Family helps one another: Dravidian NLP suite for Natural Language Understanding

Abhinav P M, Priyanka Dasari, Nagaraju Vuppala, Parameswari Krishnamurthy
IIIT Hyderabad
{abhinav.pm, dasari.priyanka, nagaraju.vuppala}@research.iiit.ac.in
param.krishna@iiit.ac.in

Abstract

Developing robust Natural Language Understanding (NLU) for morphologically rich Dravidian languages like Kannada, Malayalam, Tamil, and Telugu presents significant challenges due to their agglutinative nature and syntactic complexity. In this work, we present the Dravidian NLP Suite tackling five core tasks: Morphological Analysis (MA), POS Tagging (POS), Named Entity Recognition (NER), Dependency Parsing (DEP), and Coreference Resolution (CR), trained for monolingual models and multilingual models. To facilitate this, we present the *Dravida dataset*, meticulously annotated multilingual corpus for these tasks across all four languages. Our experiments demonstrate that a multilingual model, which utilizes shared linguistic features and cross-lingual patterns inherent to the Dravidian family, consistently outperforms its monolingual counterparts across all tasks. These findings suggest that multilingual learning is an effective approach for enhancing Natural Language Understanding (NLU) capabilities, particularly for languages belonging to the same family. To the best of our knowledge, this is the first work to jointly address all these core tasks on the Dravidian languages.

1 Introduction

The Dravidian language family, one of the world's oldest and most diverse linguistic groups, comprises over 80 languages spoken primarily in South India, Sri Lanka, and parts of Southeast Asia (Amritavalli and Narasimhan). This study focuses on four widely spoken Dravidian languages-Kannada (ka), Malayalam (ml), Tamil (ta), and Telugu (te)-which together account for over 200 million native speakers, based on the 2011 Census of India (Census, 2011).

These languages are typologically complex, featuring agglutinative morphology, free word order, and pro-drop characteristics, where subject pronouns are often omitted and inferred contextually. They lack prefixes and infixes; instead, grammatical relations are expressed solely through suffixation and compounding (Krishnamurti, 2003).

Despite their cultural and demographic significance, computational resources for Dravidian languages remain limited, especially when compared to better-resourced Indo-European languages-posing a major challenge to progress in Natural Language Processing (NLP). Their agglutinative structure requires sophisticated tools to segment words into morphemes (Creutz and Lagus, 2007), while the flexible word order complicates dependency parsing and structural interpretation. To make substantial progress, NLP for these languages must robustly handle these fundamental layers of linguistic analysis-from morphology to discourse-which collectively support comprehensive understanding.

This paper aims to bridge this gap by presenting a systematic effort to develop and evaluate foundational NLU capabilities for these languages. Our primary contributions are:

1. **The Dravidian NLP Suite:** A comprehensive suite of models for five core NLP tasks: MA, POS, NER, DEP, and CR specifically adapted to Kannada, Malayalam, Tamil, and Telugu.

2. **The Dravida Dataset:** A richly annotated corpus meticulously created to support all five tasks for each of the four languages, serving as a crucial resource for model development and comparative evaluation within this work and for future research. The dataset and resources introduced in this work are publicly available at[1].

3. **Systematic Monolingual vs. Multilingual Evaluation:** We conduct a thorough comparison of monolingual models against a multilin-

[1] https://github.com/abhinav-pm/DravidianNLP_paper

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1926–1941

gual model, including evaluation against the multilingual baseline, demonstrating the significant advantages of cross-lingual learning by leveraging shared linguistic patterns within the Dravidian family.

Our work is guided by the well-established hypothesis that linguistic relatedness enables effective cross-lingual transfer within language families (Pires et al., 2019; Conneau et al., 2019). Training multilingual models across Kannada, Malayalam, Tamil, and Telugu significantly outperforms monolingual models, confirming the benefits of shared learning. This suggests that even in low-resource settings, incorporating rich linguistic representations through shared learning is a powerful strategy (Jurafsky, 2000).

2 Related Work

Natural Language Processing (NLP) for Indian languages has progressed from rule-based and statistical methods to deep learning approaches addressing their rich morphosyntactic diversity and low-resource challenges. This review focuses on five core tasks MA, POS, NER, DEP and CR, tracing developments from finite-state and machine learning models to recent transformer-based and cross-lingual transfer techniques that have advanced NLP for Indian languages.

Morphological Analysis for Indian languages has been explored using both rule-based and neural approaches (Sarveswaran et al., 2021). Early efforts include a Telugu Morphological Analyzer (Rao et al., 2011) organized a linguistic database and employing computing resources effectively. This work is based on the word and paradigm approach (Hockett, 1954; Menaka et al., 2010; Rajendran, 2009). Improvised Tamil morphological analysis using the Apertium platform (Parameshwari, 2011), refining linguistic databases for improved inflectional and derivational analysis. A rule-based finite-state transducer (FST) for Kannada implemented by (Veerappan et al., 2011). For Telugu, (Srinivasu and Manivannan, 2018) developed a morphological analyzer and generator using the Item and Process model with finite-state machines (FSM). More recently, neural-based approaches have been explored. A large annotated Telugu dataset and evaluated it on transformer-based models (Dasari et al., 2023). For multilingual morphological analysis, (Mishra et al., 2024) proposed a multi-task learning framework integrating POS tagging, chunking, and morphological analysis, leveraging fine-tuned contextual embeddings across multiple Indian languages. (Pawar et al., 2023) shows that pretrained multilingual models like mT5 improve root extraction and GNP tagging in low-resource Indian languages through cross-lingual transfer. These approaches complement large-scale cross-lingual morphological databases such as UniMorph (Kirov et al., 2018), which provide universal morphological feature schemas across languages.

Parts of Speech Tagging helps in assigning markers to words in the sentence that give those words some lexical meaning. POS taggers are specially developed for several Indian languages such as Hindi, Bengali, Telugu, etc. (Antony and Soman, 2011) and (Antony and Soman, 2010) developed a POS tagger for Kannada using a Support Vector Machine (SVM)-based approach. Graph-based and cross-lingual techniques are explored to improve POS tagging for low-resource languages. (Imani et al., 2022) proposed a graph-based label propagation method, utilizing multilingual word alignments and graph neural networks to enhance label transfer, and (Kim et al., 2017) introduced cross-lingual transfer learning. Similarly, (Chaudhary et al., 2021) introduced an active learning approach to minimize annotation efforts while reducing conflicts in POS tagging optimization. Recent advancements include active learning approaches by (Chaudhary et al., 2021) and (Kumar et al., 2024) introduced UD-compliant POS tagging datasets for low-resource Indic languages like Angika, Magahi, and Bhojpuri.

