Novel and Hybrid Context Aware Spell Correction Approaches For Gujarati Language Using Peter Norvig With GRU And IndicBERT

Brijeshkumar Y. Panchal[1,*] [https://orcid.org/0000-0002-9836-9927], Apurva Shah[2] [https://orcid.org/0000-0002-0264-9810]

[1,2]Computer Science and Engineering Department, Faculty of Technology and Engineering, The Maharaja Sayajirao University of Baroda, Vadodara, Gujarat, India.
[1]Computer Engineering Department, Sardar Vallabhbhai Patel Institute of Technology (SVIT)- Vasad, GujaratTechnological University (GTU), Anand, Gujarat, India.

[*]Corresponding Author: *panchalbrijesh02@gmail.com*

Abstract

Numerous applications in the domain of Natural Language Processing (NLP) rely on spelling and grammatical checks, including email, opinion mining, text summarization, chatbots, and countless more. An individual's credibility, cybersecurity efforts, legal ambiguities, and NLP application performance can all take a hit if they make a mistake when dealing with regional languages such as Assamese, Gujarati, Hindi, etc. In order to lessen the frequency of spelling errors, this article examines and concentrates on Gujarati. In addition to a thorough examination of issues related to the Gujarati language, this article provides up-to-date strategies for fixing spelling mistakes based on context of the word. A novel hybrid approach ensures top-notch Gujarati context aware spelling verification. Both approaches start with Peter Norvig's method, which uses a pre-set vocabulary to correct and verify words in a Gujarati text. Following careful consideration of all suggestions, GRU and IndicBERT will select the most appropriate one, taking into account the initial goal of contextual understanding and surrounding circumstances. After comparing the current method with the proposed one in terms of accuracy, efficiency, linguistic understanding, and methodology, it was found that IndicBERT-GUJBRIJAPU, a flexible grammar checking tool, provides more thorough, context-aware corrections with 93.49 % accuracy and 94.46 % precision. With 91.59 % recall, IndicBERT-GUJBRIJAPU found with superior ability to detect incorrect sentences while missing fewer compared to other methods.

Keywords: Natural Language Processing (NLP), Gujarati, Grammar checker, Spelling checker, IndicBERT, Peter Norvig, GRU

1. Introduction

India boasts a wealth of literature in a variety of regional languages, including Gujarati, Hindi, Tamil, Assamese and many more. For speakers of other languages inside the nation, these languages nevertheless can remain unintelligible. In languages such as Gujarati and Hindi, the context of a statement is quite important in imparting its intended meaning. Natural Language Processing (NLP) discipline seeks to apply grammatical principles and linguistic structures to analyze and comprehend natural languages. By means of natural language in both speech and text, NLP research investigates how computers might understand, analyze, and modify it, hence bridging the distance between humans and technology [1]. The term "language" denotes the natural languages including Gujarati, Hindi, and English in NLP.

Preprocessing, an essential part of natural language processing, involves examining the text for spelling errors in order to improve its quality by associating words with their accurate meanings. Across several sectors as shown in Figure 1.1, NLP finds extensive use including [2]:

- Machine translation involves translating text from one language into another; information extraction and analysis handles vast and challenging datasets.
- Spam Detection: Filtering unwelcome correspondence
- Fake News Detection: Finding misleading material on internet venues
- Sentiment Analysis: Evaluating public view of governmental initiatives
- Individualized Medicine: Examining medical records to customize treatments

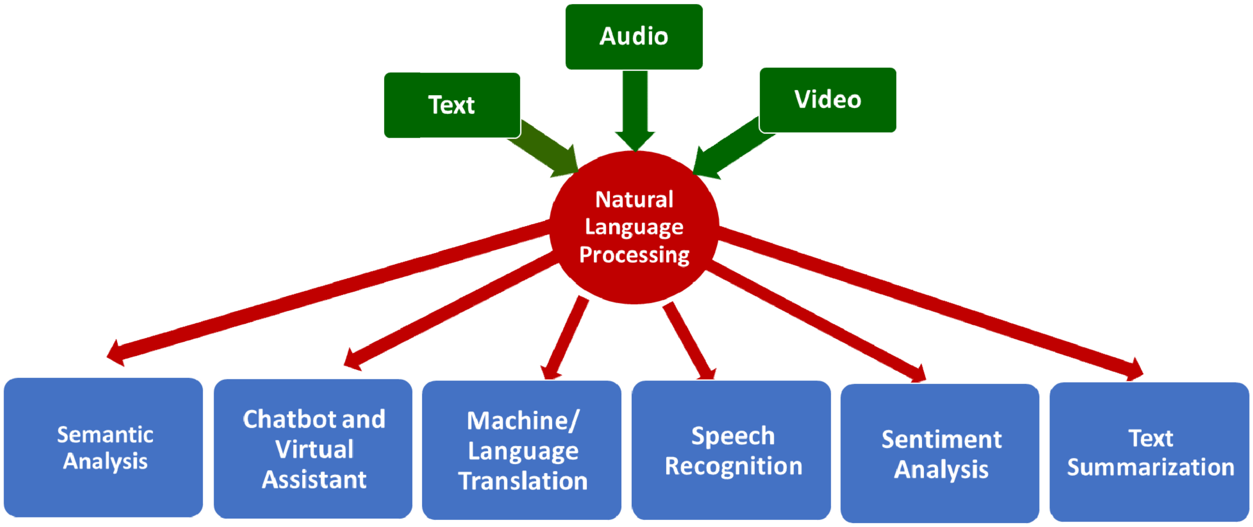
Audio
Text
Video
Natural Language Processing
Semantic Analysis
Chatbot and Virtual Assistant
Machine/ Language Translation
Speech Recognition
Sentiment Analysis
Text Summarization

