Transformer-based Context Aware Morphological Analyzer for Telugu

Priyanka Dasari [1*], Abhijith Chelpuri [1*], Nagaraju Vuppala,
Mounika Marreddy, Parameswari Krishnamurthy, Radhika Mamidi
International Institute of Information Technology, Hyderabad, India
(priyanka.dasari, abhijith.chelpuri)@research.iiit.ac.in

Abstract

This paper addresses the challenges faced by Indian languages in leveraging deep learning for natural language processing (NLP) due to limited resources, annotated datasets, and Transformer-based architectures. We specifically focus on Telugu and aim to construct a Telugu morph analyzer dataset comprising 10,000 sentences. Furthermore, we assess the performance of established multi-lingual Transformer models (m-Bert, XLM-R, IndicBERT) and mono-lingual Transformer model BERT-Te (trained from scratch on an extensive Telugu corpus comprising 80,15,588 sentences). Our findings demonstrate the efficacy of Transformer-based representations pre-trained on Telugu data improved the performance of the Telugu morph analyzer, surpassing existing multi-lingual approaches. This highlights the necessity of developing dedicated corpora, annotated datasets, and machine learning models in a mono-lingual setting. Using our dataset, we present benchmark results for the Telugu morph analyzer achieved through simple fine-tuning on BERT-Te. The morph analyzer dataset [1] and codes are open-sourced and available here.

1 Introduction

A Morphological Analyzer is a valuable tool in natural language processing (NLP) that analyzes words by breaking them down into constituent morphemes. It provides crucial grammatical information, including gender, number, person, case markers (GNP), tense-aspect-modal information, and other linguistic features, which are indispensable features for understanding the morphology of a given language(Rao and Kulkarni, 2006). Many agglutinative languages have these grammatical features as part of their words. Hence, building a morph analyzer that parses and provides this information is important. This expansion would greatly benefit various NLP tasks and applications tailored to Indian languages.

This work aims to develop a Transformer based context-aware morphological analyzer for Telugu. Telugu is known for its agglutinative nature, and various affixes were attached to the root word to convey different grammatical meanings. Nouns and pronouns in Telugu are inflected for gender, number, person and case markers, followed by clitics. Verbs exhibit extensive inflections based on tense, aspect, mood and agreement features such as gender, number, and person (GNP). Additionally, Telugu also uses productive derivational suffixes, where nouns are converted into an adjective by the addition of the suffix *-ayna* and an adverb by the addition of the suffix *-gā* and by the addition of the suffix *-aḍaM* allows verb roots to function as gerunds, thereby allowing for noun inflections (Krishnamurti and Gwynn, 1985).

The complexity of Telugu morphology necessitates a robust morphological analyzer, which plays a crucial role in various NLP applications such as speech synthesis, information extraction, information retrieval, and machine translation (Rao et al., 2011). A morphological analyzer takes a single word in isolation and provides all possible analysis. Although a word may have multiple valid analysis, when considering the context in which the word is used, often only one analysis is appropriate or meaningful. This is because the context helps determine the word's intended meaning, which can help narrow down the possible analysis. Traditionally, morphological analyzers for Indian languages have been rule-based. Still, there is a recent shift towards utilizing machine learning techniques to build computational models with the development of Transformer based models from scratch in Telugu (Marreddy et al., 2022a)

[1] https://github.com/parameshkrishnaa/Telugu-Morph-Dataset/

[1*] The first two authors contributed equally to the work.

Third Workshop on Speech and Language Technologies for Dravidian Languages,
pages 25–32, Varna, Bulgaria, Sep 7, 2023.
https://doi.org/10.26615/978-954-452-085-4_004

on 80,15,588 sentences. This shift leverages advancements in downstream NLP tasks in Telugu like named entity recognition (Duggenpudi et al., 2022), sentiment analysis, emotion identification, sarcasm detection (Marreddy et al., 2022b), clickbait detection (Marreddy et al., 2021), and summarization (Vakada et al., 2023). Also, we see it is important to explore and compare existing multi-lingual Transformer based language models like mBERT (Pires et al., 2019), IndicBERT (Kakwani et al., 2020) and XLM (Conneau et al., 2019) with Telugu Transfer models (monolingual setting) like BERT-Te for the low-resource language, Telugu.

