Enhancing Telugu Part-of-Speech Tagging with Deep Sequential Models and Multilingual Embeddings

Sai Rishith Reddy Mangamuru, Sai Prashanth Karnati,
Bala Karthikeya Sajja, Divith Phogat, Premjith B
Amrita School of Artificial Intelligence, Coimbatore
Amrita Vishwa Vidyapeetham, India
b_premjith@cb.amrita.edu

Abstract

Part-of-speech (POS) tagging is a fundamental task in natural language processing (NLP) that involves assigning grammatical categories to words in a sentence. In this study, we investigate the application of deep sequential models for POS tagging of Telugu, a low-resource Dravidian language with rich morphology. We use the Universal dependencies dataset for this research and explore various deep learning architectures, including Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRUs), and their stacked variants for POS tagging. Additionally, we utilize multilingual BERT embeddings and indicBERT embeddings to capture contextual information from the input sequences. Our experiments demonstrate that stacked LSTM with multilingual BERT embeddings achieves the highest performance, outperforming other approaches and attaining an F1 score of 0.8812. These findings suggest that deep sequential models, particularly stacked LSTMs with multilingual BERT embeddings, are effective tools for POS tagging in Telugu.

1 Introduction

Human communication, a crucial part of everyday interaction, largely hinges on language as a vessel for idea conveyance, emotional expression, and information dissemination. In line with advancements in information and communication technology, there has been a surge in the need for proficient tools to interpret and analyze languages, particularly low-resourced languages. One such language is Telugu, a Dravidian language predominantly spoken in the Indian states of Andhra Pradesh and Telangana, in which the NLP research is still in the infant stages, even for fundamental tasks such as Part-of-Speech (POS) tagging (Eluri and Lingamgunta, 2019).

Among the various fundamental NLP tasks, POS tagging plays a crucial role (Church, 1992). By attributing grammatical categories to words, POS tagging allows computational systems to extract vital syntactic and semantic information from linguistic data. This task catalyzes numerous downstream applications, such as machine translation, text summarization, sentiment analysis, and information retrieval, underscoring the importance of accurate POS tagging (Shah and Bhattacharyya, 2002).

The significance of POS tagging and the lack of resources and state-of-the-art models in the Telugu language motivated us to study the application of deep learning algorithms to build POS tagging for Telugu. In this work, we endeavour to enhance the efficiency and preciseness of POS tagging for the Telugu language (Binulal et al., 2009), bolstered by the deep learning algorithms designed for the sequential data.

We used the dataset provided by the Universal dependencies [1] in this research. The sequential dependencies in the data were captured using sequential deep learning models - Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) networks, Gated Recurrent Unit (GRU) and their stacked variants. At the core of our methodology is BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018a), a contextual language model. BERT was used for generating the input representation for the Telugu words, which were fed into the model as input. The advantage of using BERT is that it operates by considering the entire context of a word within a phrase, efficiently encapsulating word dependencies. We used bert-base-multilingual-cased embeddings with this idea in mind so that it can improve the accuracy of the POS tagging. In this work, we considered multilingual BERT (Devlin et al., 2018b) and IndicBERT (Kakwani et al., 2020) to generate the embeddings for the Telugu words. From the experiments, it is observed that the Stacked LSTM with

[1] https://universaldependencies.org/te/index.html

multilingual BERT achieved the highest accuracy and F1 score.

The rest of the contents of the paper are explained in the following sections: Section 2 presents the Related Works. Dataset Description is provided in the Section 3. Section 4 explains the Methodology followed by the Results in section 5. The work is concluded in Section 6.

2 Related Works

While there have been POS taggers developed for Indian languages like Hindi, Bengali, and Tamil, there is no publicly available dataset other than the Universal dependency dataset, unlike several foreign languages such as Arabic, English, and various European languages, which have a more extensive range of POS taggers. It is especially noticeable in the case of low-resource Dravidian languages like Telugu. The unavailability of a huge gold-standard corps hindered the research in developing computational POS tag models for Telugu.

In (Antony and Soman, 2011), Antony P J and Soman conducted a comprehensive survey focusing on the evolution of various POS tagger systems and POS tagsets for Indian languages. Their analysis encompassed the existing methodologies employed in developing POS tagger tools. Their findings led to the conclusion that the majority of Indian language POS tagging systems currently in existence predominantly rely on statistical and hybrid approaches. A Malayalam morphological analysis (Premjith et al., 2018a) was conducted using the deep learning models to identify the morphemes automatically which obtained an accuracy of 98.16%.