Figure 1.1 Applications of Natural Language Processing (NLP)

Many NLP systems use grammar and spelling correction to remove textual data mistakes. These mistakes provide noise that influences syntactic and semantic understanding, therefore influencing the performance of NLP-based systems [3]. For spell-checking and grammar correction, Deep Learning is quite successful since it lets machines learn from past data. From SMS texting to social media (Facebook, Twitter, WhatsApp), text-based data is exploding exponentially and today digital government records and e-newspapers are expanding [4] [5]. Such large volumes of text need for effective NLP systems to manage several word forms, homographs, and metaphors. Essential components of text processing, spell-checkers help to fix written language mistakes [4]. They point out two main kinds of mistakes: Words not found in the dictionary (e.g., "Gujarti" rather than "Gujarati"). Non-word errors. Real-world Errors: Dictionary words used wrongly in context (e.g., "Their going to the market" instead of "They’re going to the market"). Although spell-checkers for Latin and Western languages have been extensively developed, the great linguistic variety and complicated grammatical structures of Indian regional languages mean that research on them is still in its early years.

Gujarati is kind similar to Hindi out of the Indo-Aryan branch of the Indo-European language family. Among the 22 officially recognized languages of India, it has over 55.5 million native speakers—4.5% of the nation's total population as per the 2011 census. Though some NLP tools [6] [7] [8] [5] [9] [10]—stemmers, lemmatizers, tiny corpora—exist—the language currently lacks thorough tools for spelling and grammatical correction [3]. Gujarati is widely used, although it lacks basic NLP tools—especially in relation to spell and grammar checking [11]. Available for Gujarati now, the "Saras" spell checker detects spelling mistakes using Directed Acyclic Word Graphs (DAWG). It does not, however, consider prefixes, suffixes, or inflections, therefore creating fresh research prospects for sophisticated spell-checking methods.

Gujarati grammar adhering to rigorous guidelines comprises [12]:

- Jodani (જોડણી) - Correct spelling guidelines
- Sandhi, (સંધિ) – Word joining rules
- Samas (સમાસ) is compound word building.

The aim of this study is to solve context aware spelling mistakes in Gujarati language. To solve the shortcomings of current spell-checkers and raise Gujarati NLP application accuracy, novel and hybrid approaches are developed and implemented. In this article, the challenges related to context aware spelling checking with Gujarati language is focused and reviewed which offers valuable insights for researchers, programmers, and language technology enthusiasts who are interested in improving current models. Using NLP methods and deep learning, the proposed models seek to:

- Handle grammatical norms and morphological variances;
- Improve error identification and correction for Gujarati text.
- Improve accuracy and precision for Gujarati.
- Improve Gujarati NLP applications' language processing efficiency

The upcoming session addresses the difficulties associated with Gujarati language spelling correction, exploring several methodologies including rule-based, statistical, and deep learning approaches to rectify spelling and grammatical problems in Gujarati. The subsequent lesson presents two models designed to identify and rectify context aware

spelling mistakes in Gujarati language sentences. The initial model employs Peter Norvig's algorithm and a Gated Recurrent Units (GRU) model, trained using a Gujarati word dictionary, to detect and activate errors while analyzing phrase context. Another model employs IndicBERT to finetune the Peter Norvig-based model, enhancing efficiency and accuracy by concentrating on omitted words inside the statement and assigning scores to forecast sentence correctness. The comparative analysis with respect to accuracy, precision, recall and F1 score for correct and incorrect statements with one of the existing tools has been covered in next section.

2. Related work and background theory

2.1. Characteristics and Challenges of Gujarati Language

Gujarati is a language that is rich in vocabulary. There are a number of inflections for adjectives, verbs, and nouns. The language has 12 vowels and 34consonants, as well as the ideas of matras and half consonants [13] [3]. Gujarati has several characters with practically same phonetic characteristics. Matras' sounds match those of a vowel: આ, ઇ, ઈ, ઉ, ઊ, એ, ઐ, ઓ, ઔ, અં, and અઃ.

All consonants possess inherent vowels. Furthermore, In Gujarati, vowels and constants can stand on their own or be accompanied by one or more matras, which are seen as distinct characters after the vowels and constants. A total of twelve possible word usages exists for each constant, as it is possible for it to appear with each of the eleven matras. In this manner, a total of 374 permutations of constant and matra are generated by combining all 34 constants with 11 matras [3]. Therefore, matras must be handled with due diligence during computer processing [13]. In the Gujarati language, the phonemes ◌િ (e) and ◌ી (ee) are identical, differing only in their degree of extension. The situation is analogous for ◌ુ (u) and ◌ૂ (oo). Consequently, words containing these characters are frequently misspelled. For instance, both **પૂજા** and **પુજા** are pronounced as '{pooja}' and signify ' worship'. Also, characters that sound similar could actually indicate something completely different. Consider the words {દિન} and {દીન}, which both mean day and poor, respectively.

According to one research [3], the average length of word in Gujarati language is much higher than English Language which increases the complexity of language. Gujarati uses zero-width characters, just like other Indian languages. Multiple instances of such characters put into a word will make identification by sight challenging. In contrast to English, prepositions such as in and to can take on suffix inflections within the word, and words can even have several inflections complicates the process of spelling error detection and correction. As a highly inflected language, it is challenging to compile all potential word forms in a lexical dictionary for a spelling checker for the Gujarati language.