Named Entity Recognition research initially focused on English and major languages, with limited work on Indian languages. Multilingual learning enhances NER in low-resource languages by leveraging data from closely related languages through shared neural network layers (Murthy et al., 2018). For Indian languages, (A P et al., 2019) proposed a deep learning-based NER system, utilizing character, word, and affix-level embeddings. More recently, (Bahad et al., 2024) introduced a human-annotated corpus of 40K sentences for four Indian languages and proposed a multilingual NER model tailored for Indian language families.

Dependency Parsing is essential for understanding syntactic relations. Early work by (Kosaraju et al., 2010) evaluated the Malt parser on Hindi, Telugu, and Bangla, experimenting with various parsing strategies. Later, (Nallani et al., 2020) ex-

panded Telugu treebanks by automatically annotating intra-chunk dependencies using a Shift-Reduce parser, resulting in a fully expanded treebank of 3220 sentences. For Malayalam, (Stephen J et al., 2023) developed a Universal Dependencies (UD)-based treebank, ensuring cross-linguistic consistency. Similarly, (Krishnamurthy and Sarveswaran, 2021) created a morphosyntactically annotated Tamil treebank with 534 sentences, introducing language-specific relations for Tamil NLP. These efforts build upon foundational work in multilingual parsing shared tasks (Zeman et al., 2017) and widely-used tools like Stanza (Qi et al., 2020) and UDPipe (Straka and Straková, 2017).

Coreference Resolution research has progressed from rule-based models to deep learning approaches. End-to-end neural models were introduced by (Lee et al., 2017), eliminating explicit mention detection by directly considering all possible spans. This was further refined by (Lee et al., 2018a), who incorporated higher-order coreference resolution with attention-based mechanisms. For Indian languages, (Mishra et al., 2024) introduced TransMuCoRes, a multilingual coreference resolution dataset covering 31 South Asian languages. (Devi et al., 2024) fine-tuned XLM-Roberta for Tamil, Malayalam, and Hindi, showing that linguistic feature integration. To address low-resource challenges, (Rahothvarman et al., 2025) developed mGAP, a multilingual coreference dataset derived from translating the English GAP dataset into South Asian languages.

3 The Dravida Dataset: Creation and Characteristics

This section details the creation and characteristics of the *Dravida dataset*, a multilingual corpus supporting our Dravidian NLP suite. The dataset provides annotations for five core tasks- MA, POS, NER, DEP, and CR- across Kannada, Malayalam, Tamil, and Telugu. All data included met an inter-annotator agreement (IAA) score of at least 0.85. The manual annotation for tasks like NER and CR was carried out by native speakers of the respective languages. Annotation guidelines were developed iteratively, incorporating examples of edge cases. Each document was annotated by two annotators, and disagreements were resolved through discussion with a senior annotator to ensure consistency.

The Dravidian NLP Suite integrates key components; MA captures inflectional and derivational patterns; POS Tagging assigns grammatical roles; NER identifies and classifies proper names (Pillai and Sobha, 2013); DEP models syntactic relations between heads and dependents (Li et al., 2018); and CR links entity mentions across discourse for coherent interpretation. Table 1 presents a detailed linguistic analysis of Telugu sentences, highlighting core tasks in the Dravidian NLP Suite. While the exact annotation format within our Dravida dataset may vary per task, this example highlights the types of linguistic information each core task aims to capture.

Key aspects analyzed include root words, morphological features, UD POS tags (Petrov et al., 2012), named entities (Bahad et al., 2024), and head-dependent relations (yi Lee et al., 2009). Coreference chains are annotated using the mention%chain[2] format (Mujadia et al., 2016), linking referring expressions that refer to the same entity across sentences. In example 1, in the first sentence, i1 (*Kumar*) and i2 (*he*) are linked in coreference chain t1, indicating that '*he*' refers to '*Kumar*'. Similarly, i1 (*New York*) and i2 (*there*) belong to coreference chain t2, denoting the same location. The symbol % separates the mention ID from its coreference chain, ensuring clarity in discourse analysis.

1. **Te:** kumār nyūyārklō nivasistunnāḍu. atanu aydu saṃvatsarālugā akkaḍa maykrōsāphṭlō panicēstunnāḍu.
   En: "*Kumar lives in New York. He has been working there at Microsoft for five years.*"

3.1 Data Sources and Preprocessing by Task

Morphological Analysis (MA): The MA data focuses on root and suffix identification and includes features such as lexical category, gender, number, and person. For Kannada and Malayalam, we used (iiit, 2023)-annotated corpus, which is in CoNLL-U format. From this we extracted only the morphological features for our processing.

For Telugu and Tamil, we utilized in-house data, initially processed with a rule-based morphological analyzer and subsequently manually validated and corrected for accuracy. Following (Dasari et al., 2023), we applied noise removal and cleaning, filtering tokens with invalid annotations (e.g., anomalous person values or non-standard gender/number entries). Final dataset statistics are in Table 3.

[2]This format is just used to explain the coreference chains in this paper. In our coreference data annotation, we marked the referring expressions with mention spans

Tkn.no	Word	MA	POS	NER	DEP	CR
	<S>					
1	kumār *'Kumar'*	<kumār,n,any,sg,3,0>	PROPN	PER	3:nsubj	i1%t1
2	nyūyārk-lō *'New York-in'*	<nyūyārk,n,any,sg,3,lō>	NOUN	LOC	3:obl	i1%t2
3	nivasis-tunn-āḍu *'lives'*	<nivasiṃcu,v,m,sg,3,tunn>	VERB		0:root	
4	. </S>		PUNCT		3:punct	
	<S>					
1	atanu *'He'*	<atanu,pn,m,sg,3,0>	PRON		6:nsubj	i2%t1
2	aydu *'five'*	<aydu,n,any,sg,3,0>	NUM		3:nummod	
3	saṃvatsarā-lu-gā *'year-PL-ADVL'*	<saṃvatsaraṃ,n,any,pl,3,gā>	ADV		6:obl:tmod	
4	akkaḍa *'there'*	<akkaḍa,adv,any,any,any,0>	ADV		6:advmod	i2%t2
5	maykrōsāphṭ-lō *'Microsoft'-'in'*	<maykrōsāphṭ,n,any,sg,3,lō>	NOUN	ORG	6:obl	
6	panicēstunnāḍu *'work'*	<paniceyyi,v,m,sg,3,tunn>	VERB		0:root	
7	. </S>		PUNCT		6:punct	

Table 1: Sample of Telugu sentence analysis of all the Tasks in Dravidian NLP suite.

POS Tagging (POS): We used SSF-formatted data from (Bharati and Sangal, 2007) for all languages. After converting to word-POS pairs and performing preprocessing, BIS format POS tags (of Indian Standards, 2021) were mapped to the Universal Dependencies (UD) scheme (yi Lee et al., 2009). Statistics are in Table 3.