In their study, Prabha et al. (Prabha et al., 2018a) described a sequence-to-sequence approach to model the problem using various deep learning algorithms, including Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) Networks, Gated Recurrent Units (GRU), and their bidirectional variations. It was observed that the bidirectional versions of RNN, LSTM, and GRU exhibited superior performance, with Bidirectional LSTM (Bi-LSTM) surpassing the rest by achieving an impressive accuracy rate of 92.66%. Sunil et al. (Sunil et al., 2012) conducted a comprehensive study on verb generation in Malayalam. Their research emphasized the importance of morphological analysis and synthesis, adopting a paradigm-based approach that considered various critical morphological features, including tense, transitivity, intransitivity, causativity, gerund, aspect, modality, voice, causative, transitive, intransitive, and non-finiteness. The study centred on 55 paradigm categories, effectively categorizing 6,700 verbs and facilitating their synthesis. Symbolic mapping rules were also integrated to enhance the synthesis process.

Sentence Text	POS Tags
చూసేరా అండీ ?	VERB PART PUNCT
ఎక్కడికి వెళ్తున్నారండీ ?	NOUN VERB PUNCT
ఎప్పుడోయ్ అమెరికా నించి రావటం ?	PRON PROPN ADP VERB PUNCT

Figure 1: A sample of the input word sequence and corresponding tag sequence

The paper (Visuwalingam et al., 2021) tackles Tamil POS tagging, a complex task given the scarcity of resources and agglutinative nature. It serves as a foundational step in NLP. The study deploys deep learning models like RNN, LSTM, GRU, and Bi-LSTM at the word level. Evaluation employs metrics such as precision, recall, F1-score, and accuracy, using a dataset comprising 32 tags and 225,000 Tamil words. The findings show that increasing the hidden state parameter enhances model performance, with Bi-LSTM featuring 64 hidden states achieving the highest accuracy (94%). (Premjith et al., 2018b) created a POS tagger based on deep learning for Sanskrit, one of India's oldest languages. They incorporated character-level features and employed diverse deep-learning algorithms to model the sequential relationships among characters. Their use of Bidirectional GRU resulted in an accuracy of 97.86%.

Greeshma Prabha et al. (Prabha et al., 2018b) introduced a model to address the challenge of part-of-speech tagging in Nepali using a variety of deep learning algorithms. These algorithms include the Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) Networks, Gated Recurrent Unit (GRU), as well as their bidirectional counterparts. Notably, the bidirectional variants of RNN, LSTM, and GRU demonstrated enhanced performance, with Bi-LSTM achieving remarkable results, boasting an accuracy rate of 92.66

The research paper titled "Parts of Speech Tagging for Kannada" by Swaroop L R et al. (L R et al., 2019) introduced a POS tagger for Kannada, a low-resource South Asian language. This POS tagger leverages Conditional Random Fields and incorporates unique features specific to the Kannada

language. These distinctive features encompass Sandhi splitting, a process that dissects compound words into their meaningful constituents. The model's performance is evaluated using a dataset comprising 21,000 sentences, resulting in an impressive peak accuracy of 94.56%.

3 Dataset Description

We utilized the Universal Dependency (UD) Telugu dataset, divided into three segments: training, validation, and testing. The training dataset consists of 1,051 sentences, the testing dataset contains 146 sentences, and the validation dataset contains 131 sentences. The maximum sentence length in the training dataset is 20 words. In the testing dataset, it is 14 words, and in the validation dataset, it is 11 words. The average sentence length across all three datasets is five words.

For our first experiment, a text-to-sequences model in which the input sequence is directly passed to the neural network, we combined the training, testing, and validation datasets into a single, large dataset to address out-of-vocabulary (OOV) token issues. However, for BERT models, we maintained the original dataset separation, as they do not encounter OOV token problems due to their pretraining on extensive data. The Figure 1 displaces the sample of the dataset used.

4 Methodology

The methodology of this research comprises four main stages: data sourcing and preprocessing, sequence generation, generating the embedding using BERT models and application of sequential deep learning networks. Figure 2 illustrates our methodology for developing the POS tagger. In the first stage, we collected the UD Telugu dataset, primarily developed for dependency tree creation. The dataset consists of dependency relationships and morphological information along with the POS tags.

