SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I6P104

Volume 12 Issue 6, 44-53, June 2025
© 2025 Seventh Sense Research Group®

Original Article

Integrating Semantic Parsing with Dependency Parsing for Malayalam: A Framework for Enhanced Syntactic and Semantic Understanding

P.V. Ajusha[1], A.P. Ajees[2]

[1] School of Information Science and Technology, Kannur University, Kerala, India.
[2]Department of Computer Science, Cochin University of Science and Technology, Kerala, India.

[1]Corresponding Author : ajusha.pv@outlook.com

Received: 15 April 2025 Revised: 17 May 2025 Accepted: 18 June 2025 Published: 27 June 2025

Abstract *- The morphological complexity of the Malayalam language poses significant challenges for dependency parsing, demanding accurate syntactic and semantic analysis to advance natural language processing (NLP) for low-resource languages. This study introduces a dependency parsing approach that combines the Cross-Lingual Language Model with Roberta (XLM-Roberta) as the shared encoder and a biaffine attention mechanism for parsing with a span-based predictor for SRL. XLM-Roberta is a transformer-based multilingual model that produces high-dimensional contextual embeddings for Malayalam sentences to provide a robust syntactic and semantic analysis foundation. The biaffine attention mechanism is employed by the dependency parsing decoder to predict head-dependent relationships and assign syntactic dependency labels. The Span-Based Predictor employed for SRL assigns semantic roles to spans within sentences to effectively handle long-range dependencies common in complex languages like Malayalam. The dataset comprises a manually annotated Malayalam treebank, ensuring complete syntactic and semantic coverage. Parsing performance was evaluated using head detection accuracy, root token identification and the processing of complex sentence structures. Evaluation results indicate that integrating morphological features improves the Unlabeled Attachment Score (UAS) from 93.70% to 95.20% and the Labeled Attachment Score (LAS) from 91.45% to 93.10%. Furthermore, head detection accuracy, root token identification and complex sentence parsing demonstrate significant improvements, with respective scores increasing to 95.40%, 93.80% and 91.60%. By addressing major challenges in Malayalam dependency parsing, this study presents an efficient and scalable solution for language processing tasks. The proposed approach demonstrates significant potential for applications like machine translation, sentiment analysis and knowledge extraction, paving the way for future developments in NLP for low-resource and morphologically rich languages.*

Keywords *- Natural language processing, XLM-Roberta, Malayalam dependency parsing, Biaffine attention mechanism, Semantic role labelling.*

1. Introduction

The field of NLP has witnessed significant progress recently, especially in the areas of syntactic and semantic understanding. While syntactic parsing focuses on the structure and relationships of words in sentences, it has made remarkable improvements with different models, such as dependency parsers. There is still a gap in obtaining a thorough understanding of both syntax and semantics, particularly for languages with rich morphological structures like Malayalam [1]. Malayalam, a Dravidian language spoken mainly in Kerala, presents unique difficulties due to its agglutinative nature, complex word forms and varied syntactic constructions [2]. Therefore, creating robust syntactic and semantic models for Malayalam remains a major challenge in computational linguistics. Conventional dependency parsing approaches focus mainly on syntactic structure identification, which refers to how words in a sentence are related [3].

These syntactic structures, known as dependency trees, capture the grammatical relations between words, where each word is dependent on a headword, forming a directed acyclic graph. While dependency parsing has achieved impressive results for many languages, the performance often declines for morphologically rich and low-resource languages like Malayalam. The challenge lies in the failure of conventional methods to accurately incorporate semantic information into syntactic structures [4].

In addition, the agglutinative nature of Malayalam, where words can contain multiple morphemes that convey complex meanings, further complicates the syntactic analysis. This results in parsing errors, especially when addressing long-distance dependencies, compound words and non-canonical word orders, which occur more prevalent in languages such as Malayalam.

By contrast, semantic parsing focuses on extracting meaning from text by learning words' roles and relationships in a sentence. It attempts to capture the context and underlying sense, for example, identifying which word serves as the subject or object in a sentence or understanding the temporal or spatial relationships. Although semantic parsing is important in meaning interpretation, it does not necessarily consider the syntactic word relationships [5].

This results in a break between the two levels of language processing. His disconnect becomes particularly problematic in tasks such as machine translation, question answering, and text summarization, where a combined syntactic-semantic perspective is essential for robust language understanding.

Thus, the research gap lies in the absence of an integrated syntactic-semantic parsing framework tailored for Malayalam, which can handle its unique linguistic challenges. Addressing this gap is critical for building effective NLP systems that can perform reliably on morphologically rich and syntactically flexible languages. This paper proposes a novel parsing framework that integrates dependency parsing with semantic role labeling (SRL) to improve both syntactic and semantic analysis of Malayalam text. The key objectives of the study are given below:

- To design and implement a dependency framework using the XLM-Roberta framework with a biaffine attention mechanism for Malayalam.
- To incorporate a span-based predictor for SLR that handles long-range dependencies.
- To evaluate the parsing framework using metrics such as LAS and UAS.

The succeeding sections of the study are arranged as follows: A comprehensive analysis of conventional methods in dependency parsing is offered in Section 2. Section 3 provides the comprehensive methodology. Section 4 deliberates the obtained findings. Finally, Section 5 summarizes the method by concluding the findings.