The main contributions of this paper are as follows:

- Creation of annotated Telugu Morphological analyzer dataset of 10,000 sentences.
- We created the benchmark results for Telugu morphological analyzer.
- Extensive experimentation with available Telugu Transformers models and existing multi-lingual Transformer models.
- On our dataset, BERT-Te outperforms the existing multi-lingual Transformer models.

The rest of the paper is organized as follows: Section 2 discusses the review and comparison of existing approaches. Section 3 describes the preparation of the dataset. Section 4 provides an overview of the experiment and evaluation of approaches followed using the dataset to train different models on the Telugu morphological features. Sections 5 and 6 discuss the ethical statement, conclusion, and future work.

2 Related Work

This section discusses the related work on building a corpus for morphological analysis focusing on Indian languages, existing Telugu BERT models, and Multi-lingual models. We also review the various common approaches used to build morphological analyzers. In the case of many Indian languages, morphological analysis has traditionally used a rule-based approach. It can be helpful in linguistic research because they provide a framework for formal analysis and understanding of language structures and patterns.

A Telugu Morphological Analyzer (Rao et al., 2011) is an example of organizing a linguistic database and employing computing resources effectively. The accuracy and coverage of this morph analyzer is 95-97%. This work is based on the word and paradigm approach (Hockett, 1954). A set of morpho-phonemically different forms in their inflection and derivation processes are identified. The failure of the presence of a root word in the morphological dictionary decreases the accuracy of the morph analyzer because it cannot analyze the root word. So, it shows issues with the OOV (Out-of-Vocabulary) and is not a context-based-morphological analyzer.

(Sunitha and Kalyani, 2009) have discussed an unsupervised stemmer that provides information about various decomposition of the word inflected by many morphemes. Firstly, the given Telugu words are processed by the (TMA) Telugu rule-based morph analyzer (Rao et al., 2011). The unsupervised stemmer further processes unrecognized words by the TMA to identify the components of the stem.

(Sneha and Bharadwaja, 2013) discussed a simple framework for designing and building a Morph Analyzer for Telugu noun forms applying the Telugu orthographic rules set with Finite State Machine (FSM). (Srinivasu and Manivannan, 2018) created a computational morphological analyzer and generator for Telugu using Item and Process linguistic model and FSM as a computational algorithm. (Kanuparthi et al., 2012) developed Hindi derivational morphological analyzer with 22 derivational suffixes (Goyal and Lehal, 2008) to analyse the derivation patterns in Hindi. For Tamil, (Parameshwari, 2011) implemented the APERTIUM Morphological Analyzer and Generator by defining and specifying the relevant linguistic database required for their development. The paper additionally discusses the module's efficacy, coverage, and speed compared to large corpora. (Veerappan et al., 2011), implements the morphological analyzer and generator for Kannada based on a rule-based finite state transducer that includes suitable morphological feature information and well-written morphophonemic rules. Morphological Analyzer for Gujarati (Baxi et al., 2015) introduces a hybrid approach combining statistical, knowledge-based, and paradigm-based approaches is used to develop the Morph analyzer.

Using the paradigm-based inflectional system and finite state systems to represent the language modelling, (Bapat et al., 2010) developed a highly accurate morphological analyzer for Marathi. (Baxi and Bhatt, 2022) based on the unimorph schema or the Universal Dependency Framework with the dataset contains 16527 distinct Gujarati inflected words with their morphological segmentation and grammatical feature tagging information is annotated and evaluated using the baseline format. Deep neural network-based models have recently been widely employed for building morphological analyzers. (Premjith et al., 2018) study discusses the Malayalam morphological analysis as a character-level sequence labeling problem that has been achieved with deep learning architectures such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). The model was trained using a 128-embedding size. According to their results, GRU has the highest accuracy score. (Gupta et al., 2020) studied the performance of different composite neural models for Sanskrit morphological tagging. Using neural architecture. (Chakrabarty et al., 2016) built a lemmatizer for Bengali and studied how it performed on the problem of word sense disambiguation.