The first process was to collect the words and the corresponding POS tags from the dataset. The dataset was prepared in CoNLL format, and we transformed it to sequence-to-sequence format. This format was chosen because the sequential model can take the input one at a time sequentially and generate the corresponding tag by considering the context into account.

In all the experiments, the first task was to tokenize the input sequence into tokens. In the first experiment, the tokens were transformed into an index and the index sequence was fed into the deep sequential models. In the other experiments, the words were converted into their vector representations before feeding them to the deep sequential models.

Tokenizing the input sequence into tokens was the initial step in every experiment involving the Text-to-Sequence model. After these tokens were converted into an index, the deep sequential models were given the index sequence. This method makes it possible for our models to comprehend the text's sequential structure, which makes it easier for them to learn about the relationships between words and the POS tags that belong to them.

In the subsequent stages, we deploy multiple deep learning models. Through complex computations, these models interact with the sequences to decode the linguistic aspects of Telugu texts. In parts of speech tagging applications, where word order and context are critical, RNN, LSTM, GRU and other designs to capture complex temporal correlations in textual data are advantageous.

By utilizing the advantages of these various models, we can obtain a more sophisticated understanding of the language, which paves the way for combining the multilingual BERT and IndicBERT models with RNN, GRU and other models. BERT offers bidirectional context-based insights by understanding words that appear before or after a word to assess the word's entire context. We incorporate IndicBERT, created for Indian languages, including Telugu, to achieve deep linguistic understanding. This model accurately represents the distinct nuances and intricate grammatical constructions of the Telugu language. These models are individually and collectively optimized for POS tagging in Telugu texts through experimentation and hyperparameter tuining, resulting in a model that can handle the complex grammar and language structure of Telugu with a considerable accuracy.

5 Experiments, Results, And Discussion

This section discusses the experiments conducted and the results of the various deep learning models to tag Telugu words with Text-To-Sequence, bert-base-multilingual-cased and Indic-Bert.

All algorithms are trained till 100 epochs and implemented early stopping with a patience setting of 3 during the training process. Adam optimizer is utilized for optimization purposes and used cat-

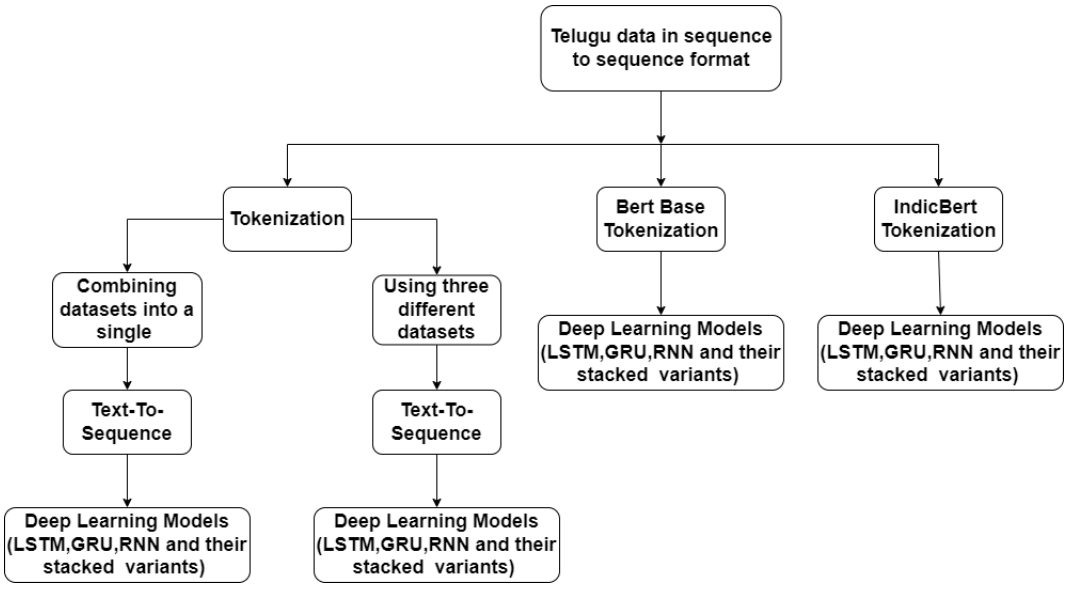

Figure 2: Methodology

egorical cross entropy as the loss function. For the stacked algorithms 2 layers of the respective algorithms are used in the architecture, and an embedding dimension of 300 is used.