2.2. Various spelling and grammar checking approaches

Creating a humanoid—the most intelligent machine ever—is the ultimate objective of artificial intelligence. An important consideration in this development process is the interaction between humans and computers for which the language tools developed that can comprehend communication languages across all technological dimensions need to be considered. Every one of the 'n' languages spoken today has its own unique set of laws and alphabet which necessitates the development of new and language specific tools for processing and deciphering a wide range of languages. To ensure that words are spelt correctly, the spell-checker consults the language's dictionary and lexicon. An easy way to understand a spellchecker is that it uses a spelling detector to look for words that are not in base form in a document and a spelling corrector to replace them with the most likely term from a database or corpus. Clusters based on grammar and speech components (nouns, pronouns, and adjectives) organize the dictionary.

Preprocessing, feature extraction, and modeling are the three primary steps that make up the Natural Language Processing (NLP) pipeline as shown in Figure 2.1 [14]. Every step of the process changes the text in some manner and generates a result that is needed by the following step. Sometimes, there are non-linear steps in the NLP pipeline. Going back and forth between the various stages is often essential in practice. For instance, if the modeling stage yields unsatisfactory results, it might be required to revisit the pre-processing or feature extraction stage in order to enhance the data's quality.

There are primarily three stages to spellchecking which include thorough pre-processing, spelling check, and creation of recommendation lists. Some of the steps involved in preprocessing include stemming, tokenization, and normalization [15] [16]. The spelling-checking module uses several dictionary lookup techniques to verify the candidate words' authenticity, while the recommendation list building module flags the list of possible suggestions for misspelled words. Suggestions are ranked by the spell-checker. This part ranks the ideas according to how necessary they are for the sentences. The primary stages of a spellchecker are as shown in Figure 2.2. After running the sentence through spellcheck, it returns a corrected version containing the correct term.

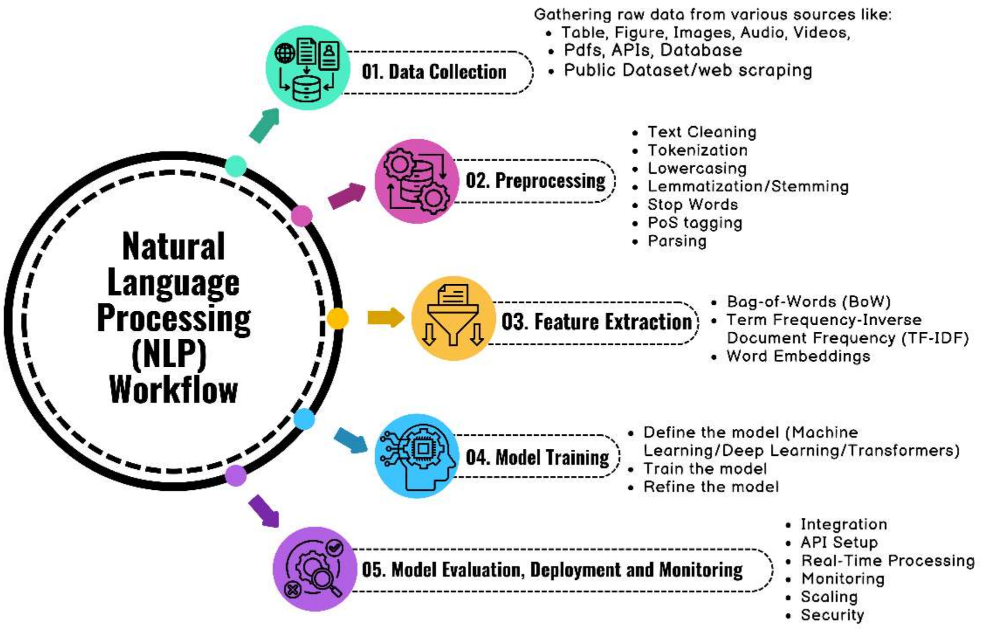

Figure 2.1 Natural Language Processing (NLP) workflow

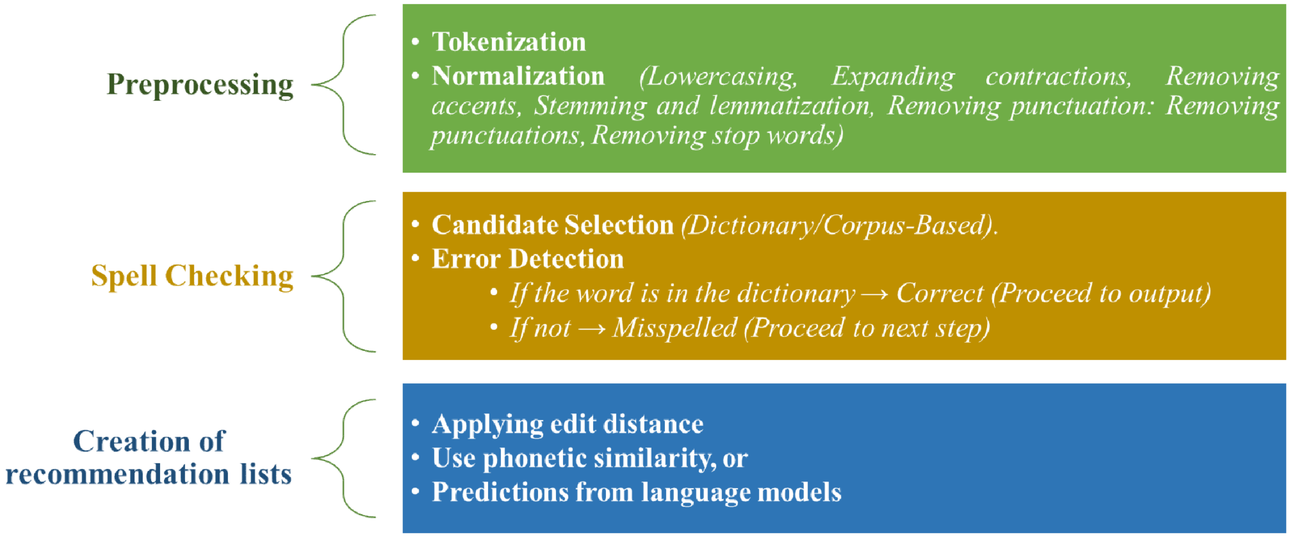
Preprocessing
• Tokenization
• Normalization (Lowercasing, Expanding contractions, Removing accents, Stemming and lemmatization, Removing punctuation: Removing punctuations, Removing stop words)
Spell Checking
• Candidate Selection (Dictionary/Corpus-Based).
• Error Detection
• If the word is in the dictionary → Correct (Proceed to output)
• If not → Misspelled (Proceed to next step)
Creation of recommendation lists
• Applying edit distance
• Use phonetic similarity, or
• Predictions from language models