Transformer-based language models like BERT-Te are available in Telugu (Marreddy et al., 2022a) trained on 80,15,588 sentences. These representations resulted in downstream NLP tasks in Telugu like named entity recognition (Duggenpudi et al., 2022), sentiment analysis, emotion identification, sarcasm detection (Marreddy et al., 2022b), and clickbait detection (Marreddy et al., 2021), and summarization (Vakada et al., 2023).

3 Dataset Description

This section elaborates on the dataset that is used to build our Transformer-based morph analyzer. We used a collection of generic Telugu corpus of 10,000 sentences as the basis for our work. Detailed statistics of lexical categories of this dataset can be found in Table 1. In order to ensure the quality of the data, we performed various cleaning and normalization procedures on the raw text. This included tasks such as correcting spelling inconsistencies and errors. By carrying out these measures, we aimed to enhance the reliability and consistency of the dataset. To facilitate further processing, we used a tokenizer[2] for dividing the text into individual tokens and to identify sentence boundaries. The tokenizer takes raw text as an input and produces the output in Shakti Standard Format (SSF) format (Bharati et al., 2007).

Lexical Categories	**Types**	**Tokens**
Nouns	4817	24904
Verbs	1636	15648
Pronouns	163	5025
Adjectives	174	2000
Adverbs	70	1018
Number words	195	716
Nouns of space & time	27	52
avyayas (indeclinables)	405	5731
Total	7,487	55,094

Table 1: Statistics of lexical categories from the dataset.

We used the LT toolbox[3] version of Telugu Morph Analyzer, which is developed by (Rao et al., 2011). To identify the POS tags within sentences, we use an existing ILMT POS tagger (Bharati and Sangal, 2007). The POS tagger assists in determining the role of each word in the sentence. Telugu morphological analyzer generates multiple possible analysis for a given word. To select the most appropriate contextual morph analysis, we used the same technique as mentioned in (Krishnamurthy, 2019) wherein we used the POS tagger that selects the relevant POS tag based on the POS category of the word. POS tagger provides the POS tag for the word in context, and then we prune out the multiple analysis in the morph based on the POS tagger's output, if any. For example, if a word has multiple morph analysis with different lexical categories, such as a noun or a verb, the POS tagger selects the noun or verb analysis according to the pruning output. If the pruning module fails, we resort to the pick-one morph strategy that selects the first analysis as the output for the word. However, it should be noted that errors in POS tagging can lead to mistakes in selecting the correct morph, thereby affecting the contextual awareness of the words in a sentence. Manual validation is necessary to identify the errors in selecting context-aware morphological analysis. We have discussed

[2] https://github.com/nagaraju291990/sentence-tokenizer

[3] https://github.com/parameshkrishnaa/Telugu-Morph-lttoolbox

specific errors pertaining to wrong GNP marking, incorrect case-marking, *sandhi* split errors and the like.

The following examples explicate the errors:

1. Wrong POS tag leads to selecting the wrong lexical category.

   (1) vāṭi-ni **pagalu** aMtā mēp-āli
   They-ACC day all graze-HORT
   ‘They should be grazed all day.’

   In example (1), *pagalu* is wrongly tagged as VM (verb main) *pagalu* ’to break ’, instead of a noun (NN) *pagalu* ’day’. This lead to errors in further stages of processing.

2. Ambiguous words which show no difference in number, person, or direct/oblique differences.

   (2) madhya-lō baḍi mānēs-ina vāḷḷu
   middle-LOC school stop-REL they
   kūḍā **cēr-ā-ru**.
   also join-PST-3.PL
   ‘Those who left school in the middle also joined.’