Table 1 showcases the results of the text-to-sequence model. It is observed that RNN and its stacked variant stand out as top performers among all deep learning models. The stack RNN exhibited an accuracy of 87.30% and an F1 score of 0.8774, whereas the accuracy and F1 score of the RNN were 87.34% and 0.8735, respectively.

Tables 2 and 3 display the results of experiments conducted using BERT models.

In Table 2, a comprehensive analysis is presented, showcasing the performance of various neural network architectures when combined with the bert-base-multilingual-cased model. These deep learning models have collectively delivered strong results, with an average accuracy, precision, F1-score, and recall of 80%. Notably, the LSTM and Stacked LSTM stand out for their impressive performance, achieving an accuracy of 88.09% and 88.07%, respectively.

It can be observed from Table 2 that the stacked RNN and GRU models outperformed their non-stacked counterparts in POS tagging. This improvement can be attributed to the hierarchical nature of the stacked models, which allows them to capture more intricate dependencies in the input text. The stacked architectures facilitate the extraction of increasingly abstract features, leading to enhanced POS tagging accuracy.

Conversely, the stacked LSTM model exhibited a less favorable performance compared to its non-stacked counterpart. This outcome suggests that the LSTM's inherent architecture, characterized by its long-term memory retention, may not always align optimally with the requirements of POS tagging. The intricate long-term dependencies captured by the stacked LSTM may lead to a higher level of noise or irrelevant information for this specific task, affecting overall accuracy.

Table 3 provides an overview of the outcomes achieved through indic-bert for POS tagging. Among all the neural networks tested, stacked LSTM excels in comparison to the others. stacked LSTM stands out as it has achieved an accuracy of 73.54, along with precision, F1-score, and recall values of 0.7461, 0.7189, and 0.7170, respectively.

Table 4 presents the hyperparameter details for the top-performing algorithms within their respective models. In the case of the Text-To-Sequence model, the RNN and stacked RNN architectures stood out as the best options. The hidden layer sizes were set at 64 for RNN and 128 for stacked RNN, with learning rates of 0.0008 and 0.0007, respectively. For the bert-base-multilingual-cased model, the LSTM algorithms demonstrated superior performance. Specifically, for LSTM, a hidden layer of size 128 is utilized, while for Stacked LSTM, a hidden layer of size 128, with learning rates of 0.09 and 0.0007, respectively. In the case of Indic BERT, the top-performing models are "stacked LSTM"and "RNN", with learning rates of 0.005

Algorithm	Accuracy	Precision	Recall	F1 Score
RNN	87.34	0.8853	0.8546	0.8735
GRU	86.21	0.8634	0.8662	0.8606
LSTM	85.23	0.8378	0.8003	0.8403
Stacked RNN	87.30	0.8909	0.8449	0.8774
Stacked GRU	84.44	0.8709	0.8313	0.8338
Stacked LSTM	83.96	0.8349	0.8289	0.8319

Table 1: Performance scores of the Text To Sequence Model for Telugu POS tagging

Algorithm	Accuracy	Precision	Recall	F1 score
RNN	85.99	0.8718	0.8710	0.8629
GRU	87.10	0.8752	0.8682	0.8740
LSTM	88.09	0.8926	0.8765	0.8842
Stacked RNN	86.13	0.8636	0.8571	0.8618
Stacked GRU	87.93	0.8923	0.8834	0.8854
Stacked LSTM	88.07	0.8845	0.8890	0.8812

Table 2: Performance scores of the deep learning models with multilingual bert-base-multilingual-cased embedding for Telugu POS tagging

Algorithm	Accuracy	Precision	Recall	F1 score
RNN	72.56	0.7609	0.7170	0.7096
GRU	72.42	0.7226	0.7018	0.7167
LSTM	71.57	0.7437	0.7073	0.7002
Stacked RNN	70.65	0.7067	0.7156	0.7023
Stacked GRU	71.04	0.7189	0.7087	0.7076
Stacked LSTM	73.54	0.7461	0.7170	0.7189

