A reservoir of Indian theses.

Pragyanur Saikia, Arindam Karmakar, and Utpal Sharma. 2025. Issues and challenges in training vector representation for assamese words. In *2025 IEEE Guwahati Subsection Conference (GCON)*, pages 1–6.

Wolfgang Seeker and Jonas Kuhn. 2013. Morphological and syntactic case in statistical dependency parsing. *Computational Linguistics*, 39(1):23–55.

Utpal Sharma, Jugal K. Kalita, and Rajib K. Das. 2008. Acquisition of morphology of an indic language from text corpus. *ACM Transactions on Asian Language Information Processing*, 7(3).

Qinlan Shen, Daniel Clothiaux, Emily Tagtow, Patrick Littell, and Chris Dyer. 2016. The role of context in neural morphological disambiguation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 181–191, Osaka, Japan. The COLING 2016 Organizing Committee.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

XOBDO. 2026. Xobdo: An online multilingual dictionary for north-east indian languages. Accessed: 2026-05-24.

Eray Yildiz, Caglar Tirkaz, H. Sahin, Mustafa Eren, and Omer Sonmez. 2016. A morphology-aware network for morphological disambiguation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

A Tags and Features: Details

A.1 List of Morpheme Tags

Following is the comprehensive list of all the morpheme tags used in the system. All morphemes in Dictionary-A belong to any of these categories. Special symbols, word-boundary tags and numeric-strings do not belong to this dictionary. The following classification of the tags is only done for the purpose of readability.

PoS tags: 'Abstract Noun', 'Common Noun', 'Material Noun', 'Verbal Noun', 'Noun', 'Proper Noun', 'Pronoun', 'Adverb', 'Verbable', 'neg_v_verb', 'Verb-Trans.', 'Verb-Intran.', 'Adjective', 'Adjective Adj.', 'Proper Adj.', 'Verbal Adj.', 'Conjuction', 'Interjection', 'Adposition', 'Others'

Suffix tags: 'derivational', 'abs', 'nom', 'gen', 'acc', 'dat', 'loc', 'inst', 'number', 'modal', 'adj_comp', 'after_comp', 'reason_comp', 'when_comp', 'ardha', 'infix', 'dash', 'manner', 'mod_attr', 'quantity', 'compare', 'emph', 'NA'

Verb-tense tags: 'present-2i', 'present-3a', 'present-2n', 'present-2f', 'present-1', 'past-3a', 'past-2f', 'past-2a', 'past-1', 'future-2f', 'future-3a', 'future-1'

Preffix tags: 'numerical', 'without', 'other', 'time', 'negative'

Word-boundary: 'ws', 'we'

A.2 Feature Vector Details

The features contain super-set PoS tags (e.g. Noun is a super-set for {Abstract Noun, Common Noun, Proper Noun}.) For any PoS tag, the weight of its corresponding super-set is also increased. The symbols are further categorized into – 'end_sym', 'pause_sym' and 'other_sym'. String of digits are separated from words during preprocessing, and marked using 'numeric-string'. Honorifics are marked using – 'a' for 'any', 'i' for 'informal', 'n' for neutral, and 'f' for formal. These can be extracted from the verb-tense tags and integrated pronouns. 'p1', 'p2', and 'p3' are used to mark person.

Morphological feature set = { 'Abstract Noun', 'Adjective', 'Adjective Adj.', 'Adposition', 'Adverb', 'Common Noun', 'Conjuction', 'Conjunction', 'Interjection', 'Material Noun', 'Noun', 'Others', 'Pronoun', 'Proper Adj.', 'Proper Noun', 'Verb', 'Verb-Intran.', 'Verb-Trans.', 'Verbable', 'Verbal Adj.', 'Verbal Noun', 'a', 'abs', 'acc', 'adj_comp', 'after_comp', 'ardha', 'compare', 'dash', 'dat', 'derivational', 'emph', 'end_sym', 'f', 'future', 'gen', 'i', 'infix', 'inst', 'loc', 'manner', 'mod_attr', 'modal', 'n', 'neg_v_verb', 'negative', 'nom', 'numeric-string', 'numerical', 'other', 'other_sym', 'p1', 'p2', 'p3', 'past', 'pause_sym', 'pl', 'present', 'quantity', 'reason_comp', 'sg', 'time', 'when_comp', 'without' }

B Incorporation of Morphological Characteristics of Assamese

We incorporated multiple characteristics of the language into our methodology and algorithm. We have only considered the major phenomenon of the language and avoided the exceptions.