Named Entity Recognition (NER): Given the scarcity of NER-annotated data for Dravidian languages, we combined manually annotated data (Table 2, annotated using a tool detailed in the Appendix) with synthetically generated data. A portion of the manually annotated data, following BIO format, forms our test set. The remaining manually annotated sentences are combined with synthetically generated data to create the training corpus. For synthetic data, we translated Hindi NER data (Bahad et al., 2024) into the four Dravidian languages using Google Translate. High-quality word alignments are then generated using Awesome-Align (Dou and Neubig, 2021), which uses multilingual BERT embeddings (Devlin et al., 2019), a technique proven effective for low-resource scenarios (James and Krishnamurthy, 2025). This allowed accurate projection of BIO tags from Hindi to the Dravidian languages. The process is illustrated in Figure 1.

While large-scale manual validation of the synthetic data was not feasible, we performed a small-scale error analysis to assess its quality. Out of 344 tokens reviewed in Telugu, we identified around 20 annotation issues, while Tamil and Malayalam had approximately 15 and 10 errors, respectively. The most common issues were span boundary mismatches and fragmented entities. For example, Telugu expressions like '*2020 ḍisemḃar 10na*[3]' ('10th December 2020') were incorrectly split across multiple entity tags. These findings inform our analysis of the results in Section 5. Overall NER statistics are in Table 3.

Language	Sentences	Tokens
Kannada	1276	14560
Malayalam	1703	17643
Tamil	2132	38712
Telugu	2016	34179

Table 2: Statistics of the Manually Annotated NER Dataset

Dependency Parsing (DEP): We used in-house CoNLL-U data for Tamil. For Telugu, the limited in-house data was supplemented with the UD Dependency Treebank (Rama and Vajjala, 2017), though Telugu data remains comparatively smaller. Kannada and Malayalam data were sourced from (iiit, 2023) (CoNLL-U format), with Pāṇinian

[3]The Telugu text is transliterated using the ISO 15919 standard.

kāraka relations converted to UD relations as per (Tandon et al., 2016) (tagset in Appendix). Preprocessing involved filtering sentences with multiple root annotations and correcting formatting inconsistencies. Statistics are in Table 3.

Task	Lang.	Train (#Sent, #Tokens)	Test (#Sent, #Tokens)
MA	ka	(8311, 87197)	(1039, 10933)
	ml	(5666, 55901)	(708, 6794)
	ta	(14940, 169717)	(1896, 21748)
	te	(11737, 75610)	(1467, 9476)
POS	ka	(10680, 120123)	(1335, 14816)
	ml	(9549, 104470)	(1195, 12967)
	ta	(14940, 169717)	(1896, 21748)
	te	(18099, 120707)	(2263, 14873)
NER	ka	(6320, 114611)	(1000, 11407)
	ml	(6281, 91678)	(1000, 10467)
	ta	(10568, 175834)	(1083, 13590)
	te	(7164, 133842)	(1008, 16926)
DEP	ka	(9348, 97890)	(733, 7941)
	ml	(6311, 61842)	(799, 8044)
	ta	(5220, 69426)	(1000, 15373)
	te	(1051, 8140)	(838, 7100)

Table 3: Dataset statistics for MA, POS, NER, and DEP tasks.

Coreference Resolution (CR): Manually annotated data (Table 4, tool in Appendix) was augmented with synthetic data for training. Similar to NER, we translated Hindi coreference data (Mujadia et al., 2016) (275 documents, 3,523 sentences) and used Awesome-Align (Dou and Neubig, 2021) for mention mapping (Figure 1). Ten manually annotated documents per language form the test set. The Tamil test set has fewer sentences due to the nature of the source texts (short stories). The combined CR dataset statistics (manually annotated + synthetic data) are presented in Table 5, including the number of documents and sentences in each split, mentions per document, and the percentage of singleton mentions. To understand the quality of the projected coreference data, we did a manual error analysis on a sample. We found quite a few incomplete or inconsistent coreference chains-especially in Telugu and Tamil, where more than 45% of the sampled spans had annotation issues. These problems, likely due to translation and alignment errors, help explain the relatively modest improvements we report in Section 5.

4 Methodology

This study evaluates the effectiveness of multilingual learning for core NLU tasks in four Dravidian languages: Kannada, Malayalam, Tamil, and Telugu. For each of the five tasks-MA, POS, NER, DEP, and CR- we developed and compared two distinct modeling approaches:

Lang.	Docs	Sent	Mentions	%Sing
Kannada	43	1306	12.6/doc	0.2
Malayalam	15	698	19.2/doc	1.3
Tamil	72	2607	18.1/doc	1.4
Telugu	71	2127	16.2/doc	2.7

Table 4: Statistics of the manually annotated coreference resolution dataset

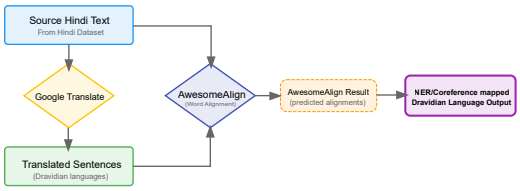

Figure 1: Workflow for Synthetic Data Preparation.

1. **Monolingual Models:** Trained independently on language-specific data for each of the four languages.

2. **Multilingual Baseline:** An additional baseline experiment, conducted using the IndicBERT (Doddapaneni et al., 2023) model, compares its performance with our proposed multilingual setup. For each task, IndicBERT is trained on the combined Dravidian dataset using the same model configuration as our multilingual model.

3. **Multilingual Model:** Trained on a combined dataset comprising data from all four languages for a given task. To ensure a fair comparison, the multilingual model for each task employs the same architecture as its monolingual models.

This comparative setup allows us to assess baseline language-specific performance and quantify the benefits derived from shared linguistic patterns utilized through multilingual training. All models were evaluated on their respective language-specific test sets as detailed in Section 3. The combined multilingual training data for each task was created by simple concatenation of the individual language training sets. The following subsections detail the model architectures and methods for each task.

Language	Train		Test		Avg.Mentions	%Sing
	# Docs	# Sent	# Docs	# Sent		
Kannada	225	3825	10	299	14/doc	5.2
Malayalam	198	3366	10	317	15/doc	7.6
Tamil	336	5712	10	154	12/doc	4.3
Telugu	253	4316	10	332	15/doc	5.3

Table 5: Combined Data Statistics of Coreference Resolution

4.1 Morphological Analysis (MA)

MA in our suite involves two sub-tasks: (1) root word and suffix identification, and (2) prediction of other morphological features lexical category, gender, number, and person.

Root Word and Suffix Identification: As a monolingual model, for this task, we employed a character-level lemmatization model using the Flair framework (Akbik et al., 2019), known for its effective character-level modeling capabilities and adapted specifically for Dravidian languages. To handle the agglutinative nature of Dravidian languages and to operate without pre-trained word embeddings for this sub-task, we developed a custom `FixedLemmatizer`. This involved subclassing Flair's Lemmatizer to handle tensor processing and enable direct character-based sequence generation. A character dictionary is built from the training corpus, encoding words as sequences of characters (including start and end symbols). The model uses a bidirectional RNN with two layers and a hidden size of 256 to predict root and suffix forms at the character level. This architecture served as the basis for both monolingual and multilingual models.