Table 3: Performance scores of the deep learning models with indicbert-base-multilingual-cased embedding for Telugu POS tagging

Model	Algorithm	Embedding dimension	Hidden layer Size	Learning Rate
Text-To-Sequence	RNN	300	256	0.0008
Text-To-Sequence	Stacked RNN	300	256	0.0007
bert-base-multilingual-cased	LSTM	300	128	0.006
bert-base-multilingual-cased	Stacked LSTM	300	128	0.006
indic-bert	Stacked LSTM	300	128	0.005
indic-bert	RNN	300	128	0.004

Table 4: Hyperparameters used for the Best Performing algorithm for models used

and 0.004 respectively.

Based on the results obtained, within the group of models including Text-To-Sequence, bert-base-multilingual-cased, and indic-bert, indic-bert displayed the least effective performance in comparison to the other models. Both the Text-to-Sequence and bert-base-multilingual-cased models demonstrate accurate classification of parts of speech tags. An example output from the bert-base-multilingual-cased model is illustrated in Figure 3.

6 Conclusion

In conclusion, the work proposed exemplifies the potential of multilingual models to enhance POS tagging for the Telugu language. By addressing the challenges faced by low-resourced languages,

```
sentence: రాము కమలకు పుస్తకం ఇచ్చేడు .
predicted tags: propn propn noun verb punct
actual tags : propn propn noun verb punct
```

Figure 3: An example showing the performance of the model predicting the POS tags in comparison with the original tags

we contribute to the broader field of NLP and pave the way for more inclusive and efficient communication technologies. Based on the experiments conducted the bert-base-multilingual-cased model has outperformed the other models.

As the NLP community continues to evolve, we look forward to further advancements in linguistic diversity and improved accessibility for all. By harnessing advanced deep learning techniques, we aspire to build upon the foundation established by this research, ensuring that linguistic inclusivity remains at the forefront of our shared objectives.

References

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.

G Sindhiya Binulal, P Anand Goud, and KP Soman. 2009. A svm based approach to telugu parts of speech tagging using svmtool. *International Journal of Recent Trends in Engineering*, 1(2):183.

Kenneth Church. 1992. Current practice in part of speech tagging and suggestions for the future. *In Honor of Henry Kucera*, pages 13–48.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018a. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018b. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Suneetha Eluri and Sumalatha Lingamgunta. 2019. Arpit: Ambiguity resolver for pos tagging of telugu, an indian language. *i-Manager's Journal on Computer Science*, 7(1):25.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of EMNLP*.

Swaroop L R, Rakshith Gowda G S, Sourabh U, and Shriram Hegde. 2019. Parts of speech tagging for Kannada. In *Proceedings of the Student Research Workshop Associated with RANLP 2019*, pages 28–31, Varna, Bulgaria. INCOMA Ltd.

Greeshma Prabha, PV Jyothsna, KK Shahina, B Premjith, and KP Soman. 2018a. A deep learning approach for part-of-speech tagging in nepali language. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1132–1136. IEEE.

Greeshma Prabha, P.V. Jyothsna, K.K. Shahina, B. Premjith, and K.P. Soman. 2018b. A deep learning approach for part-of-speech tagging in nepali language. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1132–1136.

B Premjith, KP Soman, and M Anand Kumar. 2018a. A deep learning approach for malayalam morphological analysis at character level. *Procedia computer science*, 132:47–54.

B Premjith, KP Soman, and Prabaharan Poornachandran. 2018b. A deep learning based part-of-speech (pos) tagger for sanskrit language by embedding character level features. In *FIRE*, pages 56–60.

Chirag Shah and Pushpak Bhattacharyya. 2002. A study for evaluating the importance of various parts of speech (pos) for information retrieval (ir). In *Proc. International Conference on Universal Knowledge and Languages (ICUKL)*.

R Sunil, Nimtha Manohar, V Jayan, and KG Sulochana. 2012. Morphological analysis and synthesis of verbs in malayalam. *ICTAM-2012*.

Hemakasiny Visuwalingam, Ratnasingam Sakuntharaj, and Roshan G Ragel. 2021. Part of speech tagging for tamil language using deep learning. In *2021 IEEE 16th International Conference on Industrial and Information Systems (ICIIS)*, pages 157–161. IEEE.