   In example (2), the subject agreement of *cēr-ā-ru* ’to join’ can be analysed both as 2nd person (exclusive or honorofic pronoun (*mīru* ‘you’) and 3rd person. However, in the example (2), the subject *vāḷḷu* ’they’ is the third person pronoun that resolves the ambiguity. It is noted that the morph analyser fails to provide the correct analysis in such cases.

   (3) ilā **amma pani** kūḍā nā netti-na
   this-way mother-OBL work also I-GEN
   paḍ-iM-di.
   head-LOC fall-PST-3.PL.FN
   ’This way, mother’s work also fell on me.’

   In Telugu, not all nouns overtly show differences in direct and oblique case marking. One such example, as in (3), is noted in the corpus. The noun *amma* ‘mother’ here, when associated with another noun *pani* leading to a chunk *amma pani* ‘mother’s work’, does not show any change in form. This leads to an error in the marking of the case for *amma* ‘mother’. It is observed that the noun is marked with a direct case instead of an oblique case.

3. *sandhi* split errors

   Other common errors include the *sandhi* splitting errors. Telugu being rich in *sandhi*, requires a *sandhi* splitting module before morph analysis for appropriate marking of features. In some cases, *sandhi* splitter fails to split certain words as in (4), where *āḍavāḷḷaMdarikī* ’all women’ is not split. It should be split into *āḍavāḷḷu* & *aMdarikī* ’women’ & ’all; only then morph analyser provides an accurate analysis. *sandhi* splitting is also done manually. Consider the example:

   (4) **āḍavāḷḷaMdari-kī** ī śakti rāv-āli
   women+all-DAT this power get-HORT
   ’All women should get this power.’

To ensure the reliability of our dataset, we conducted extensive manual validation. Our analysis found that some words needed to be listed in the dictionary, resulting in the tool’s inability to analyze those words automatically. To solve this, we manually assigned paradigms to these Out-Of-Vocabulary (OOV) words, ensuring they could be processed effectively and provides the analysis. We made 34% of changes in the dataset due to pre-processing errors. This validation process played a crucial role in enhancing the accuracy and quality of the dataset, providing reliable results for our analysis. Continuous refining of the dataset through manual validation makes the development of transformer-based context-aware morphological analysis more accurate.

4 Methodology

We first obtain the sentences with morphological tags for each word in the sentences, and then we feed those sentences to our language model to refine it using this dataset. We segregate the words’ properties, such as lexical category, gender, and person, after acquiring the morphological tags for each word in the sentence before feeding them separately to the classifiers. In this section, we develop a comprehensive exploration of the different language models analyzed for the morph tag prediction study, elucidating their configuration

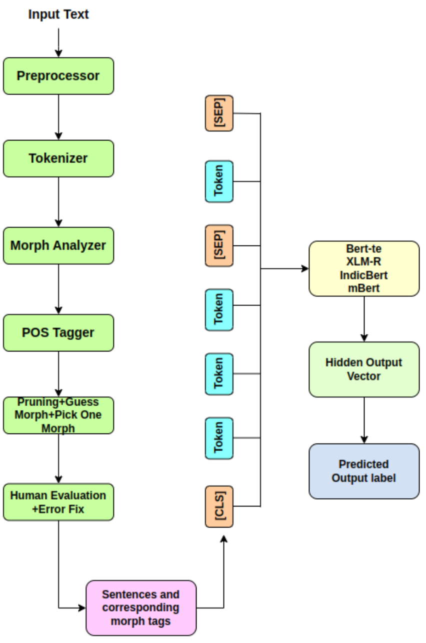

Figure 1: Flowchart of the work

in greater detail. In a later section of this article, we present results from language models that were examined in relation to several types of morphological tags. Figure 1 depicts the overall process flowchart.

4.1 Approaches

This section provides a description of the language models utilized, followed by an evaluation of their performances in the later section.