Multi-Task Learning for Morphological Features: For the other four morphological features, we employed a multi-task learning approach. This choice was motivated by the potential for these related features to benefit from shared representations. The architecture features a shared XLM-RoBERTa (Conneau et al., 2019) encoder, selected for its strong performance on multilingual and morphologically rich languages. The encoder captures the contextual meaning of input words, while task-specific classification heads predict individual morphological features. This setup enables the model to utilize shared linguistic patterns through common layers while refining predictions with specialized output layers for each feature. The same architecture is used in both monolingual and multilingual settings.

4.2 POS Tagging (POS)

For POS tagging, we utilize a sequence tagging framework built on Flair (Akbik et al., 2019), leveraging the XLM-RoBERTa (Conneau et al., 2019) model to generate multilingual embeddings pre-trained on a diverse set of languages. The model employs a first-last pooling operation, which combines information from the first and last transformer layers to capture both low-level and high-level linguistic features. These contextualized word representations are then passed to a linear tagging layer to predict UD POS tags for each token.

4.3 Named Entity Recognition (NER)

NER is also implemented as a sequence labeling task within the Flair framework (Akbik et al., 2019), taking advantage of its flexibility for sequence labeling tasks. Our approach employed a transformer-based sequence tagger built on pre-trained multilingual embeddings, fine-tuned specifically for NER. The model uses multilingual BERT (bert-base-multilingual-cased) (Devlin et al., 2019) to generate contextualized token representations, capturing both word-level and contextual information. This model is selected due to its widespread success in multilingual NER tasks. The tagger, configured with a hidden size of 256 and without a CRF layer, directly predicting BIO-formatted NER labels.

4.4 Dependency Parsing (DEP)

We built the parser using a Biaffine Dependency Parser, which relies on biaffine attention (Dozat and Manning, 2016), utilizing the implementation provided by the SuPar toolkit (Zhang et al., 2020). The Biaffine architecture is known for strong performance in dependency parsing. To help the model better understand the structure of Dravidian languages, we used multilingual BERT (bert-base-multilingual-cased) (Devlin et al., 2019) as the encoder. This setup allows the parser to make use of context-aware word representations, which is

especially helpful for handling the complexity of morphologically rich languages. The parser predicts head-dependent arcs and their corresponding labels.

4.5 Coreference Resolution (CR)

For coreference resolution, we extended the transformer-based end-to-end model proposed by Lee et al. (2017), designed to identify and link mentions referring to the same entity. The model uses a span-based architecture, encoding input documents with a multilingual BERT model (bert-base-multilingual-cased) (Devlin et al., 2019) to generate contextualized token representations. For each potential entity span, a representation is created by combining embeddings of boundary tokens, an attention-weighted representation of tokens within the span, and feature embeddings for span width. The model scores candidate spans to identify valid entity mentions, selects the top spans, and computes antecedent scores to link mentions to their references. We applied higher-order refinement using the attended-antecedent strategy (Lee et al., 2018b), where span representations are refined by attending to antecedents based on pairwise scores.

4.6 Training Details

Across all tasks, models were trained using appropriate optimizers (typically AdamW (Loshchilov and Hutter, 2017) or Adam) and learning rates suited to their architectures. For tasks leveraging Transformer-based models (MA-features, POS, NER, DEP, CR), fine-tuning was generally conducted for a number of epochs based on the task and dataset size, often incorporating early stopping where applicable. Common batch sizes ranged from 32 to 128. All experiments were conducted on NVIDIA L40S GPU. Detailed hyperparameters for each task-specific model, including precise learning rates, epoch counts, and optimizer configurations, are provided in Appendix.

5 Results and Analysis

This section presents model performance across all tasks for each language, evaluated using F1 scores (MA, POS, NER, CR) and Unlabeled and Labeled Attachment Scores (UAS and LAS). Results are shown in Table 6 and Table 7.

For Morphological Analysis (Table 6), our primary multilingual model performs better across most languages and features. The root+suffix identification task, being a character-level model, does not have a Transformer-based baseline, hence the empty cells for IndicBERT in the table. In both monolingual and multilingual settings, the root and suffix prediction task for Tamil recorded the highest score among all. For example, the multilingual model improved gender prediction in Tamil by 19.25% and person prediction in Kannada by 3.1%. These improvements show that the multilingual model benefits from shared patterns across the Dravidian languages, leading to better generalization. On the other hand, the lower root+suffix scores for Malayalam, especially the slight dip in the multilingual setting, as noted, potential inconsistencies in the annotation of its root forms which the multilingual model might be more sensitive to if other languages present clearer patterns.

Language	root+suffix	lcat	gender	number	person
Monolingual Model					
Kannada	94.37	69.94	61.66	52.45	67.24
Malayalam	79.31	63.36	40.80	58.32	57.26
Tamil	97.82	70.82	56.27	69.03	66.54
Telugu	96.39	83.36	81.75	72.70	75.29
Multilingual Baseline (IndicBERT)					
Kannada	-	48.23	45.54	52.48	53.08
Malayalam	-	41.47	42.43	49.91	43.57
Tamil	-	58.57	63.63	61.71	62.91
Telugu	-	58.57	63.63	61.71	62.91
Primary Multilingual Model					
Kannada	94.53	72.39	65.43	53.94	70.34
Malayalam	78.53	68.28	52.63	65.97	64.59
Tamil	97.83	75.12	75.52	75.28	76.42
Telugu	96.34	84.96	84.30	74.90	78.43

Table 6: F1 scores of Monolingual, Multilingual Baseline (IndicBERT), and primary Multilingual models on the test set for Morphological features.

The monolingual models perform well in POS tagging, with F1 ranging from 85.65 (Kannada) to 97.14 (Tamil), reflecting a strong understanding of word-level syntax. The multilingual model improves performance, with Kannada F1 rising from 85.65 to 93.88 and Tamil from 97.14 to 97.23.

NER remains challenging, especially for Kannada and Malayalam in the monolingual model (F1: 42.83, 38.65). The multilingual model improves F1 to 49.46 and 46.77, respectively, indicating better entity detection. Tamil also shows a slight gain (76.42 $\rightarrow$ 76.49), while Telugu exhibits a minor drop, possibly due to noise in synthetic data and annotation errors in projected entities (Section 3). These factors, along with underrepresentation in the base BERT model, this suggests the need for further investigation.