Figure 2.2 The primary stages of Spelling checker using NLP

2.2.1. Syntax based

Each sentence obtains a parse tree created according to the base language's grammar. Text is inaccurate if full parsing fails. So, the parser should be as thorough as possible to reduce false alerts. The main advantage of this method is that the grammar checker will detect all errors if the grammar is complete and covers all possible syntactic rules. Due to natural languages' ambiguities, it's hard to list all their syntactic rules. So, the parser may give many parse trees, even for correct sentences. This approach just detects erroneous sentences. Nevertheless, additional rules that parse badly constructed sentences are needed to warn the user of the issue;

this method is referred to as constraint relaxation. If a statement can only be parsed using this additional rule, it is erroneous and a rule description and recommendation may be supplied.

2.2.2. Rule based

When checking for spelling errors, these systems use heuristics derived from various word properties, including as morphology, part-of-speech, stemming, and more [4]. One of the spell-checkers of the Assamese language uses morphology and dictionary search to detect and repair errors. One researcher has developed a Tamil spell-checker that uses morphological analysis to detect and rectify errors. Later, morphological analysis was used to propose many language spell-checkers. A rule-based spell-checker using part-of-speech (POS) tagging for English language spell-checking is also being used. Additionally, text chunkers were created using the Hidden Markov Model to improve spell-checker speed [4]. A POS-tagged text is compared to a set of established rules in the form of error patterns. If a pattern is found, the text is wrong. Patterns may be based on words, their POS tags, or chunk tags. This strategy is like the statistics-based one, but all the rules are made manually. Rule-based systems will never be done, unlike syntax-based ones. Even with many mistake rules, it's nearly difficult to predict every grammatical inconsistency, therefore some errors will be missed. It's better to miss some issues than to have a faulty parser create false warnings. This technology allows for individual rule activation and deactivation, and the system can give thorough error warnings and helpful comments, including grammar rule explanations. This technique allows for progressive system expansion by starting with one rule and adding more [17].

2.2.3. Statistical based

The ability to speak a specific language is not necessary to understand statistical procedures. Examples of spellcheckers that use word counts and word characteristics include those that are frequency-based, n-gram-based, and finite state automata-based [4]. The statistical method greatly enhances performance without requiring knowledge of the particular language, which is a major advantage. One problem with these approaches is that they rely on metrics like word count, frequency, and characteristics to do spellchecking, yet processing certain spelling mistakes necessitates familiarity with the target language. Many academics employed a combination of rule-based and statistical approaches to address this type of problem. To get past the problems, a hybrid model combines rule-based and statistical approaches [4] [18].

2.2.4. Deep learning based

Deep learning is the specialty of artificial neural networks, or ML algorithms. Deep learning algorithms have been widely employed and effective lately. Deep learning approaches' success is partly due to the freedom of architecture selection. Deep learning methods were used in ML research for natural language processing [17]. While the rule-based and statistical methods demonstrate significant effectiveness, the performance of spell-checking can be further improved through the application of deep-learning techniques. Specifically, the real-world errors that necessitate understanding the context of the word within the sentence demonstrate that these deep-learning methods are highly beneficial. When it comes to researchers using deep-learning techniques for error correction, Ghosh and Kristensson were pioneers. They put out a model for English text repair. The study into language processing through deep learning remains in its early stages. Regarding regional languages, the deep-learning-based spell-checker is currently available for the Malayalam and Tamil language which utilizes an LSTM network [19] [20]. This spellchecker involves a network that is both trained and tested to detect spelling errors and pinpoint their locations [4] [21] [22].

3. Comparative analysis of Spell checker for various regional languages of India

Natural language processing (NLP) depends much on spell and grammar checkers since they find and fix textual data mistakes. Although a lot of study has been done on English and other generally spoken languages, regional languages of India provide special difficulties because of their rich morphology, complicated phonetic structures, and different scripts. The spell-checking methods created for several Indian languages—including Hindi, Bengali, Tamil, Telugu, Gujarati, Dogri, Malyalam and Assamese—are compared in this work [23]. Examining several approaches including rule-based, statistical, hybrid, and deep learning-based spell-checkers, the paper assesses their performance in managing orthographic variants, phonetic mistakes, and real-word errors. Particularly languages with strong inflectional morphology, like Tamil and Telugu, call for more complex methods like sandhi-based and morphological analyzers; languages like Hindi and Bengali gain from hybrid approaches combining edit distance and phonetic algorithms [24] [6] [21] [25]. Emphasizing the importance of language-specific optimizations, the study shows the benefits and restrictions of every technique. Future lines of research include using transformer-based models such IndicBERT and GRU to better contextual spell-checking, hence improving accuracy over several languages. Through tackling these difficulties, our work hopes to help to create more strong and effective spell-checking systems for India's linguistically varied terrain.