2. Related Works

Sakthi Vel et al. [7] proposed a translation-based Cross-Language Information Retrieval (CLIR) framework for Tamil and Malayalam using a Long Short-Term Memory (LSTM) encoder-decoder model. The methodology involved training on two bilingual parallel corpora, each with 373 sentence pairs, collected from CLARIN-ERIC. The model incorporated modules for query expansion, translation, and language modeling. BLEU score evaluation showed superior performance of the LSTM model over Google Translate, with scores of 0.912 for English-Tamil and 0.954 for Malayalam-Tamil, compared to 0.874 and 0.752, respectively. The LSTM model effectively handled sequence-to-sequence translation. A key limitation was the small dataset size.

A method for interactive semantic parsing was proposed by Yao et al. [8] named MISP to increase parsing accuracy. The WikiSQL and spider datasets were used to develop the text for parsing operations. While achieving better accuracy, the method achieved less user feedback. Khalifa et al. [9] addressed the issue of fine-tuning PLM on Dialectal Arabic (DA) using MSA data. The study examined the zero-shot performance of PLMs using self-training with unlabeled DA. When evaluated on dialectal Arabic, there is a noticeable decline in performance. The approach developed a model for sarcasm detection, named entity identification, and part-of-speech tagging across different dialectal kinds. The methods showed better accuracy in self-training and improvements in zero-shot MSA-to-DA transfer.

Mollakuqe et al. [10] addressed a challenge of part-of-speech tagging in the context of the Albanian language by providing a set of part-of-speech tags. The authors introduced a substantial corpus that contained over 250000 tokens with a medium-sized tag set for in-depth syntactic and morphological analysis. A major limitation of the method was the lack of discussion on difficulties within the manual annotations, which affected the performance of the model. Also, the large corpus size posed challenges for some NLP applications.

In the context of dependency parsing, Zmigrod et al. [11] noticed an important factor: the distinction between dependency trees and spanning trees. In most of the methods, the distinction was not fully considered. The study examined the degree to which the current parsers violate this constraint and noticed the performance decline as the training set decreases. A major limitation of the method was that it did not specifically address the practical implications.

Park et al. [12] proposed the Korean language Understanding evaluation (KLUE), which is composed of eight tasks related to understanding the Korean language. The authors provided KLUE from scratch while adhering to copyrights. The method recognized many challenges, mostly related to overall performance across diverse tasks and lacking a single metric for NLU capability. A morpheme-based annotation method for Korean dependency parsing within the Universal dependency framework was proposed by Chen et al. [13]. The authors developed conversion scripts to facilitate the transition that transforms current universal dependencies into the suggested morpheme-based structure. The conversion scripts and datasets offered a useful tool for the research community to facilitate advancements in this field.

Albertson et al. [14] proposed TextMix, a Computer Assisted Language Learning application. The method used NLP to generate "sentence scramble" learning tasks. TextMix addressed restrictions in the conventional methods by parsing and scrambling syntactic elements and connecting them to API for real-world sentence exposure. It offered a method of chunking and rearranging sentences for language learning to enhance syntactic awareness. The method recognized potential learning advantages but did not offer empirical evidence or an evaluation of TextMix's efficiency in improving language learning outcomes.

Zmigrod et al. [15] explored the link between generating dependency trees in NLP and the concept of maximum spanning trees in directed graphs by introducing k-best dependency trees. A variety of dependency trees are considered by expanding the K-best dependency tree decoding to handle the root constraint.

Sandhan et al. [16] introduced TransLIST, a Sanskrit tokenizer built on a Transformer architecture. It combines character-level encoding with hidden word-level details, uses a soft-masked attention mechanism to highlight important words, and applies a path ranking algorithm to fix tokenization errors. Experiments on benchmark datasets demonstrated an average 7.2-point improvement in the perfect match (PM) metric over current methods. Kumar et al. [17] employed BERT-based tokenization and machine learning algorithms to predict dependency relations. An oversampling technique, namely SMOTE, was applied to address data imbalance, resulting in improved parsing results with a label accuracy of 69.94%.

Despite notable advancements, existing multilingual NLP and Cross-Language Information Retrieval (CLIR) methods face several limitations. One major constraint is the limited size and scope of available datasets, which hinders model generalizability and domain adaptability [7, 10]. Manual annotation inconsistencies and lack of discussion on annotation challenges further affect model accuracy and reliability, particularly in syntactic and morphological tasks [10]. In dialectal language processing, performance significantly declines when models trained on standard language data are applied to dialects, highlighting the need for dialect-specific tuning [9]. Dependency parsing approaches often overlook critical structural constraints, such as the distinction between dependency and spanning trees, leading to reduced performance with smaller training datasets [11]. Furthermore, while certain tools show promising theoretical contributions, they lack empirical validation or comprehensive evaluation, limiting their practical applicability in real-world scenarios [14]. These limitations underscore the need for richer datasets and domain adaptation strategies that can successfully handle the linguistic intricacies of Malayalam and improve parsing performance for more effective applications in natural language processing techniques.

3. Materials and Methods

The complex morphological structure of the Malayalam language poses significant challenges for dependency parsing, which requires accurate syntactic and semantic analysis for advancing NLP in low-resource languages. XLM Roberta, combined with a biaffine attention mechanism, is employed in the suggested architecture for the dependency parsing decoder, and for the semantic role labeling decoder, the architecture utilizes a span-based predictor. Figure 1 shows the proposed method. This study proposes an integrated method combining dependency and semantic parsing to enhance Malayalam text's overall syntactic and semantic comprehension. Combining the advantages of both parsing paradigms, the suggested approach seeks to extract the syntactic structure and the underlying meaning of Malayalam sentences. The proposed method introduces a hybrid model that combines existing dependency parsing algorithms with advanced semantic analysis techniques to enable a more comprehensive understanding of Malayalam sentences. Firstly, this approach is expected to significantly enhance the performance of various NLP applications by improving syntactic and semantic interpretation. Secondly, it addresses the unique linguistic complexities of the Malayalam language by incorporating its syntactic structure and subtle semantic nuances, which are features frequently disregarded by conventional methods.