Bert-Te: Similar to the pre-trained BERT model introduced by (Devlin et al., 2019) in 2019, which is trained on the BooksCorpus and English Wikipedia, we opted for a model based on the Transformer architecture called BERT-basecased for Telugu. This Telugu variant of BERT is trained on a large corpus comprising 8 million sentences. The BERT-basecased model has 110 million parameters in total, 12 transformer blocks, 768 hidden layers, and 12 self-attention blocks.(Marreddy et al., 2022a) For the purposes of our investigation, we adjusted a BERT-Te model separately. We identified the following hyperparameters for fine-tuning the BERT-Te model to obtain optimal performance: (i) 64 batch size (ii) 3e-5 learning rate (iii) Number of training epochs: 30. To address the overfitting issue, we monitored the validation loss and stopped training if it did not decrease for five consecutive epochs.

IndicBERT: AI4Bharat, an AI research organization, has created a multilingual " IndicBERT " model that focuses on Indian languages and utilizes the BERT architecture(Kakwani et al., 2020). IndicBERT has undergone training on a large corpus of text originating from various Indian languages, including Hindi, Bengali, Tamil, and Telugu. This training enables the model to incorporate and understand these languages' unique linguistic characteristics and complexities. As a result, IndicBERT can comprehend and generate text within the context of multilingual Indian languages.

Multilingual BERT: Multilingual BERT (Bidirectional Encoder Representations from Transformers) is a variant of the BERT model that has been specifically trained on multilingual text data(Pires et al., 2019). This training enables the model to comprehend and generate text in multiple languages, making it a valuable tool for various multilingual natural language processing (NLP) tasks. Multilingual BERT has gained significant popularity within the NLP community.

The architecture of multilingual BERT closely resembles that of the original BERT model. It consists of a transformer-based neural network that utilizes self-attention mechanisms to capture contextual information from both the left and right contexts of each word in a given sentence. This mechanism allows the model to grasp the subtleties of language and the relationships between words.

During the training process of multilingual BERT, the model undergoes pretraining on an extensive corpus of text encompassing multiple languages. Throughout this pretraining phase, the model learns to predict missing words in sentences, which helps it develop a profound understanding of language structures and semantics. By training on a diverse range of languages, multilingual BERT can effectively capture cross-lingual information and transfer knowledge between different languages.

XLM-R: XLM-R (Cross-lingual Language Model - RoBERTa) is an advanced multilingual language model developed by Facebook AI. It is an extension of RoBERTa, which is itself a variant of the BERT model(Conneau et al., 2020). XLM-R has been specifically designed to excel in multilingual natural language processing (NLP) tasks and supports a wide array of languages.

The architecture of XLM-R is based on the transformer neural network, similar to

Model/Category	Bert-te			Indicbert			mBert			XLM-R		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Lexical category	0.642	0.821	0.602	0.951	0.706	0.702	0.623	0.822	0.581	0.720	0.688	0.576
number	0.906	0.77	0.738	0.821	0.703	0.607	0.819	0.781	0.680	0.726	0.74	0.590
person	0.820	0.771	0.704	0.665	0.554	0.475	0.796	0.78	0.690	0.672	0.770	0.570
gender	0.875	0.833	0.778	0.854	0.535	0.527	0.805	0.843	0.729	0.757	0.809	0.624
TAM/CM	0.588	0.78	0.515	0.549	0.503	0.409	0.575	0.696	0.564	0.720	0.530	0.527

