# FastPOS: Language-Agnostic Scalable POS Tagging Framework Low-Resource Use Case

Anonymous ACL submission

## Abstract

This study proposes a language-agnostic transformer-based POS tagging framework designed for low-resource languages, with Bangla and Hindi serving as case studies. With only three lines of framework-related code, the framework was adapted to a new language, namely, from Bangla to Hindi. Displaying its effectiveness with minimal code change. Additionally, the framework achieves 96.85% and 97% token-level accuracy across POS categories, maintaining robust F1 scores despite dataset imbalance and linguistic overlaps in Bangla and Hindi, respectively. However, the performance discrepancy in a specific POS type highlights challenges in dataset curation. Moreover, the performance is due to the transformer used under the hood of this framework, which can be swapped with minimal code changes. The framework's modular, language-agnostic design and open-source design enable rapid adaptation with minimal code modification. By reducing model design and tuning overhead, researchers can prioritize linguistic preprocessing and dataset refinement, key tasks in advancing NLP for underrepresented languages.

## 1 Introduction

Parts of speech (POS) tagging is a core NLP task essential for higher-level applications such as parsing, sentiment analysis, machine translation, and many more. However, building an effective POS tagger for low-resource languages (LRLs) remains challenging due to the limited availability of annotated data, a lack of standardization, and high morphological complexity.

While language-specific models exist, they are hard to scale across diverse LRLs. This study introduces a novel language-agnostic, transformer-based POS tagging framework engineered explicitly for rapid adaptability with minimal code modification. Additionally, the framework supports modular integration and dataset flexibility across languages. We use Bangla and Hindi as a case study to demonstrate their effectiveness, achieving strong token-level accuracy despite class imbalance and linguistic overlaps, and also propose a standardized dataset for future investigations.

## 2 Background

The severe lack of high-quality linguistic resources hinders the development of practical NLP tools. In LRLs, it is one of the primary issues. POS tagging tools are not too different. Significant progress has been achieved for high-resource languages, but for LRLs, it remains a formidable challenge. Data scarcity is prevalent and often limited to only thousands of tokens, a stark contrast to the millions available for high-resource languages (McGiff and Nikolov, 2025). This limitation significantly constrains the ability of neural models to learn robust generalizations. Furthermore, many LRLs, including Bangla and Hindi, exhibit complex morphological structures, which leads to a high out-of-vocabulary problem (Alam et al., 2017). Additionally, the available datasets suffer from annotation inconsistencies, noise, and a lack of standardization (Kim et al., 2015). As a result, special processing steps require a considerable amount of time. Furthermore, finetuning of transformers usually offers strong performance in language-related workflows (Devlin et al., 2019; Conneau et al., 2020; Liu et al., 2019). Language-specific models, such as BanglaBERT, can capture language-specific features (Bhattacharjee et al., 2022). Moreover, tokinization plays an essential role in transformer performance due to the morphisms, and enriching supervision with sentence type helps models capture semantic nuance, benefiting tasks like question answering and dialogue systems (Dang et al., 2024; Majumdar et al., 2022). However, most existing transformer-based systems are not plug-and-play. They require substantial to-

kenization decisions and model engineering for each new language. This creates a barrier for scalable applications across LRLs. Researchers dedicate time and effort to maintaining the code base, where the primary focus should be on preprocessing, dataset standardization, and other related tasks. Recent benchmarking efforts (e.g., Indic-Transformers) confirm that transformer models, when properly fine-tuned, can achieve high F1 scores (e.g., 92.60 for Bengali) (Sarker, 2021). Yet, the ease of transferring these models across languages, as well as the use of modular frameworks that abstract underlying complexity, remains underexplored. To bridge this gap, a language-agnostic POS tagging framework that can adapt to new languages with minimal changes is essential. Such a system would support scalable deployment, rapid experimentation, and reduced engineering overhead, particularly critical in under-resourced environments. A modular design enables interchangeable transformer backbones, allowing researchers to experiment with different architectures (e.g., ByT5 for character-level modeling (Dang et al., 2024)) without rewriting core logic. Minimal code changes ensure quick language adaptation, facilitating cross-lingual experimentation and comparative studies. This study proposes a language-agnostic, open-source POS tagging framework that enables high performance with minimal code modification. Such a system not only facilitates faster adaptation to new languages, evidenced by successful Bangla-to-Hindi transfer, but also shifts the focus towards linguistic preprocessing and dataset refinement, where most of the performance bottlenecks now reside.