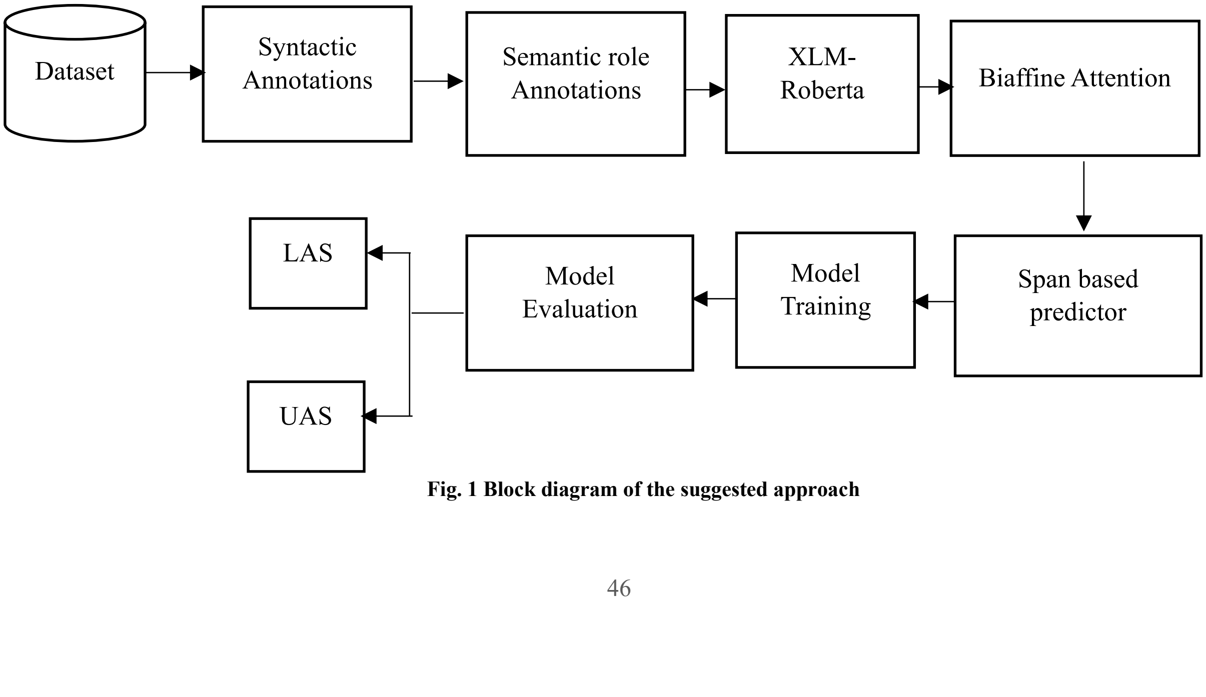

Fig. 1 Block diagram of the suggested approach

3.1. Dataset Description

The dataset plays a significant role in combining dependency and semantic parsing for the Malayalam language. The approach utilizes the existing resources and supplements with manual annotations for SLR. The Universal Dependencies (UD) Malayalam treebank has annotated sentences, a well-known source of syntactic parsing tasks. Various formal and informal Malayalam sentences are provided with dependency relations in the dataset. The collection of 5000 sentences from the UD treebank supplemented with semantic role annotations was employed in the suggested study to perform semantic parsing.

Morphological analysis was conducted using Helsinki Finite-State Technology (HFST) based tools to enhance the utility of the dataset. For morphological parsing of Malayalam, these tools extract necessary information like case markers, root words and tense. For model training and to enhance the diversity of the dataset, minor structural variations were added, thus producing synthetic data. The syntactic and semantic annotations together provide a comprehensive resource for developing and evaluating the suggested integrated framework.

Figure 2 visualizes the distribution of simple, compound and complex sentence types within the Malayalam Treebank dataset. The data reveals that simple sentences dominate the dataset, comprising 60% of the total sentences. This prevalence underscores the prominence of basic syntactic structures in Malayalam, reflecting their widespread use in both formal and informal contexts.

Compound sentences accounting for 30% of the dataset indicate the moderate representation of coordination between clauses. This suggests that compound sentences are less frequent than simple ones, but they still play a significant role in constructing slightly more intricate syntactic structures. In contrast, complex sentences represent only 10% of the dataset.

This limited presence highlights the relative infrequency of subordinate clause structures, which are attributed to the dataset's focus on syntactically simpler or less dense texts. However, their inclusion remains vital for studying advanced syntactic and semantic relationships in Malayalam.