Dependency Parsing is evaluated using Unlabeled (UAS) and Labeled Attachment Scores

Language	POS	NER	CR	DEP	
	F1	F1	F1	UAS	LAS
Monolingual Model					
Kannada	85.65	42.83	44.26	91.15	81.11
Malayalam	93.22	38.65	31.89	87.67	72.87
Tamil	97.14	76.42	39.23	82.54	71.58
Telugu	96.71	74.18	25.67	79.87	65.62
Multilingual Baseline (IndicBERT)					
Kannada	89.97	32.57	2.17	84.43	70.97
Malayalam	90.87	42.41	1.77	74.74	59.68
Tamil	90.91	60.38	2.07	80.03	64.60
Telugu	92.72	62.67	0.06	71.65	50.06
Primary Multilingual Model					
Kannada	93.88	49.46	46.32	91.52	81.22
Malayalam	93.28	46.77	33.51	87.97	73.26
Tamil	97.23	76.49	45.31	82.65	71.64
Telugu	97.10	72.94	31.06	82.46	68.18

Table 7: Results of Monolingual, Multilingual Baseline (IndicBERT), and primary Multilingual models on the test set for POS, NER, CR, and DEP tasks.

(LAS). UAS measures correct head assignment, while LAS also checks the dependency label. The monolingual model achieves UAS above 79% (e.g., 91.15 for Kannada) with slightly lower LAS, indicating challenges in labeling relations. The multilingual model offers marginal gains (Kannada UAS: 91.15→91.52; LAS: 81.11→81.22), suggesting limited cross-lingual benefit due to syntactic variation.

For Coreference Resolution, the monolingual Model achieves F1 scores ranging from 25.67 (Telugu) to 44.26 (Kannada). The multilingual model improves these results, with F1 score for Kannada increasing to 46.32 and for Malayalam from 31.89 to 33.51. However, gains are modest, indicating that discourse-level relationships may require further refinement. Here, too, the multilingual model performs better than the monolingual model.

Comparison with IndicBERT Baseline. Our primary multilingual models consistently outperform the IndicBERT multilingual baseline across nearly all tasks and languages (Tables 6 and 7). For instance, in POS tagging, our primary multilingual model surpasses IndicBERT by approximately 3-4 F1 points across languages. The most striking difference appears in Coreference Resolution, where IndicBERT achieves near-zero F1 scores (0.06-2.17) while our model achieves scores from 31.06 to 46.32. This substantial performance gap is likely attributable to differences in model capacity. IndicBERT, based on an ALBERT-base (Lan et al., 2019) architecture with approximately 22 million parameters, is considerably smaller than the 179 million parameters of the mBERT encoder used in our primary CR model. Coreference Resolution is a complex discourse-level task requiring the modeling of intricate, long-distance relationships. The limited capacity of IndicBERT may be insufficient to capture these complex patterns, particularly when combined with the noisy synthetic data used for training.

The consistent outperformance of the multilingual model across all four Dravidian languages strongly validates our central hypothesis: utilizing shared linguistic patterns within this language family significantly enhances NLU capabilities. Notable improvements are observed in MA, POS Tagging, NER, and DEP. However, the gains for Coreference Resolution are more modest. This suggests that further refinements, such as expanding annotated training data, could help the model capture discourse-level relationships more effectively, leading to improved performance in coreference resolution and related tasks.

6 Conclusion and Future work

This paper introduces the Dravidian NLP suite and the corresponding Dravida dataset, providing comprehensive NLU tools for MA, POS, NER, DEP, and CR across Kannada, Malayalam, Tamil, and Telugu. Our key finding is the consistent and significant performance improvement achieved by multilingual models over their monolingual versions, demonstrating the effectiveness of leveraging shared linguistic characteristics within the same language family. This work underscores multilingual learning as an effective strategy for advancing NLU in linguistically related languages with varying degrees of resource availability. As part of our future work, we will increase the size and quality of annotated data for each language and task, and to potentially incorporate additional languages from the Dravidian family to our suite. We explore Large Language Models (LLMs) to develop a unified multilingual multitasking framework that jointly models all core tasks, enabling a single, robust model that can be seamlessly applied to a wide range of downstream applications across multiple Dravidian languages.

Limitations

This study presents several limitations that open avenues for future work in Dravidian NLU. While the Dravida dataset is a significant contribution, its size and diversity-particularly for discourse-level tasks like CR-remain limited compared to high-resource languages. This likely constrained CR performance in the multilingual setup and may hinder generalization to broader discourse phenomena. Synthetic data used for NER and CR helped address annotation scarcity but may introduce errors or miss important language-specific patterns, especially in entity and reference expression. Our MA covers suffixation and predefined features but does not fully address complex phenomena such as compounding or sandhi. The study is restricted to four major Dravidian languages, leaving out others in the family. Moreover, we did not benchmark against widely used multilingual toolkits like Stanza. Such a comparison would help clarify the trade-offs between broad multilingual systems and our targeted, family-specific models. Finally, observed performance fluctuations-such as in Telugu NER under the multilingual setting-highlight the need for a more systematic error analysis beyond the preliminary investigation reported in Section 3.

Acknowledgements

The authors thank the dedicated annotators of the research team for their contributions to the Dravida dataset, and the anonymous reviewers for their valuable feedback. We also acknowledge the use of AI tools for grammar checking.

References

Ajees A P, Manju K, and Sumam Mary Idicula. 2019. An improved word representation for deep learning based ner in Indian languages. *Information*, 10(6).

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics (demonstrations)*, pages 54–59.

R. Amritavalli and Bhuvana Narasimhan. *The Oxford Handbook of Dravidian Languages*. Oxford University Press.

PJ Antony and KP Soman. 2010. Kernel based part of speech tagger for Kannada. In *2010 International conference on machine learning and cybernetics*, volume 4, pages 2139–2144. IEEE.

PJ Antony and KP Soman. 2011. Parts of speech tagging for Indian languages: a literature survey. *International Journal of Computer Applications*, 34(8):0975–8887.

Sankalp Bahad, Pruthwik Mishra, Karunesh Arora, Rakesh Chandra Balabantaray, Dipti Misra Sharma, and Parameswari Krishnamurthy. 2024. Fine-tuning pre-trained named entity recognition models for Indian languages. *arXiv preprint arXiv:2405.04829*.

Akshar Bharati and Rajeev Sangal. 2007. Computational paninian grammar framework. *Supertagging: Using Complex Lexical Descriptions in Natural Language Processing*, 355.

Census. 2011. Census of India: Language. Accessed: March 15, 2025.

Aditi Chaudhary, Antonios Anastasopoulos, Zaid Sheikh, and Graham Neubig. 2021. Reducing confusion in active learning for part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 9.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(1):1–34.

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.

Sobha Lalitha Devi, Vijay Sundar Ram, and Pattabhi RK Rao. 2024. End to end multilingual coreference resolution for Indian languages. In *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pages 256–259.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2023. Towards

leaving no Indic language behind: Building monolingual corpora, benchmark and models for Indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12402–12426, Toronto, Canada. Association for Computational Linguistics.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. *arXiv preprint arXiv:2101.08231*.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Charles F Hockett. 1954. Two models of grammatical description. *Word*, 10(2-3):210–234.

iiit. 2023. Dependency parsing for 7 Indian languages. https://ssmt.iiit.ac.in/dependency_parsing. Last retrieved on Feb, 2025.