Table 1 Comparison of various spell checker and grammar checker for low resource languages

Reference	Year	Language	Research Focus	Methodology /Approach	Key Findings & Accuracy
[24]	2002	Assamese	Dictionary-based spellchecker	Dictionary lookup, bigram search, Soundex code integration	Adequate results with over 5000 words, integration with Assamese-English dictionary in progress
[26]	2012	Kashmiri	Spellchecker development	Standalone application, non-word error correction	80% error detection, 85% correct suggestions, plans for real-word error handling
[25]	2013	Urdu	Spellchecker evaluation	Reverse edit distance method	High complexity (86n + 41 comparisons), needs improved methods for better accuracy
[27]	2015	Hindi	HINSPELL spellchecker	Error detection, repair, substitution	83.2% detection, 77.9% correction, future focus on grammatical errors
[28]	2015	Tamil	Morphological analyzer	Linguistic analysis, POS tagging	Efficiency between 60-97%, useful for NLP tasks like MT, lemmatization, parsing

[29]	2016	Kashmiri	Improved spellchecker	Standalone application, lexicon development	80% detection, 85% correct recommendations, integration with OpenOffice needed
[30]	2016	Tamil	Hybrid spellchecker	N-gram, stemming, tree-based algorithm	91% accuracy, tree-based method better for error detection
[31]	2016	Telugu	Spellchecker with sandhi analysis	Morphophonemic, external sandhi handling	Addresses Telugu’s complex linguistic features
[19]	2018	Malayalam	Deep learning-based spellchecker	LSTM neural networks, error detection & correction	Outperforms Unicode splitting, limited by computational resources
[32]	2019	Bengali	Spellchecker development	Hybrid of edit distance, Soundex	Adapts existing methods for better Bengali spell correction
[33]	2020	Multilingual	Comprehensive spellchecker review	Literature analysis, NLP methods (rule-based, statistical, deep learning)	Categorizes spellcheckers, compares performance across languages
[25]	2020	Tamil	Alternative spellchecking methods	Bloom-filter, Symspell, LSTM	Symspell is fast but lacks accuracy; LSTM is promising but underexplored
[3]	2021	Gujarati	Jodani spellchecker	Root word-based, Levenshtein distance	91.56% accuracy, plans to improve character assumption handling
[23]	2022	Dogri	Hybrid spellchecker	Hybrid methodology for detection & correction	First known attempt for Dogri spellchecking
[34]	2023	Gujarati	Enhancing ASR Performance with Improved Spell Corrector	Combination of MFCC and CQCC features, GRU-based DeepSpeech2 architecture, and enhanced spell corrector	Improved Word Error Rate by 17.46% compared to the model without post-processing
[11]	2024	Gujarati	Spell Checker	Implementati	Achieved 80–90%

			Using Norvig Algorithm	on of Norvig's algorithm with a dataset of 16,937 distinct Gujarati words	accuracy in identifying and correcting misspelled words

The table 2 highlights a diverse range of spell correction models designed for multilingual and low-resource languages. Models like IndicBERT, MuRIL, L3Cube-IndicSBERT, ArabicCorrectionCntxt and Amazon's real-time spell checker utilize context and probabilistic models to enhance accuracy, while Icelandic and Indic approaches explore morphological and linguistic challenges.

Table 2 Comparison of spell correction models designed for multilingual and low-resource languages

Model	**Language Coverage**	**Architecture**	**Error Handling (Morph., Contextual)**	**Dialect Support**	**Computational Efficiency**	**Novelty / Remarks**
IndicBERT [35]	12 Indic + English	ALBERT (Transformer-based)	Limited analysis; strong recall	Not evaluated	Efficient (ALBERT backbone)	First shared Indic ALBERT model
MuRIL (Google) [36]	17+ Indian Languages	Multilingual BERT variant	Better contextual handling (cross-lingual pretraining)	Partial	Moderate to high	Strong cross-lingual transfer
XLM-R [37]	100+ languages	RoBERTa-based	Strong contextual handling	Weak on Indic dialects	High resource requirements	Robust multilingual performance
Adapter-BERT for Indic [38]	Varies	BERT + Adapters	Task-specific error modeling possible	Extendable	High efficiency (modular)	Scalable low-resource adaptation

L3Cube-IndicSBERT [39]	10 Indic Languages	Multilingual SBERT	Enhanced sentence-level contextual embeddings	-	Efficient fine-tuning	First multilingual sentence representation model for Indic languages
ArabicCorrectionCntxt [40]	Arabic	Levenshtein + Context Probabilities	Handles contextual errors via paragraph-level keyword-based context detection	Not specified	Efficient (simple lexical + context)	Uses paragraph context and keyword frequency to re-rank corrections
Icelandic Contextual Spell Corrector [41]	Icelandic	ML Classifiers + Morphological Tags	Strong contextual disambiguation; affected by rich morphology	Not specified	Moderate (due to tag sparsity)	Contextual confusion-set disambiguation with lemmatized and PoS features
Context-Free ML Spell Corrector [42]	English (demonstrated); extendable	Supervised ML (e.g., Naive Bayes)	No context used; character/word/token-based input features	Not applicable	Efficient for standalone terms	Context-free, character-level input with multiple ML classifiers
Amazon Multilingual Spell Checker [43]	24 Languages (Indic included)	N-gram-based + SymSpell Ranking	Context-aware using n-gram conditional probabilities	No dialect handling	Real-time capable (optimized Trie)	Real-time, extendable to new languages via Wikipedia + subtitle corpora