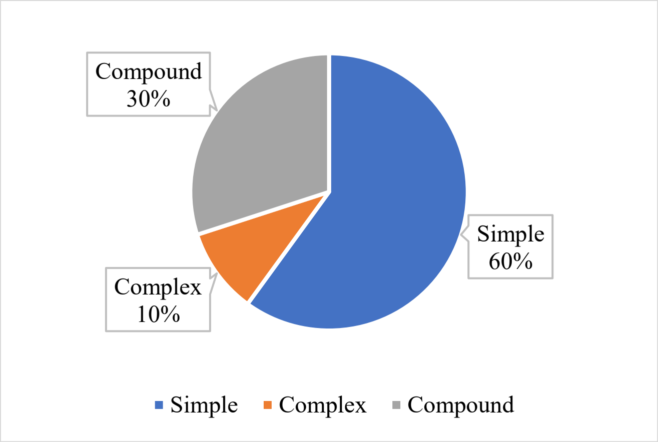

Fig. 2 Distribution of sentences in Malayalam treebank

Figure 3 provides insights into the distribution of various morphological features, namely root words, case markers, tense, number and person within the Malayalam dataset. Among these features, root words are the most frequently occurring, with a count of 2500, reflecting their fundamental role in representing the base meaning of words in Malayalam, which is a morphologically rich language. Case markers appearing 1800 times are the second most prominent feature. Their high frequency underscores the importance of case systems in Malayalam, which heavily relies on them to indicate syntactic relationships like subject-object alignment.

Tense-related features are also significant, with 1,500 highlighting their essential role in capturing temporal aspects within sentences. This demonstrates the dataset's capability to represent temporal complexities in Malayalam. Features such as number (1200) and person (1000) occur less frequently, suggesting that while these features contribute to subject-verb agreement and pronoun usage, their presence is comparatively limited within the dataset.

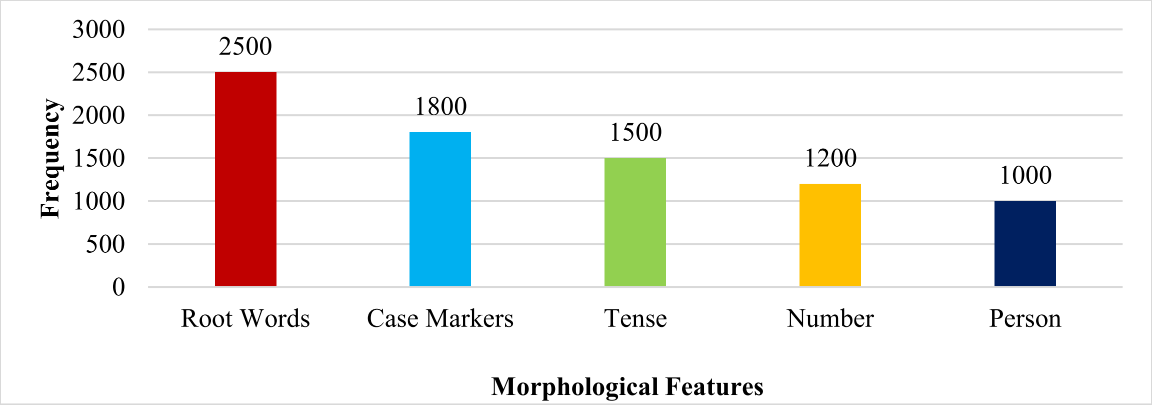

Fig. 3 Morphological features in the dataset

3.2. Shared Encoder: XLM-Roberta

The proposed architecture integrates XLM-Roberta as the shared encoder to generate contextual embeddings for Malayalam sentences paired with a biaffine attention mechanism for dependency parsing. XLM-Roberta is a transformer-based multilingual model that is the shared encoder to extract high-dimensional contextual representations of input sentences. The process begins with tokenization using the XLM-Roberta tokenizer, which divides sentences into sub-word units to handle the complex morphology of Malayalam. Morphological features such as root words, case markers, tense, number and person are extracted using HFST tools and appended as auxiliary inputs, enriching the token embeddings with linguistic information.

The architecture comprises multiple transformer layers, each consisting of multi-head self-attention and feed-forward sublayers. The self-attention mechanism computes dependencies between tokens to capture both local and global contexts. Positional embeddings are added to the token embeddings to preserve the sequential nature of the input. The output is a sequence of contextual embeddings, each representing a token enriched with syntactic and semantic information. The dependency parsing decoder in the proposed architecture utilizes a biaffine attention mechanism to predict syntactic head-dependent relationships and assign dependency labels [18]. This mechanism operates in three primary stages: feature extraction, biaffine scoring and dependency prediction.

After contextual embeddings are generated by the XLM-Roberta shared encoder for the input sentence, these embeddings are passed through task-specific Feed-Forward Neural Networks (FFNNs). This step produces two representations: one for the head token and one for the dependent token. Let the token embeddings from XLM-Roberta be represented as h_i for token i, where $i = 1, 2 \ldots, n$. The FFNNs process these embeddings to create separate representations for the head and dependent tokens, as expressed in Equation (1) and Equation (2), respectively.

$$h_i^{head} = FFFN_{head}(h_i) \qquad (1)$$

$$h_i^{dep} = FFFN_{dep}(h_i) \qquad (2)$$

In the biaffine stage, the biaffine layer computes pairwise scores for all possible head-dependent pairs in the sentence. The scores are computed using a bilinear transformation, which combines the head and dependent representations into a score that quantifies the likelihood of one token being the head of the other. For a given pair of tokens i and j, the biaffine scoring function is expressed in Equation (3).

$$S_{ij} = h_i^{head} W h_j^{dep} + b \qquad (3)$$

Where S_{ij} is the score for the pair (i, j), W is the bilinear weight matrix, h_i^{head} is the head representation of token i, h_j^{dep} Is the dependent representation of token j, b is a bias term. The matrix W captures the interaction between the head and dependent tokens. The resulting score S_{ij} Indicates how likely token i is the head of token j in the syntactic tree.