Ayyoob Imani, Silvia Severini, Masoud Jalili Sabet, François Yvon, and Hinrich Schütze. 2022. Graph-based multilingual label propagation for low-resource part-of-speech tagging. *arXiv preprint arXiv:2210.09840*.

Antony Alexander James and Parameswari Krishnamurthy. 2025. Pos-aware neural approaches for word alignment in dravidian languages. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 154–159.

Dan Jurafsky. 2000. *Speech & language processing*. Pearson Education India.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2832–2838.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sabrina J Mielke, Arya D McCarthy, Sandra Kübler, and 1 others. 2018. Unimorph 2.0: universal morphology. *arXiv preprint arXiv:1810.11101*.

Prudhvi Kosaraju, Sruthilaya Reddy Kesidi, Vinay Bhargav Reddy Ainavolu, and Puneeth Kukkadapu. 2010. Experiments on Indian language dependency parsing. *Proceedings of the ICON10 NLP Tools Contest: Indian Language Dependency Parsing*, pages 40–45.

Parameswari Krishnamurthy and Kengatharaiyer Sarveswaran. 2021. Towards building a modern written Tamil treebank. In *Proceedings of the 20th International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2021)*, pages 61–68.

B. Krishnamurti. 2003. *The Dravidian Languages*. Cambridge Language Surveys. Cambridge University Press.

Sanjeev Kumar, Preethi Jyothi, and Pushpak Bhattacharyya. 2024. Part-of-speech tagging for extremely low-resource Indian languages. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 14422–14431.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. *arXiv preprint arXiv:1707.07045*.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018a. Higher-order coreference resolution with coarse-to-fine inference. *arXiv preprint arXiv:1804.05392*.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018b. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana. Association for Computational Linguistics.

Zuchao Li, Jiaxun Cai, Shexia He, and Hai Zhao. 2018. Seq2seq dependency parsing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3203–3214.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

S Menaka, Vijay Sundar Ram, and Sobha Lalitha Devi. 2010. Morphological generator for tamil. *Proceedings of the Knowledge Sharing event on Morphological Analysers and Generators (March 22-23, 2010), LDC-IL, Mysore, India*, pages 82–96.

Pruthwik Mishra, Vandan Mujadia, and Dipti Misra Sharma. 2024. Multi task learning based shallow parsing for Indian languages. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(9):1–18.

Vandan Mujadia, Palash Gupta, and Dipti Misra Sharma. 2016. Coreference annotation scheme and relation types for Hindi. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 161–168, Portorož, Slovenia. European Language Resources Association (ELRA).

Rudra Murthy, Mitesh M Khapra, and Pushpak Bhattacharyya. 2018. Improving ner tagging performance in low-resource languages via multilingual learning. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(2):1–20.

Sneha Nallani, Manish Shrivastava, and Dipti Misra Sharma. 2020. A fully expanded dependency treebank for Telugu. In *Proceedings of the WILDRE5–5th Workshop on Indian Language Data: Resources and Evaluation*, pages 39–44.

Bureau of Indian Standards. 2021. Linguistic resources - pos tag set for Indian languages guidelines for designing tagsets and specification. Technical Report IS 17627:2021, Bureau of Indian Standards.

K Parameshwari. 2011. An implementation of apertium morphological analyzer and generator for Tamil. *Parsing in Indian Languages*, 41.

Siddhesh Pawar, Pushpak Bhattacharyya, and Partha Talukdar. 2023. Evaluating cross lingual transfer for morphological analysis: a case study of Indian languages. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 14–26.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2089–2096, Istanbul, Turkey. European Language Resources Association (ELRA).

Anitha S Pillai and L Sobha. 2013. Named entity recognition for Indian languages: A survey. *International Journal*, 3(11).

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. *arXiv preprint arXiv:2003.07082*.

P Rahothvarman, Adith John Rajeev, Kaveri Anuranjana, and Radhika Mamidi. 2025. Bridge the gap: Multi-lingual models for ambiguous pronominal coreference resolution in south asian languages. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 104–114.

S Rajendran. 2009. A novel approach to morphological analysis for Tamil language.

Taraka Rama and Sowmya Vajjala. 2017. A Telugu treebank based on a grammar book. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 119–128.

G. Uma Maheshwar Rao, Amba P. Kulkarni, and Christopher M. 2011. A Telugu morphological analyzer. *International Telugu Internet Conference Proceedings*.

Kengatharaiyer Sarveswaran, Gihan Dias, and Miriam Butt. 2021. Thamizhi morph: A morphological parser for the Tamil language. *Machine Translation*, 35(1):37–70.

B Srinivasu and R Manivannan. 2018. Computational morphology for Telugu. *Journal of Computational and Theoretical Nanoscience*, 15(6-7):2373–2378.

Abishek Stephen J, Daniel Zeman, and 1 others. 2023. Universal dependencies for Malayalam. *The Prague Bulletin of Mathematical Linguistics*, (120).

Milan Straka and Jana Straková. 2017. Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies*, pages 88–99.

Juhi Tandon, Himani Chaudhry, Riyaz Ahmad Bhat, and Dipti Misra Sharma. 2016. Conversion from paninian karakas to universal dependencies for hindi dependency treebank. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 141–150.

Ramasamy Veerappan, PJ Antony, S Saravanan, and KP Soman. 2011. A rule based Kannada morphological analyzer and generator using finite state transducer. *International Journal of Computer Applications*, 27(10):45–52.

Hung yi Lee, Danqi Chen, Christopher D. Manning, Christopher Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, Noah A. Smith, Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, Xuezhe Ma, Zecong Hu, J. Liu, Nanyun Peng, and Graham Neubig. 2009. Dependency parsing. In *Encyclopedia of Artificial Intelligence*.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, and 43 others. 2017. CoNLL 2017 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.

Yu Zhang, Zhenghua Li, and Zhang Min. 2020. Efficient second-order TreeCRF for neural dependency parsing. In *Proceedings of ACL*, pages 3295–3305.

A Appendix

A.1 Morphological Analysis

A Morphological Analysis involves analyzing tokenized wordforms into their roots, lexical category, and other morphosyntactic information in terms of their constituent morphemes. The six fields include root and suffix, lexical category (lcat), gender (gen), number (num), and person (pers). The field for each attribute feature is fixed, and a comma is

used as a delimiter, as shown in Table 1. In case no value is given for a particular attribute, then the field is left blank. Each feature is associated with one or more values related to it. The values of a feature are called feature values. For Instance, the feature *gender* has three feature values as *male, female* and *neuter*.