## 3 Methodology

This study primarily utilizes a Bangla dataset (2696 instances across five main POS categories in Bangla) for Part-of-Speech (POS) tagging, complemented by a collected Hindi dataset (14963 instances across sixteen POS categories) to evaluate the framework's adaptability (hin). Bangla dataset comprises 2,696 sentences (simple, complex, and compound), meticulously annotated with five primary POS types. Figure 1 details these POS types, including English interpretations, which will be referred to as "POS type #" hereafter. Figure 2 illustrates the imbalanced POS distribution inherent in the Bangla dataset, a common characteristic in real-world linguistic data, where POS types 1 and

POS Information

| Bengali Term | Label | Pronunciation | English Meaning | Part of Speech Type |
|---|---|---|---|---|
| বিশেষ্য | 1 | *bishesho* | Noun | Name of a person, place, or thing |
| বিশেষণ | 2 | *bisheshoṇ* | Adjective | Describes a noun |
| সর্বনাম | 3 | *shorbonam* | Pronoun | Replaces a noun |
| ক্রিয়া | 4 | *kriya* | Verb | Action or state of being |
| অব্যয় | 5 | *obyoi* | Indeclinable / Particle | Words that don't change form (e.g., connectors, adverbs, interjections) |

Figure 1: Part of Speech Information (Bangla)

4 collectively represent over half of the instances, similar to English. Figure 3 illustrates an example

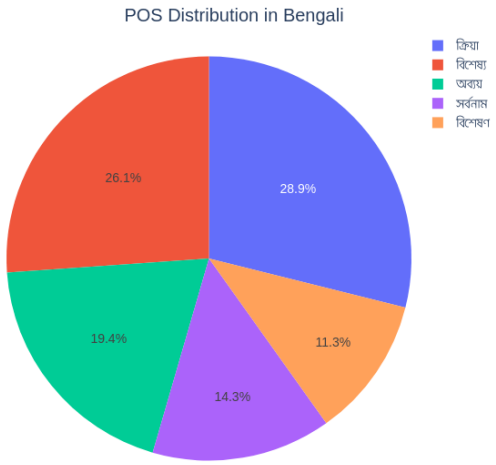


Figure 2: Parts of Speech Distribution (Bangla)

of how this collected Hindi dataset was modified to align POS tags with words, fulfilling the framework's input requirements. The specific code used for this adaptability test is accessible on (hin)

| Words | POS |
|---|---|
| ['लंदन', 'से', 'प्रकाशित', '"', 'दि', 'संडे', ... | ['NNP', 'PSP', 'JJ', 'SYM', 'NNP', 'NNP', 'NNP'... |
| ['शोम', 'ने', 'कहा', 'कि', 'एफबीटी', 'के', 'तहत', ... | ['NNP', 'PSP', 'VM', 'CC', 'NNP', 'PSP', 'PSP'... |
| ['अमेरिकी', 'सरकार', 'ने', 'यह', 'पाबंदी', 'चुश', ... | ['NNP', 'NNP', 'PSP', 'PRP', 'NN', 'NNP', 'PSP'... |
| ['इसके', 'महेनजर', 'संशोधित', 'तपेदिक', 'नियम', ... | ['PRP', 'PSP', 'JJ', 'NNP', 'NNP', 'NNP', 'PSP'... |
| ['अधिसूचना', 'के', 'मुताबिक', '१८', 'मीटर', 'वाले', ... | ['NN', 'PSP', 'PSP', 'QC', 'NN', 'PSP', 'NN', ... |

Figure 3: Example of Modified Hindi Dataset

### 3.1 Framework:

This POS tagging framework is designed for low-resource languages, leveraging the transformer model in its core. It supports custom datasets, training, prediction, and model management using PyTorch, HuggingFace, and Scikit-learn. Its modular

and extensible architecture ensures easy integration into broader NLP pipelines with minimal overhead.

#### 3.1.1 UML Diagram:

The following figure 4 illustrates the UML Diagram, which shows the architecture of the proposed POS tagging framework for low-resource languages, highlighting the interaction between core components that enable model training, prediction, and deployment.SentenceClassifier class details are not shared, as that is irrelevant to this study, and this framework is published as a class or module of the LowResNLTK framework (ano).

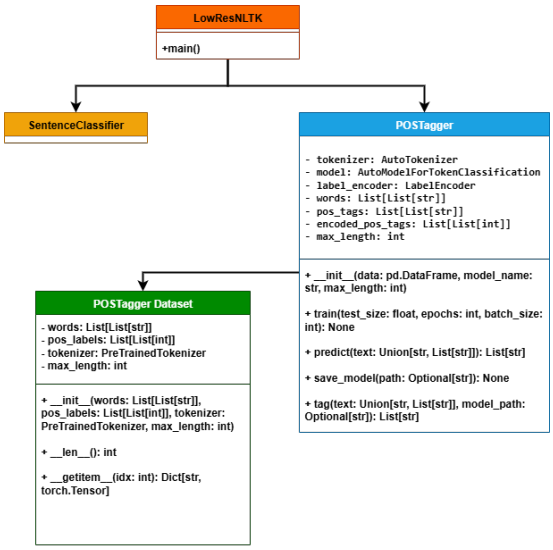


Figure 4: UML Diagram

#### 3.1.2 Purpose:

- Designed for Part-of-Speech (POS) tagging, especially for low-resource languages.
- Utilizes transformer-based models, giving it freedom to choose any HuggingFace transformer.