The highest-scoring head for each token is selected to form the syntactic dependency tree. Let $\hat{h}_i$ Be the predicted head for token i. The dependency prediction is performed by selecting the head with the highest score for each token per Equation (4).

$$\widehat{h_\iota} = \arg\max_j S_{ij} \qquad (4)$$

Where $\widehat{h_\iota}$ Is the index of the token selected as the head for token i. Additionally, dependency labels are assigned to the dependency arcs in the tree. These labels are predicted using additional FFNNs, which classify the relationships between head-dependent pairs. The dependency label Y_{ij} For a pair (i, j) is predicted as per Equation (5).

$$Y_{ij} = FFNN_{label}(h_i^{head}, h_j^{dep}) \qquad (5)$$

where $FFNN_{label}$ It is a feed-forward neural network that outputs the dependency label for the pair.

3.3. Span-Based Predictor for Semantic Role Labeling Decoder

The Span-Based Predictor used for SRL in this architecture assigns semantic roles to spans in a sentence. The span-based approach is particularly useful in handling long-range dependencies between words, which are common in natural language, especially in complex languages like Malayalam [19].

The architecture identifies spans of words that form the arguments and predicates in a sentence and then labels them according to their corresponding semantic roles. The process is broken down into three key components, namely span representation, span scoring and role classification [20].

For a given sentence, the input to the span-based predictor is the set of contextual embeddings produced by the shared encoder (XLM-Roberta). For a span $[i, j]$, the representation of the span is typically formed by combining the embeddings of tokens at the boundaries of the span. Let h_i And j be the contextual embeddings of the start token i and end token j, respectively. The span representation $h_{[i,j]}$ It is computed as per Equation (6).

$$h_{[i,j]} = Concat(h_i, h_j) \qquad (6)$$

Where Concat $(\cdot)$ represents the concatenation of the two token embeddings. Once the span representations are obtained, the next step is to score the span. For each candidate span $[i, j]$, computes a score S_{ij} That reflects how likely this span corresponds to an argument for a semantic role. The score for each span S_{ij} It is computed using an FFNN applied to the span representation. The span scoring is calculated as per Equation (7).

$$S_{ij} = h_{[i,j]} W_{span} h_{[i,j]} + b_{span} \tag{7}$$

Where $h_{[i,j]}$ Is the span representation, W_{span} It is a learned weight matrix that maps the span representation to a scalar score and b_{span} It is a biased term. The span S_{ij}It has a valid semantic role, indicating how the span probably represents an argument. After scoring the spans, the next step is to classify the semantic role of each valid span. For each span $[i, j]$ that has been identified as an argument, classify it with a semantic role label. This is done by passing the span representation $h_{[i,j]}$ Through another FFNN for role classification. Let Y_{ij} Represent the semantic role label for the span $[i, j]$. The role prediction is expressed as per Equation (8).

$$Y_{ij} = \text{FFNN}_{role}(h_{[ij]}) \tag{8}$$

Where FFNN_{role} It is an FFNN that classifies the span representation into one of the possible semantic role labels. The output Y_{ij} Is the semantic role label assigned to the span $[i, j]$ indicating the function of the span in relation to the predicate.

3.4. Hardware and Software Setup

The study uses a robust hardware and software setup to handle the computational demands of training and evaluating the integrated framework for semantic and dependency parsing in Malayalam. Hardware includes a high-performance CPU, such as Intel Core i9 with 12 cores and 24 threads and a powerful GPU like NVIDIA RTX 3090 (24 GB VRAM) or higher, for accelerated training of transformer models like XLM-Roberta.

A 32 GB RAM is used for memory-intensive tasks, along with a 1 TB SSD for efficient storage of datasets and model checkpoints. Python serves as the primary programming language on the software side, with TensorFlow for model implementation and Hugging Face's Transformers library for integrating XLM-Roberta. The Helsinki Finite-State Transducer (HFST) toolkit is used for morphological analysis, while NumPy, pandas and scikit-learn support data preprocessing and evaluation.

4. Results and Discussion

Two significant metrics for evaluating the dependency parsers are Labeled Attachment Score (LAS) and Unlabeled Attachment Score (UAS). The accuracy of the syntactic dependency trees is assessed by comparing the parser-generated outputs with the standard annotations in the dataset. To evaluate the quality of parsing, LAS and UAS are employed, as given in Equation (9) and Equation (10), respectively.

UAS considers solely the structural accuracy of the dependency tree. It calculates the percentage of tokens in a sentence for which the syntactic head is correctly predicted regardless of the dependency label given to the arcs. Ignoring semantics evaluates how effectively the parser identifies the structural relationship between words.

$$UAS = \frac{Number\ of\ correctly\ predicted\ head\ dependent\ pairs}{Total\ number\ of\ tokens} \tag{9}$$

LAS is a more comprehensive metric that accounts for structural and labelling accuracy. LAS computes the token % for which the predicted head and dependency label are correct. This metric is significant for morphologically rich languages like Malayalam, as it provides a deeper understanding of the parser's ability to capture a sentence's grammatical and semantic nuances.

$$LAS = \frac{\begin{matrix} Number\ of\ correctly\ predicted\ head\ dependent\ pairs \\ with\ correct\ labels \end{matrix}}{Total\ number\ of\ tokens} \tag{10}$$

The difference between LAS and UAS is indicative of their unique focus. LAS evaluates the ability to provide both grammatical and semantic labels correctly, while UAS assesses the accuracy of structural predictions. Together, they provide a well-rounded view of parser performance. Thus, LAS and UAS are therefore essential in determining how well the parser handles the complexities of the syntax and the semantics of the natural languages, particularly for richly morphologically structured languages with free word order.