The possible features and feature values of a category are given below in Table 8

A.2 POS tagging

The Bureau of Indian Standards (BIS) tagset and the Universal Dependencies (UD) tagset are two different sets of part-of-speech (POS) tags used for annotating text data. While both tagsets serve the same purpose, they have different categorizations and granularities. The BIS tagset is designed explicitly for Indian languages, taking into account their unique linguistic characteristics. On the other hand, the UD tagset is a more universal standard, designed to be applicable across multiple languages. This conversion Table 9 provides a mapping between the BIS tags and the corresponding UD tags, facilitating the transition between the two tagsets.

A.3 Dependency Parsing

The Paninian dependency framework is widely used for annotating Indian languages, especially for syntactic and semantic structure analysis. It offers a rich set of language-specific grammatical relations, etc., which are tailored for Indic linguistic structures. However, with the increasing adoption of Universal Dependencies (UD) as a cross-linguistic syntactic annotation standard, converting Paninian tags to UD tags becomes essential for broader compatibility and multilingual NLP research. Figure 2 shows an example of dependency annotation for a Tamil sentence, and Table 10 presents the mapping between Pāṇinian and Universal Dependency labels.

A.4 NER Tagset

NER is a fundamental task in Natural Language Processing (NLP) that involves identifying and categorizing named entities in unstructured text into predefined categories. Table 11 lists the NER tags used in our annotations, and Figure 3 shows the NER annotation tool.

A.5 Coreference Resolution annotation tool

Coreference resolution is the task of identifying all expressions that refer to the same entity within a discourse. Annotating coreference chains involves marking such referring expressions and linking them to form equivalence chains. The tool shown in Figure 4 shows the chains and their mentions. The right-side column of the tool shows the list of chains with their mention.

A.6 Hyperparameters and Training Details

This subsection provides key hyperparameter details for the models described in Section 4.6.

S.No.	Features	Feature values
1.	**Root**	lemma
	Suffix	Case Markers or Tense, Aspect and Mood markers
2.	**Lexical Category** (lcat)	Nouns (n) Verbs (v) Pronouns (pn) Adjectives (adj) Number words (num) Nouns of space and time (nst) Avyayas (avy)
3.	**Gender (gen)**	Masculine (m) Feminine (f) Neuter (n) Human (Feminine +Masculine) (mf) Non-masculine (Feminine+Neuter) (fn) Any gender (any)
4.	**Number (num)**	Singular (sg) Plural (pl) Any number (any)
5.	**Person (pers)**	First (1) Second (2) Third (3)

Table 8: Feature Values in Morph Analysis

Tag Name	BIS Tag	UD Tag
Noun	NN, NST, NNV	NOUN
Proper Noun	NNP	PROPN
Pronoun	PRP, PRF, PRL, PRC, PRQ	PRON
Determiner	DM, DMD, DMR, DMQ	DET
Verb	VM, VF, VNF, VINF, VNG	VERB
Auxiliary Verb	VAUX	AUX
Adjective	JJ, QTF, INTF, CL	ADJ
Adverb	RB	ADV
Adposition	PSP	ADP
Coordinating Conjunction	CC, CCD	CCONJ
Subordinating Conjunction	CCS, UT	SCONJ
Interjection	INJ	INTJ
Negation	NEG	PART
Numeral	QT, QTC, QTO	NUM
Symbol	SYM	SYM
Punctuation	PUNC	PUNCT
Unknown/Other	UNK, ECH, RD, RDF	X

Table 9: BIS POS Tagset to UD POS Tag conversion

Pāninian	**UD**
main	root
k1, k1u, k4a	nsubj
k1s	nsubj:pass
k2	obj
k4, k2s	iobj
k3, k5, k7, k7p, k7t, k7a, k2u, k2p, k2g, k4u, k5prk, k71, k7pu,k7u,ras, ras-k1, ras-k2, ras-NEG, ras-avy, r6-k1, r6-k2, jk1, mk1, pk1	obl
nmod, nmod__k1inv	nmod
r6	nmod:poss
nmod__adj	amod
pof, pof__cn	compound
lwg__rp	compound:prt
vmod, adv, rd, rsp	advmod
vmod_emph,jjmod__intf, nmod-emph	advmod:emph
mod__wq	advmod:wh
mod	acl
nmod__relc, rbmod__relc, jjmod__relc	acl:relcl
ccof	conj
lwg__psp	case
lwg__vaux, lwg__neg	aux
rs	appos
coref	expl
lwg__uh	discourse
rad	vocative
rh, rt, sent_adv	advcl
rh-neg	advcl:neg
rsym, rsym_eos	punct
interrogative	mark
lwg__psp_cont, pof_idiom	fixed
enm	list
jjmod	adjmod

Table 10: Pāninian to UD Conversion tags

Tag	Description	Example
NEP	Person names	Leonardo DiCaprio
NEL	Locations	Tokyo
NEO	Organization Names	Microsoft
NEAR	Artefacts	Golconda
NEN	Numbers	twenty million
NETI	Time, Day, & Date	24th January 2022
NEF	Facility	airports, ports, hospitals, financial institutions
NEMI	Miscellaneous	Designations/Posts, Language names, Award and competition names