4. Proposed GUJAPUBRIJ and GUJBRIJAPU Models

The models presented in this article for the Gujarati Language utilize Peter Norvig's spelling correction algorithm to verify spelling accuracy by segmenting the provided text into smaller units. The algorithm proposed by Peter Norvig addresses errors by employing probability and utilizing edit distance. The initial implementation of the Gujarati spell checker was based on Peter Norvig's methodology, employing a Gujarati lexicon. This method has laid a solid foundation for identifying and correcting spelling inaccuracies. This method is ideal for identifying and rectifying typos that include words. It performs effectively with a predefined dictionary; however, it encounters challenges in grasping context. To check the context of the text along with spelling, the first GUJAPUBRIJ model employs a Gated Recurrent Unit (GRU)-based neural network, while the second GUJBRIJAPU model utilizes IndicBERT. Both of these approaches monitor interdependencies in context to improve the spell checking process. These model names—GUJAPUBRIJ and GUJBRIJAPU—are derived from the researchers' names, APURVA and BRIJEHKUMAR. The models have been designated with these names by the researchers for novel identification purposes.

4.1. Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with GRU – GUJAPUBRIJ Model

Incorporating a neural network based on Gated Recurrent Units (GRUs) with Peter Norvig's spelling correction technique, this GUJAPUBRIJ model improves accuracy and contextual comprehension. The preprocessing framework was designed to handle both lexical and contextual correction of Gujarati sentences. At the lexical level, the Peter Norvig's probabilistic spell correction algorithm is adopted, tailored for Gujarati. A custom vocabulary, derived from a curated corpus of valid Gujarati words, served as the reference dictionary. For each token in the input text, the algorithm generated candidate corrections using character-level operations—insertions, deletions, substitutions, and transpositions—defined over a comprehensive set of Gujarati characters and matras. Only those candidates found in the reference vocabulary were retained to ensure linguistic validity.

Following lexical correction, the first step of this approach is to break the provided text into smaller units so that the word can be checked in the Peter Norvig dictionary. The text was tokenized using TensorFlow's Tokenizer, with an <UNK> token to manage out-of-vocabulary words. Tokenized sequences were padded to a uniform length using pad_sequences, preparing them for the GRU-based GUJAPUBRIJ model. The GUJAPUBRIJ model will choose the top five elements with edit distant 1 or 2 if the word is going to be wrong. Before sending the tokenized word to GRU for spelling and context checks, it will be represented numerically. This GRU network, trained on sequences of words, captured contextual dependencies and enabled selection of the most contextually appropriate corrections from among the candidate words. Thus, the preprocessing pipeline ensured both orthographic normalization and semantic coherence before feeding data to the GUJAPUBRIJ model.

The proposed GUJAPUBRIJ novel model by Peter Norvig and the Gujarati text grammar and spelling checker based on GRU are depicted in Figure 4.1.

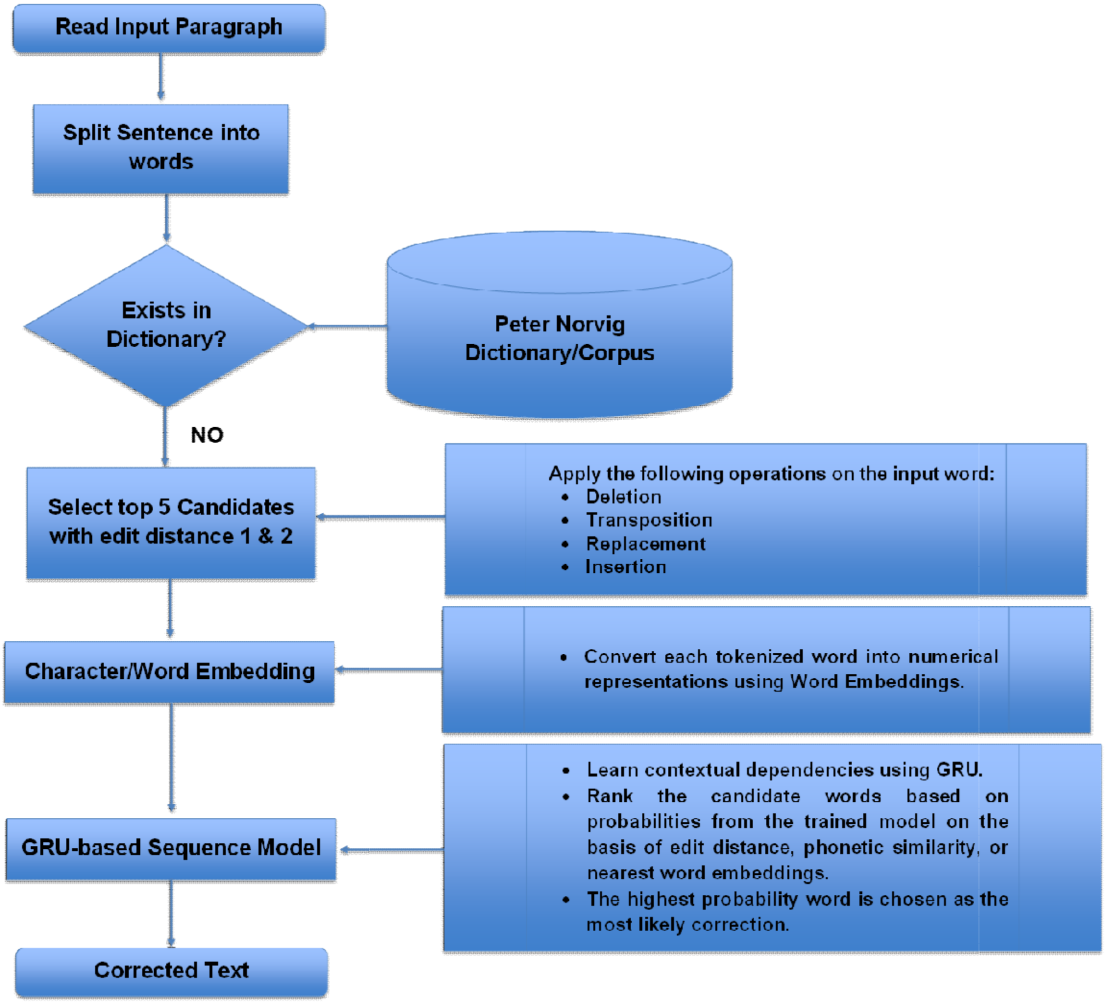
Read Input Paragraph
Split Sentence into words
Exists in Dictionary?
Peter Norvig Dictionary/Corpus
NO
Select top 5 Candidates with edit distance 1 & 2
Apply the following operations on the input word:
• Deletion
• Transposition
• Replacement
• Insertion
Character/Word Embedding
• Convert each tokenized word into numerical representations using Word Embeddings.
GRU-based Sequence Model
• Learn contextual dependencies using GRU.
• Rank the candidate words based on probabilities from the trained model on the basis of edit distance, phonetic similarity, or nearest word embeddings.
• The highest probability word is chosen as the most likely correction.
Corrected Text