Table 1 presents the comparison of two proposed frameworks, namely XLM-Roberta and the other, including morphological features in terms of UAS and LAS. The XLM-Roberta-based framework, with the addition of the Biaffine attention mechanism, achieves a UAS of 93.70% and a LAS of 91.45%. However, the inclusion of morphological features in the framework leads to an increase in both metrics, with a UAS of 95.20% and an LAS of 93.10%. This suggests that including morphological features enhances the model's accuracy in syntactic dependency parsing, leading to better attachment predictions and more accurate labelling of dependencies.

Table 1. Comparison of UAS and LAS scores for proposed frameworks with and without morphological features

Model	UAS (%)	LAS (%)
Proposed Framework (XLM-Roberta + Biaffine)	93.70	91.45
Proposed Framework (with Morphological Features)	95.20	93.10

Table 2 and Figure 4 compare the performance of two proposed frameworks on various syntactic evaluation metrics. The morphological features-based framework performs better than the XLM-Roberta model on all evaluated metrics. The accuracy of correct head detection is increased from 92.80% to 95.40%, root token identification is enhanced from 91.50% to 93.80% and complex sentence is enhanced from 88.20% to 91.60%. These enhancements indicate that the addition of morphological features increases the model's ability to identify accurate syntactic structures like head tokens, root tokens and complex sentence structures, rendering it more suitable for use in deep syntactic analysis tasks.

Table 2. Syntactic head identification by various methods

Model	Total Tokens Evaluated	Correct Head Detection (%)	Root Token Identification (%)	Complex Sentence Accuracy (%)
Proposed Framework (XLM-Roberta)	10,000	92.80	91.50	88.20
Proposed Framework (with Morphological Features)	10,000	95.40	93.80	91.60

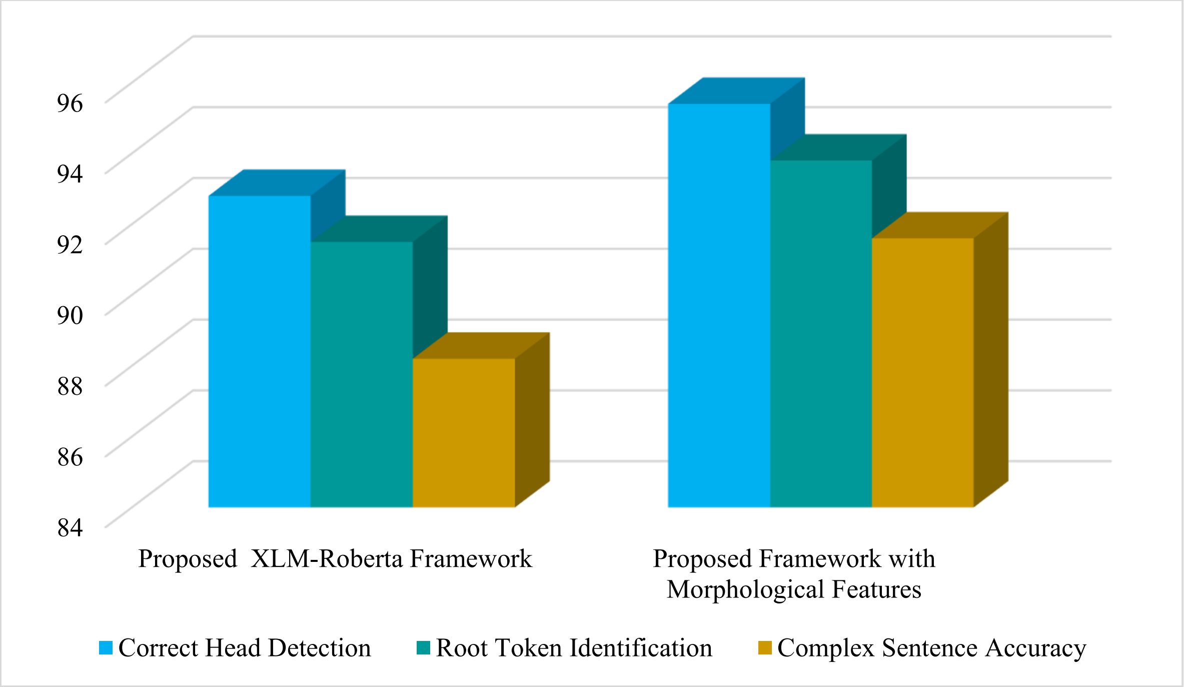

Fig. 4 Comparison of syntactic head identification

Table 3 and Figure 5 present a comparison of the performance of the two proposed frameworks across several linguistic aspects. The inclusion of morphological features significantly enhances the framework's capabilities, as indicated by higher scores in all metrics. Specifically, the embedding quality improves from 92.60% to 95.40%, morphological feature retention increases from 90.70% to 94.20%, and case marker handling and tense detection accuracies show notable improvements. Additionally, the token-level context preservation score rises from 91.80% to 94.90%. These results suggest that integrating morphological features leads to a more robust and accurate model, particularly in handling linguistic complexities related to case markers, tense and token-level context.