#### 3.1.3 Key Features:

- Data Handling: Accepts data in Pandas DataFrame format with words and POS tags.
- Custom Dataset: Implements a PyTorch Dataset for tokenization and label alignment.
- Label Encoding: Uses sklearn's LabelEncoder for mapping POS tags to numerical labels.
- Model Training: Supports training with HuggingFace's Trainer and TrainingArguments.
- Prediction: Provides methods to predict POS tags for new text inputs.
- Model Saving/Loading: Can save and reload trained models and label encoders for reuse.
- Pretrained Model Support: Can load and use pretrained transformer models.

#### 3.1.4 Extensibility:

Easily adaptable to other transformer models and languages, and any number of POS tags.Modular design allows for integration into larger NLP pipelines.

#### 3.1.5 Usages:

- Train a POS tagger on a custom dataset easily.
- Predict POS tags for new sentences.
- Save and Load model for deployment.

#### 3.1.6 Integration:

Built on popular libraries like PyTorch, HuggingFace, scikit-learn, and pandas. Can be used as a standalone tool or easily integrated into other Python and NLP projects. It is open-source and easily modifiable to meet one's specific needs.

## 4 Experiments

This study used Bangla and Hindi as use cases to demonstrate the working principle. Here, we developed the system with the help of transformers, specifically "BanglaBERT" for Bangla and **Model name** with a token classification head, and an 80%-20% train-test split for each case study.

Table 1: Classification report with token-level accuracy

| Class | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| 1 | 0.97 | 0.97 | 0.97 | 1265 |
| 2 | 0.97 | 0.93 | 0.95 | 513 |
| 3 | 0.97 | 0.97 | 0.97 | 622 |
| 4 | 0.98 | 0.98 | 0.98 | 1376 |
| 5 | 0.96 | 0.97 | 0.97 | 831 |
| **Accuracy** | | | 96.85 | 4607 |
| **Macro Avg** | 0.97 | 0.96 | 0.97 | 4607 |
| **Weighted Avg** | 0.97 | 0.97 | 0.97 | 4607 |

**Token-level accuracy:** 0.9685

Table 1 and Figure 5 show that the token-level accuracy is 96.85%, indicating reliable performance at the individual token level, which is particularly important in POS tagging. Focusing on the performance in detail reveals that the model performed well across all POS types, except for types two and five. For POS type two, low recall shows that the model is struggling to identify this type of POS.

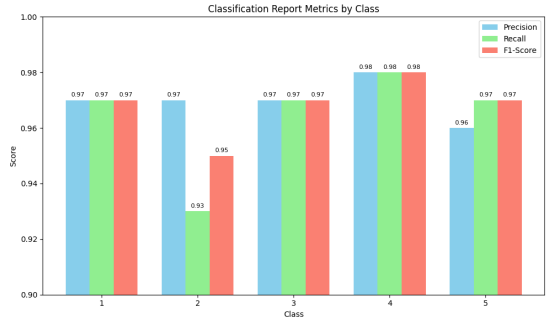


Figure 5: Confusion Matrix Results Bangla

| Category | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| CC | 0.71 | 0.50 | 0.59 | 10 |
| INTF | 1.00 | 1.00 | 1.00 | 1 |
| JJ | 0.25 | 0.23 | 0.24 | 26 |
| NEG | 1.00 | 1.00 | 1.00 | 4 |
| NN | 0.67 | 0.72 | 0.69 | 100 |
| NNP | 0.71 | 0.63 | 0.67 | 38 |
| NST | 0.00 | 0.00 | 0.00 | 1 |
| PRP | 0.90 | 0.75 | 0.82 | 12 |
| PSP | 0.93 | 0.85 | 0.89 | 47 |
| QC | 1.00 | 0.40 | 0.57 | 5 |
| QF | 1.00 | 1.00 | 1.00 | 1 |
| RB | 0.50 | 1.00 | 0.67 | 1 |
| RP | 0.33 | 1.00 | 0.50 | 1 |
| SYM | 1.00 | 1.00 | 1.00 | 2666 |
| VAUX | 0.83 | 0.88 | 0.86 | 34 |
| VM | 0.73 | 0.83 | 0.78 | 46 |
| **Accuracy** | | | 0.97 | 2993 |
| **Macro Avg** | 0.72 | 0.74 | 0.70 | 2993 |
| **Weighted Avg** | 0.97 | 0.97 | 0.97 | 2993 |

Table 2: Performance metrics (Precision, Recall, F1-Score, and Support) per category on the Hindi dataset.