B.1 Morphophonological Alternation

Morphophonological alternation produces multiple surface realization of morphemes during affixation. To incorporate some of these characteristics we have categorized all the surface forms of the affixes into the same category of their role. However, we only address a few forms of base that undergo morphophonological alternation. These are – 1) the affixation phenomenon in verbs and 2) number-words (first 9 natural numbers) as numerical prefixes.

With verbs, prefixation by 'negation' affix and some kind of suffixation cause alterations. Due to this we created two categories:

1. **'neg_v_verb':** These are verbs whose starting *vowel-character* is removed to facilitate affixation using the 'negation' prefix.

2. **'Verbable':** These are verbs whose ending *vowel-symbol* is removed. This facilitates the construction of 'causative stems' (Gogoi, 2020) that attracts verb-like suffixation.

Similarly, to accommodate another major phenomenon, the vowel ending from the number-

words are subtracted to create a class of numeric prefixes.

B.2 Bigram Transition Rules

The bigram transition rules has been provided in the supplementary material along with the description of each tags.

B.3 Derivational Probabilities

The tag transition probabilities for each of the derivational suffix has been provided in the supplementary material.

B.4 Integrated Tags

The suffixes under the tag 'number' – also called as 'definitives' in Assamese (Sharma et al., 2008; Das et al., 2019) – can either be singular or plural. We have used a mapping for all such suffixes which is provided as a supplementary material.

Similarly, we maintain a dictionary for the tags associated with all pronouns that are listed in 'XOBDO' (XOBDO, 2026). We also maintain a list for the verb-suffixes that convey tense, person and honorific information. E.g. 'present-1', 'past-2f', etc. To extract the integrated features we use regular expressions.

C Experiment Setup: Details

The preprocessing of the 'Sangraha' corpus was carried out using the methodology mentioned by Saikia et al. (Saikia et al., 2025). The system works better with preprocessed sentences. However, all the resource available in the supplementary materials already preprocessed. Details of the other half of the experiments are shown in Table 4. Additional hardware and neural model details is provided in Table 5.

K_n	K_a	Rank Acc. (%)	Overall Word (%)	Ambigious Word (%)	Resolved (nos./579)
1	1	26.79	83.91	68.91	399
1	2	28.57	86.15	73.23	424
1	4	21.43	83.91	68.91	399
1	16	24.11	86.15	73.23	424
2	1	16.96	80.70	62.69	363
2	2	26.79	86.15	73.23	424
2	4	**39.29**	89.45	79.62	461
2	16	33.93	88.38	77.55	449
4	1	21.43	84.09	69.26	401
4	2	20.54	83.38	67.88	393
4	4	33.04	88.74	78.24	453
4	16	29.46	86.68	74.27	430
16	1	16.96	79.89	61.14	354
16	2	38.39	88.47	77.72	450
16	4	29.46	86.86	74.61	432
16	**16**	**39.29**	**89.63**	**79.97**	**463**

Table 4: Effect of negative sampling and random-offset augmentation on dataset scale, sequence ranking, and morphological disambiguation performance with model dimension = 64.

Component	Value
Framework	PyTorch
Programming Language	Python 3.13
CUDA Used	Yes
Precision	float32
Optimizer	Adam
Loss Function	Binary Cross Entropy Loss
Inference Strategy	Highest ranked analysis sequence
Total Parameters (32 dim)	30,081
Approximate Size (32 dim)	0.1147 MB
Total Parameters (64 dim)	84,481
Approximate Size (64 dim)	0.3223 MB

Table 5: Other details regarding experimental setup.