Table. 3 Performance comparison of the baseline and proposed frameworks for Malayalam parsing and contextual analysis

Model	Embedding Quality Score (%)	Morphological Feature Retention (%)	Case Marker Handling Accuracy (%)	Tense Detection Accuracy (%)	Token-Level Context Preservation (%)
Proposed Framework (XLM-Roberta)	92.60	90.70	88.40	89.50	91.80
Proposed Framework (with Morphological Features)	95.40	94.20	92.30	93.10	94.90

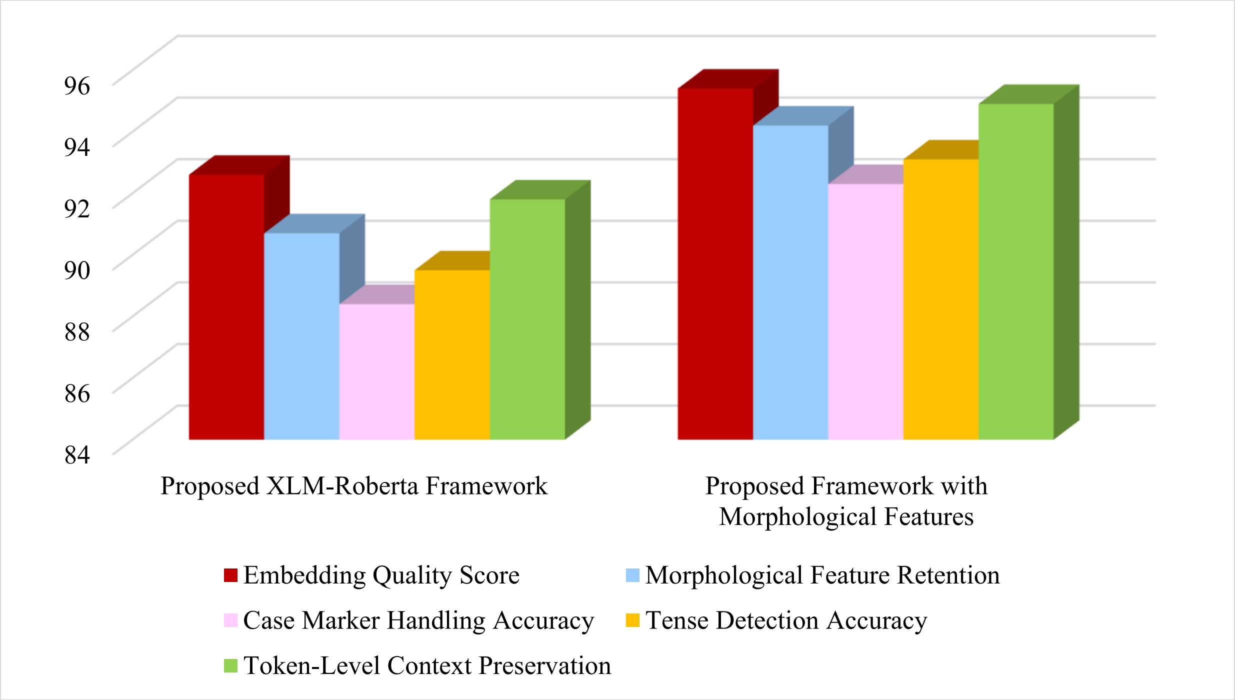

Fig. 5 Contextual analysis of the proposed model

The error analysis, as given in Table 4, provides insights into the types of mistakes made by the proposed parsing system while handling the syntactic structure of Malayalam sentences. A significant portion of errors, approximately 25%, arises from the incorrect parsing of compound words, a common feature in Malayalam, where two words are combined to form a new meaning. This suggests that the model struggles to correctly identify and process compound word structures due to the language's rich morphology and the lack of explicit separation between compound elements in the sentence.

Another considerable source of errors (17%) is handling case markers, which play an essential part in Malayalam grammar. Misidentification of these markers indicates that the model cannot understand the syntactic roles of these markers, potentially due to the language's agglutinative nature and the subtleties involved in their usage. Around 20% of errors are linked to free word order, which is important in Malayalam syntax. The model struggles with sentences where word order varies, and dependencies must be captured over longer text spans. This demonstrates the challenge of handling languages where the syntax is flexible, and word order becomes a matter of choice.

Other errors, including ambiguous word sense identification (10%) and pronoun resolution (15%), indicate limitations in the model to process context-dependent tasks. Ambiguous word sense errors indicate problems in solving polysemy, whereas pronoun resolution errors indicate issues in the ability to connect pronouns to their appropriate antecedents. This error analysis informs future enhancements in the model, highlighting improved management of Malayalam's complex linguistic features like compound words, case markers and free word order.