Table 11: NER Tagset

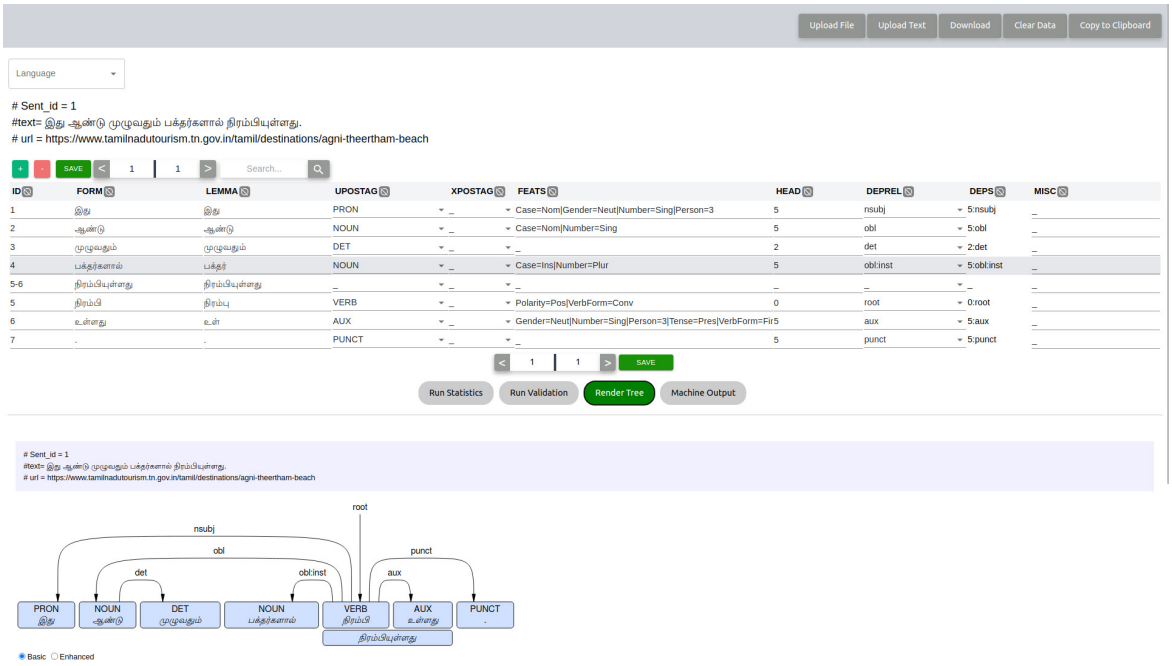

Figure 2: Dependency Parsing annotation tool

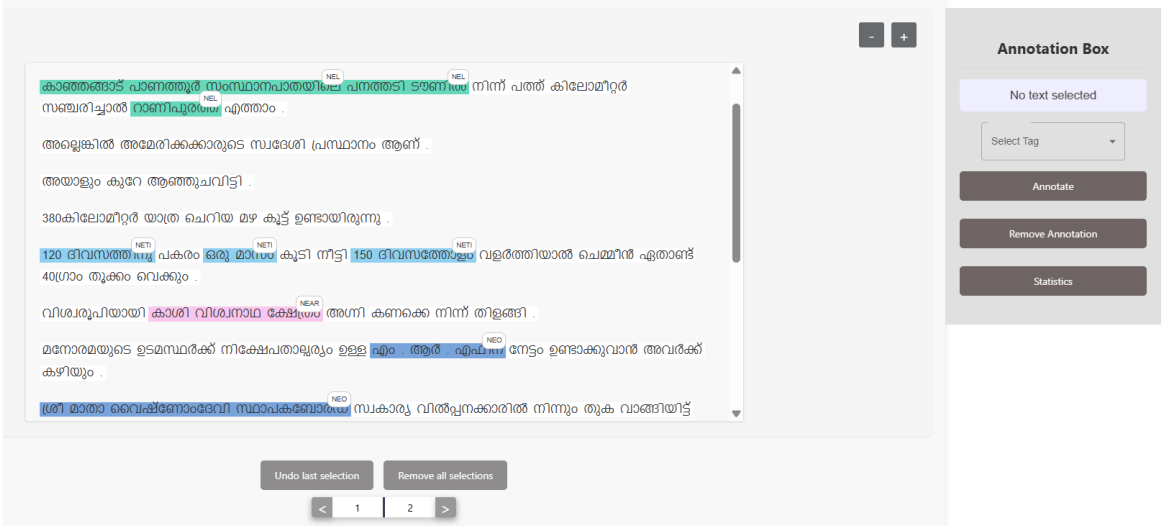

Figure 3: NER annotation Tool

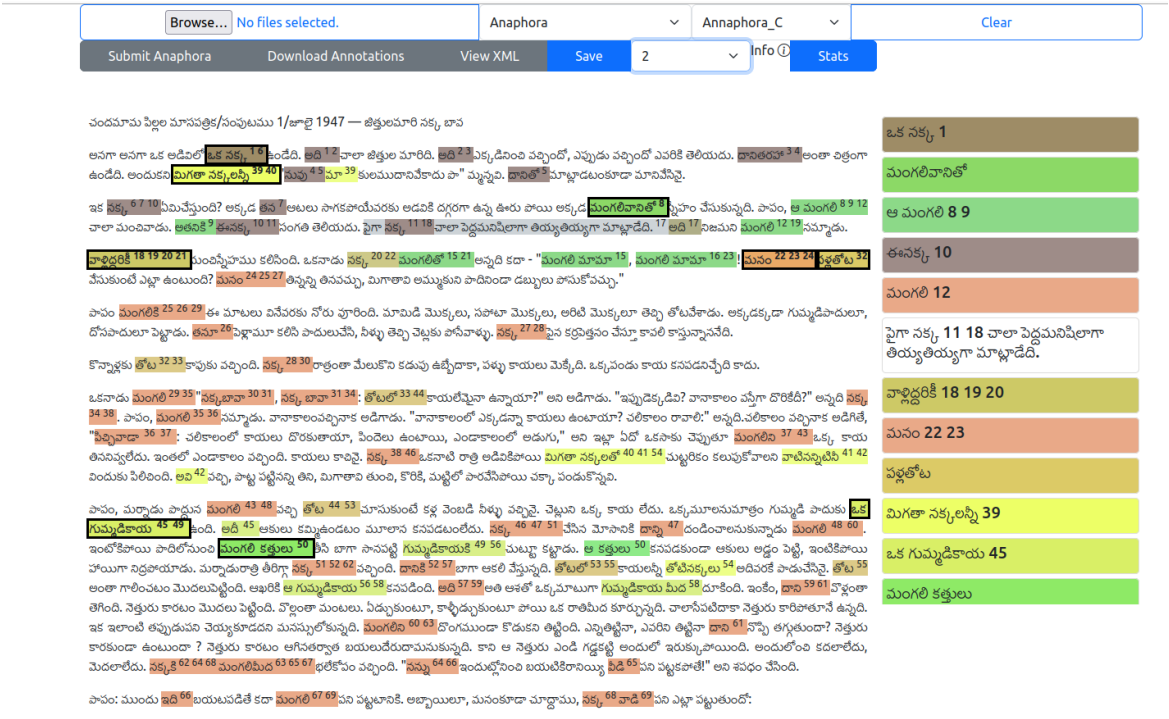

Figure 4: Coreference Resolution annotation Tool

Task	Model/Encoder	Learning Rate	Batch Size	Epochs	Other Parameters
Morphological Analysis	(Root/Suffix) Flair FixedLemmatizer (BiRNN)	0.1	32	50	RNN Layers: 2, Hidden: 256
	(Other features) XLM-RoBERTa-large	2×10^{-5}	64	20	Grad. Accum.: 2, Seq. Len.: 128
POS Tagging	Flair SequenceTagger (XLM-RoBERTa)	1×10^{-4}	128	30	Hidden: 128, Pooling: First-last
Named Entity Recognition	Flair SequenceTagger (mBERT)	3×10^{-5}	32	10	Hidden: 256, CRF: False, Weight Decay: 0.0
Dependency Parsing	SuPar Biaffine Parser (mBERT)	1×10^{-5}	64	15	LR Multiplier (non-BERT params): 20
Coreference Resolution	Extended Lee et al. (mBERT)	BERT: 1×10^{-5} Task: 2×10^{-4}	–	80	Grad. Accum.: 1, Seg. Len.: 512

Table 12: Hyperparameters and Training Details for NLP Tasks