Table 2: Precision,Recall and F1 scores for models tested

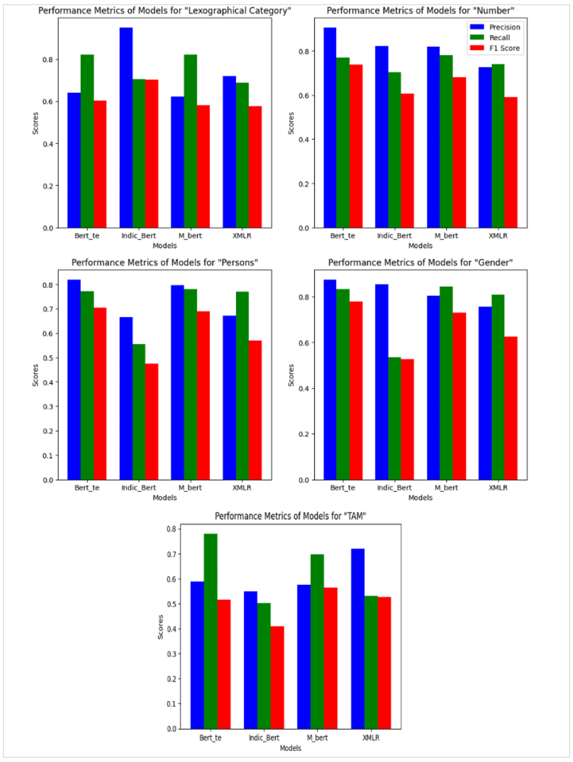

Figure 2: Comparision between all the different Models run for different tags.

BERT. It comprises multiple layers of self-attention mechanisms that effectively capture contextual information from the input text. This enables the model to comprehend the intricate relationships between words and sentences, and learn representations that accurately capture the semantics of the text.

One notable feature of XLM-R is its capability to align representations across various languages. By learning a shared representation space, XLM-R can proficiently transfer knowledge from high-resource languages to low-resource languages, even in scenarios where training data is limited. This makes XLM-R particularly valuable for multilingual transfer learning tasks, as it can utilize the knowledge acquired from one language to enhance performance in another.

4.2 Dataset Splitting

The dataset we used consists of 10,000 sentences, which we divided into two parts. The testing set accounts for 20 percent of the data, while the training set accounts for the remaining 80 percent. We evaluated the performance of the models mentioned earlier using the test data, and the precision, recall, and f1 scores obtained are presented in the following section.

4.3 Results

Precision, recall, and F1 score are common evaluation measures to gauge the performance of classification models. These metrics are derived by comparing a model's predictions with the actual labels assigned to the data. By providing valuable insights into the effectiveness of a classification model, these evaluation metrics assist practitioners in assessing and optimizing its utility. Below, we showcase the precision, recall, and F1 scores of the different models examined in this section.

In our study, for each category, we developed separate classifiers, and the performance of each classifier using various models is shown in Figure 2. We can see from the results (Table- 2) that Bert-te outperforms all other models in terms of F1 scores for the person (0.70), gender (0.77), and number (0.73) categories. Bert-te surpasses all other models in their respective categories with recall scores of 0.82 for lexical category, 0.77 for number, 0.83 for gender, and 0.77 for person. Bert-te has the greatest precision score in the person category (0.90), which completely outperforms all other models in that category. We discovered that our model, Bert-te, which is trained purely on Telugu language data, performs better than other multilingual models trained on various languages.

The Bert-te model is specifically able to understand the complexities and nuances peculiar to Telugu owing to the concentrated Telugu language instruction. As a result, when compared to the more broad multilingual models, it exhibits improved performance in Telugu language-related tasks.

This result emphasizes the value of domain-specific training and shows that optimizing models for a particular language can improve performance in tasks requiring that language. The Bert-te model's ability to outperform other multilingual models demonstrates the value of specialized language instruction in generating superior outcomes.

5 Ethical Statement

We created a dataset for the Telugu Morph Analyzer and open source the dataset [4]. The codes can be downloaded from here. We reused publicly available Telugu Transformer models (BERT-Te) to compare with existing multi-lingual Transformers models (IndicBERT, XLM-R, mBERT).

6 Conclusion and Future Work

Understanding the structure of individual words is made easier by morphological analysis. In terms of morph information, we have produced a trustworthy dataset. Various NLP tasks can now use this dataset. With the aid of the Morph Analyser, language models can effectively learn and utilize the additional details provided, enabling them to make more accurate predictions, generate more coherent and contextually appropriate responses, and better comprehend the subtleties of human language. By leveraging the insights from the Morph Analyser, language models become more efficient at processing and utilizing the available information, leading to improved language processing capabilities and more refined language generation.