Table 4. Analysis of grammatical parsing challenges in Malayalam language processing

Error Type	Number of Errors	Percentage of Total Errors	Description	Possible Causes
Compound Words	120	25%	Incorrectly parsed compound words often split	Failure to recognize compound word structures
Case Markers	80	17%	Misidentification of case markers or their role	Inadequate understanding of syntactic roles
Free Word Order	95	20%	Incorrect dependency relations due to free word order	Failure to capture long-range dependencies
Ambiguous Word Sense	50	10%	Incorrect identification of word senses in context	Lack of contextual understanding of polysemy
Pronoun Resolution	70	15%	Errors in resolving references between pronouns	Difficulty in resolving pronouns with respect to antecedents
Others (Miscellaneous)	35	8%	Miscellaneous parsing errors	Overfitting on training data or low-resource issues

5. Conclusion

Overcoming the linguistic constraints of the morphological complexity of Malayalam, a low-resource language, is essential for advancing dependency parsing techniques suited to its distinctive syntactic and semantic structures. Introducing an effective dependency parsing framework for solving the morphological complexity of the Malayalam language, this study employs XLM-Roberta as the shared encoder coupled with a biaffine attention mechanism for dependency parsing and a span-based predictor for SLR in order to achieve substantial improvements in parsing accuracy. Adding morphological features significantly enhances the Unlabelled Attachment Score (UAS) from 93.70% to 95.20% and the Labelled Attachment Score (LAS) from 91.45% to 93.10%. The metrics such as head detection accuracy, root token identification and complex sentence parsing also achieve substantial gains, up to 95.40%, 93.80% and 91.60%, respectively. These observations depict the effectiveness of combining advanced contextual embeddings with morphological feature integration to overcome the linguistic challenges of Malayalam. The suggested framework improves syntactic and semantic analysis and exhibits substantial potential for applications in machine translation, sentiment analysis and knowledge extraction.

Acknowledgements

The author sincerely thanks the supervisor for guidance and continuous support throughout the study.

References

[1] Artur Kulmizev, and Joakim Nivre, “Schrödinger's Tree-On Syntax and Neural Language Models,” *Frontiers in Artificial Intelligence*, vol. 5, pp. 1-14, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Gayathri G. Krishnan, “*Malayalam Morphosyntax: Inflectional Features and their Acquisition*,” Thesis, Indian Institute of Technology Bombay, pp. 1-275, 2020. [Google Scholar] [Publisher Link]

[3] Haitao Liu, Chunshan Xu, and Junying Liang, “Dependency Distance: A New Perspective on Syntactic Patterns in Natural Languages,” *Physics of Life Reviews*, vol. 21, pp. 171-193, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[4] Abhilasha A. Kumar, “Semantic Memory: A Review of Methods, Models, and Current Challenges,” *Psychonomic Bulletin & Review*, vol. 28, pp. 40-80, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Aishwarya Kamath, and Rajarshi Das, “A Survey on Semantic Parsing,” *Arxiv*, pp. 1-22, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Carlos Gómez-Rodríguez, Iago Alonso-Alonso, and David Vilares, “How Important is Syntactic Parsing Accuracy? An Empirical Evaluation on Rule-based Sentiment Analysis,” *Artificial Intelligence Review*, vol. 52, pp. 2081-2097, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] S. Sakthi Vel, and R. Priya, “A Translation Framework for Cross Language Information Retrieval in Tamil and Malayalam,” *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, vol. 12, no. 2, pp. 319-332, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[8] Ziyu Yao et al., “Model-Based Interactive Semantic Parsing: A Unified Framework and A Text-to-SQL Case Study,” *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, pp. 5447-5458, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Muhammad Khalifa, Hesham Hassan, and Aly Fahmy, “Zero-Resource Multi-Dialectal Arabic Natural Language Understanding,” *Arxiv*, pp. 1-15, 2021. [CrossRef] [Publisher Link]

[10] Diellza Nagavci Mati, Mentor Hamiti, and Elissa Mollakuqe, “Morphological Tagging and Lemmatization in the Albanian Language,” *Seeu Review*, vol. 18, no. 2, pp. 4-16, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[11] Ran Zmigrod, Tim Vieira, and Ryan Cotterell, “Please Mind the Root: Decoding Arborescences for Dependency Parsing,” *Arxiv*, pp. 1-11, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Sungjoon Park et al., “KLUE: Korean Language Understanding Evaluation,” *Arxiv*, pp. 1-76, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Yige Chen et al, “Yet Another Format of Universal Dependencies for Korean,” *Arxiv*, pp. 1-6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Brendon Albertson, “TextMix: Using NLP and APIs to Generate Chunked Sentence Scramble Tasks,” *29th Conference CALL and Professionalisation: Short Papers from EUROCALL 2021*, pp. 6-11, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] Ran Zmigrod, Tim Vieira, and Ryan Cotterell, “On Finding the K-Best Non-Projective Dependency Trees,” *Arxiv*, pp. 1-14, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Jivnesh Sandhan et al., “TransLIST: A Transformer-Based Linguistically Informed Sanskrit Tokenizer,” *Arxiv*, pp. 1-11, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] C.S. Ayush Kumar et al., “BERT-Based Sequence Labelling Approach for Dependency Parsing in Tamil,” *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, Dublin, Ireland, pp. 1-8, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Zhengqiao Zeng et al., “A Conspiracy Theory Text Detection Method Based on RoBERTa and XLM-RoBERTa Models,” *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024)*, Grenoble, France, pp. 1-5, 2024. [Google Scholar] [Publisher Link]

[19] Jiangzhou Ji, Yaohan He, and Jinlong Li, “A Biaffine Attention-Based Approach for Event Factor Extraction,” *Conference Proceedings 6th China Conference on Knowledge Graph and Semantic Computing*, Guangzhou, China, pp. 1-10, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Rexhina Blloshmi et al., “Generating Senses and Roles: An End-to-End Model for Dependency-and Span-based Semantic Role Labeling,” *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*,” pp. 3786-3793, 2021. [Google Scholar] [Publisher Link]