Figure 4.1 Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with GRU – GUJAPUBRIJ Model

Peter Norvig's Algorithm:

- Apply edit distance to fix errors based on probability.
- Perfect for finding and correcting typos that contain words.
- It requires a defined dictionary to function properly, but does not have contextual awareness.

GRU-Based neural networks:

- Improvements in mistake detection can be achieved by sequential language data processing.
- keeps track of interdependencies in context to enhance grammar checkers.
- It deals with real errors that Norvig's method cannot fix on its own.

4.2. Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with IndicBERT – GUJBRIJAPU Model

This GUJBRIJAPU model outlines the development process of a Gujarati spell checker using advanced machine learning techniques. The research advanced from a fundamental dictionary-based model to a more sophisticated BERT-based model skilled in ranking and selecting the most accurate sentences.

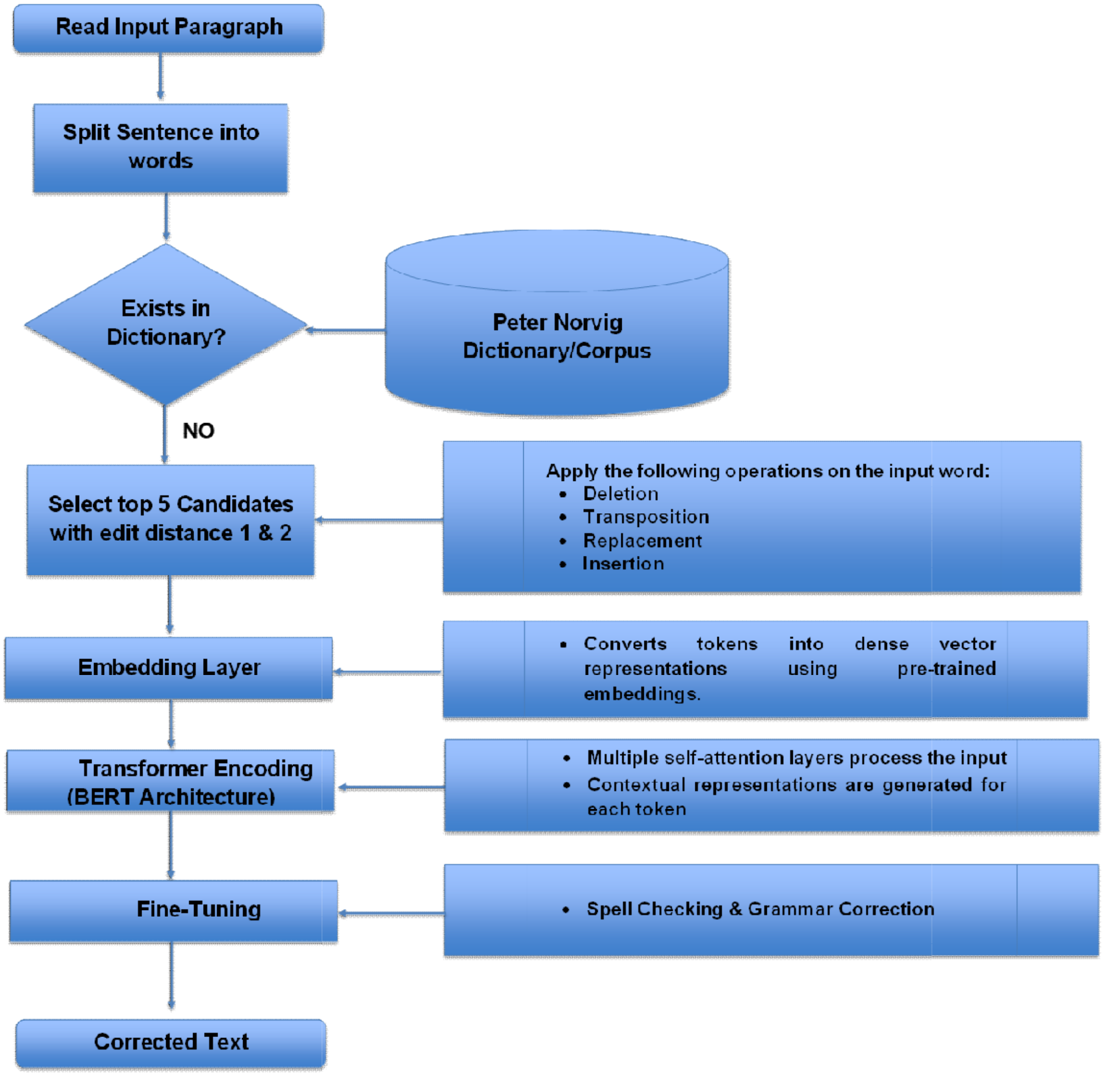
Read Input Paragraph
Split Sentence into words
Exists in Dictionary?
Peter Norvig Dictionary/Corpus
NO
Select top 5 Candidates with edit distance 1 & 2
Apply the following operations on the input word:
Deletion
Transposition
Replacement
Insertion
Embedding Layer
Converts tokens into dense vector representations using pre-trained embeddings.
Transformer Encoding (BERT Architecture)
Multiple self-attention layers process the input
Contextual representations are generated for each token
Fine-Tuning
Spell Checking & Grammar Correction
Corrected Text