D Algorithms

Algorithm 1 OPTIONSOFSENTENCETOKENS

Input: Sentence s, special character set $\mathcal{C}$
Output: Sequence of token-level morphological option sets $\mathcal{O}$

Initialize $\mathcal{O} \leftarrow [\,]$
foreach *token* w *in* s **do**
  $opts \leftarrow$ RULEBASEDFORWARDANALYSIS(w)
  if $opts \neq \emptyset$ **then**
    Append $opts$ to $\mathcal{O}$
  else if $w \in \mathcal{C}$ **then**
    Append [[[w, `'symbol'`]]] to $\mathcal{O}$
  else if w *matches numeric pattern* **then**
    Append [[[w, `'numeric-string'`]]] to $\mathcal{O}$
  else
    Append [[[w, `'Others'`]]] to $\mathcal{O}$
return $\mathcal{O}$

Algorithm 2 RULEBASEDFORWARDANALYSIS

Input: Input word w, morpheme lexicon $\mathcal{M}$, morphotactic transition matrix $\mathcal{R}$

Output: Set of valid analyses $\mathcal{A}$

Initialize $\mathcal{A} \leftarrow \emptyset$

```
Analyze(r, t, s)(
if r = ϵ then
    if R(t, we) > 0 then
        Add s to A
    return
T = {t' | R(t, t') > 0 ∧ t' ≠ we}
for i ← 1 to |r| do
    m ← r[1 : i]
    u ← r[i + 1 : |r|]
    foreach t' ∈ T do
        if m ∈ M(t') then
            s' ← s ∪ {(m, t')}
            if t' = 'derivational' then
                Determine derived category d
                s' ← s' ∪ {(ϵ, d)}
                Analyze(u, d, s')
            else if t' = 'pronoun' then
                Determine integrated case p
                if p ≠ 'pronoun' then
                    s' ← s' ∪ {(ϵ, p)}
                    Analyze(u, p, s')
                else
                    Analyze(u, t', s')
            else
                Analyze(u, t', s')
)
Analyze(w, 'ws', ∅)
return A
```

Algorithm 3 FEATUREVECTORCONSTRUCTOR

Input: Sequence of (morpheme, tag) tuple $(m, t) \in \mathcal{S}$, detail map of number tag $\mathcal{N}_{detail}$, integrated tags in pronouns, integrated tags in verb suffixes $\mathcal{V}_{suffixes}$, super set mapping of PoS $\mathcal{P}_{map}$, initial feature state $\mathcal{F} \leftarrow \mathcal{F}_{default}$, and tracking counter $pos_num \leftarrow 0$.

Output: Populated morphological feature vector $vector$

```
foreach (m, t) ∈ S (right-to-left) do
    if t = 'number' then
        F[N_detail[m]] ← F[N_detail[m]] + 1
    else if t = 'Pronoun' then
        T ← person, honorific tags of m
        foreach valid f ∈ T do
            F[f] ← F[f] + 1
    else if t = 'symbol' then
        F[S_map[m]] ← F[S_map[m]] + 1
    else if t = 'derivational' then
        F[t] ← 1
    else if t = 'adj_comp' then
        F[t] ← F[t] + 1
        F['Adjective'] ← F['Adjective'] + 0.25
    else if t ∈ P_map then
        w ← 1 − 0.25 × pos_num
        if F[t] = 0 then
            F[t] ← w
            pos_num ← pos_num + 1
        if F[P_map[t]] = 0 then
            F[P_map[t]] ← w
            pos_num ← pos_num + 1
    else if t ∈ V_suffixes then
        T ← person, honorific, tense tags of m
        foreach valid f ∈ T do
            F[f] ← F[f] + 1
    else
        F[t] ← F[t] + 1
vector ← values of F sorted alphabetically by key
return vector
```