In Table 2 and Figure 3, despite achieving an impressive 97% accuracy, the proposed framework, specifically the transformer, still faces the same issue as seen in Bangla. The performance analysis

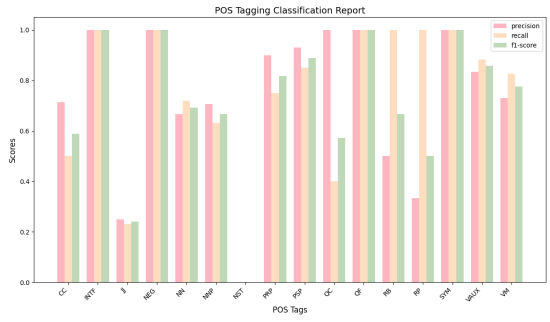


Figure 6: Confusion Matrix Results Hindi

across both Bangla (Table 1, Figure 5) and Hindi (Table 2, Figure 6) consistently reveals a critical limitation of the proposed transformer-based framework: its struggle with data sparsity and highly imbalanced classes. Despite achieving high overall accuracies (e.g., 96.85% token-level accuracy in Bangla, 97% overall accuracy in Hindi), the models exhibit apparent weaknesses in low-resource categories. For instance, Bangla's Class 2 exhibited lower recall due to fewer instances, while Hindi's 'NST' and 'RP' categories (with minimal support) yielded 0.00 F1-scores, and 'JJ' performed poorly. These discrepancies highlight that even advanced transformer models face significant hurdles with subtle grammatical distinctions and rare categories, often leading to under-prediction or false positives. This suggests that the performance limitations are not a flaw in the framework's design, but rather stem from the transformer's inherent inductive biases or insufficient pre-training on the specific linguistic nuances of such low-resource data.

## 5 Conclusion

This study presents a language-agnostic transformer-based POS tagging framework, using Bangla as a compelling case study. Data imbalance and Linguistic challenges, considering the model, yield an impressive accuracy of 96.85% and 97%, demonstrating a consistently high F1 Score across the five main POS categories. Performance discrepancies observed in POS types across both Bangla and Hindi highlight the limitations of current transformer architectures when confronted with the severe data sparsity and subtle linguistic overlaps prevalent in low-resource settings. With its language-agnostic architecture and minimal code dependency, the framework offers a scalable solution for morphologically rich, underrepresented languages in NLP.

## 6 Limitations

Despite its strong overall accuracy and adaptability for low-resource POS tagging, the proposed transformer-based framework has two key limitations. First, data imbalance causes underpredictions. Second, grammatical overlap with similar constructs leads to false positives. These challenges underscore the necessity for advanced data-centric approaches, such as multitask or few-shot learning, to address the inherent linguistic asymmetries and data sparsity in low-resource languages. We also plan to improve user adoption by aligning the framework's API with the scikit-learn interface.

## References

Hindi pos tagging test.

lowresnltk.

Firoj Alam, Shammur Absar Chowdhury, and Sheak Rashed Haider Noori. 2017. Bidirectional lstms - crfs networks for bangla pos tagging. *19th International Conference on Computer and Information Technology, ICCIT 2016*, pages 377–382.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Kazi Samin, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. Banglabert: Language model pretraining and benchmarks for low-resource language understanding evaluation in bangla. *Findings of the Association for Computational Linguistics: NAACL 2022 - Findings*, pages 1318–1327.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Thao Anh Dang, Limor Raviv, and Lukas Galke. 2024. Tokenization and morphology in multilingual language models: A comparative analysis of mt5 and byt5.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova Google, and A I Language. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North*, pages 4171–4186.

Young-Bum Kim, Benjamin Snyder, and Ruhi Sarikaya. 2015. Part-of-speech taggers for low-resource languages using cca features. pages 17–21.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov, and Paul G Allen. 2019. Roberta: A robustly optimized bert pretraining approach.

Pritha Majumdar, Deepak Alok, Akanksha Bansal, Atul Kr. Ojha, and John P. McCrae. 2022. Bengali and Magahi PUD treebank and parser. In *Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, pages 60–67, Marseille, France. European Language Resources Association.

Josh McGiff and Nikola S. Nikolov. 2025. Overcoming data scarcity in generative language modelling for low-resource languages: A systematic review.

Sagor Sarker. 2021. Bnlp: Natural language processing toolkit for bengali language.