References

Mugdha Bapat, Harshada Gune, and Pushpak Bhattacharyya. 2010. A paradigm-based finite state morphological analyzer for marathi. In *Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing*, pages 26–34.

Jatayu Baxi and Brijesh Bhatt. 2022. Gujmorph-a dataset for creating gujarati morphological analyzer. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 7088–7095.

Jatayu Baxi, Pooja Patel, and Brijesh Bhatt. 2015. Morphological analyzer for gujarati using paradigm based approach with knowledge based and statistical methods. In *Proceedings of the 12th International Conference on Natural Language Processing*, pages 178–182.

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

Akshar Bharati, Rajeev Sangal, and Dipti M Sharma. 2007. Ssf: Shakti standard format guide. *Language Technologies Research Centre, International Institute of Information Technology, Hyderabad, India*, pages 1–25.

Abhisek Chakrabarty, Akshay Chaturvedi, and Utpal Garain. 2016. A neural lemmatizer for bengali. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2558–2561.

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*.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Suma Reddy Duggenpudi, Subba Reddy Oota, Mounika Marreddy, and Radhika Mamidi. 2022. Teluguner: Leveraging multi-domain named entity recognition with deep transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 262–272.

Vishal Goyal and Gurpreet Singh Lehal. 2008. Hindi morphological analyzer and generator. In *2008 First International Conference on Emerging Trends in Engineering and Technology*, pages 1156–1159. IEEE.

Ashim Gupta, Amrith Krishna, Pawan Goyal, and Oliver Hellwig. 2020. Evaluating neural morphological taggers for sanskrit. *arXiv preprint arXiv:2005.10893*.

[4] https://github.com/parameshkrishnaa/Telugu-Morph-Dataset/

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

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. 2020. inlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 4948–4961.

Nikhil Kanuparthi, Abhilash Inumella, and Dipti Misra Sharma. 2012. Hindi derivational morphological analyzer. In *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, pages 10–16.

Parameswari Krishnamurthy. 2019. Development of telugu-tamil transfer-based machine translation system: An improvization using divergence index. *Journal of Intelligent Systems*, 28(3):493–504.

Bhadriraju Krishnamurti and John Peter Lucius Gwynn. 1985. *A grammar of modern Telugu*. Oxford University Press, USA.

Mounika Marreddy, Subba Reddy Oota, Lakshmi Sireesha Vakada, Venkata Charan Chinni, and Radhika Mamidi. 2021. Clickbait detection in telugu: Overcoming nlp challenges in resource-poor languages using benchmarked techniques. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Mounika Marreddy, Subba Reddy Oota, Lakshmi Sireesha Vakada, Venkata Charan Chinni, and Radhika Mamidi. 2022a. Am i a resource-poor language? data sets, embeddings, models and analysis for four different nlp tasks in telugu language. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(1):1–34.

Mounika Marreddy, Subba Reddy Oota, Lakshmi Sireesha Vakada, Venkata Charan Chinni, and Radhika Mamidi. 2022b. Multi-task text classification using graph convolutional networks for large-scale low resource language. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

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

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

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

G. Uma Maheshwar Rao and Amba P. Kulkarni. 2006. Computer applications in indian languages.

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

DL Sneha and K Bharadwaja. 2013. A novel approach for morphing telugu noun forms using finite state transducers. *IJERT*, 2(7):550.

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

KVN Sunitha and N Kalyani. 2009. A novel approach to improve rule based telugu morphological analyzer. In *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, pages 1649–1652. IEEE.

Lakshmi Sireesha Vakada, Anudeep Ch, Mounika Marreddy, Subba Reddy Oota, and Radhika Mamidi. 2023. Gae-isumm: Unsupervised graph-based summarization for indian languages. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

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.