Figure 4.2 Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with IndicBERT- GUJBRIJAPU Model

The preprocessing pipeline for the context-aware Gujarati spell checker integrates both lexical normalization and semantic validation. Initially, a cleaned Gujarati word corpus was used to construct a vocabulary that serves as the reference dictionary. The initial phase of this method is to segment the given text into smaller pieces to facilitate word verification in the Peter Norvig lexicon. For each token in an input sentence, Peter Norvig's probabilistic spell correction algorithm was applied to generate candidate corrections using edit-distance-based operations—insertions, deletions, substitutions, and transpositions tailored to the Gujarati script. This step ensured that all generated candidates conformed to orthographic rules of the language. The model will select the top five items with an edit distance of 1 or 2 if the word is incorrect.

Subsequently, to incorporate contextual understanding, IndicBERT, a multilingual language model which is pre-trained on Indian languages, is employed. Candidate-corrected sentences were passed through IndicBERT to obtain contextual embeddings. A scoring mechanism, typically involving masked language modeling or sentence-level probability estimation, was used to rank candidates based on their contextual fit. The candidate sentence with the highest semantic plausibility score was selected as the final correction. This two-stage preprocessing approach enabled robust handling of both non-word and real-word errors by aligning lexical correction with contextual relevance.

The subsequent embedding layer transforms tokens into dense vector representations using pre-trained embeddings. These representations are then processed through multiple self-attention layers of IndicBERT, enabling fine-tuning of the model for contextual analysis. This approach enhances the accuracy of both grammar and spelling checks.

The BERT model (mMML) was implemented to improve the accuracy and efficiency of the spell checker. BERT comprises two principal variants:

- Masked Language Model: Concentrates on forecasting absent or obscured words within a sentence.
- Sentence Scoring Model: Evaluates the probability of a sentence's correctness. The Sentence Scoring Model was considered more appropriate for the Gujarati spell checker, as it enables the grading of candidate sentences according to their accuracy.

5. Analysis and Discussion

5.1. Dataset

To develop a Gujarati language dataset for the study, data was sourced from publicly available resources provided by the Ekatra Foundation, accessible via https://www.ekatrafoundation.org/. Additionally, a curated Google Drive folder containing extensive Gujarati textual data was utilized (https://drive.google.com/drive/folders/17gskNhAGgzOpncOh2VsAKC4Fc0ju5GaC?usp=sharing). Over 100,000 sentences were initially collected from these sources. After performing a thorough data cleaning and preprocessing process to ensure quality and relevance, a final dataset comprising 20,000 sentences was created. This refined dataset includes both correct and erroneous sentence pairs, making it suitable for tasks such as spell error correction and language model training.

5.2. Training and Testing of Proposed model

The Peter Norvig and GRU based GUJAPUBRIJ novel context aware spelling checker for Gujarati Text was trained using 4204 validation data points and 16816 training data points. The Figure 5.1 shows the graph for training and validation accuracy while Figure 5.2 shows the graph for training and validation loss.

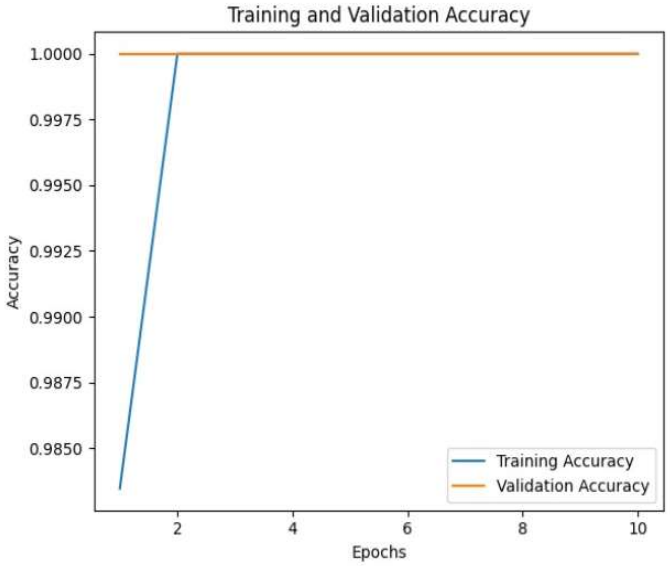

Figure 5.1 Training and Testing accuracy for Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with GRU – GUJAPUBRIJ Model

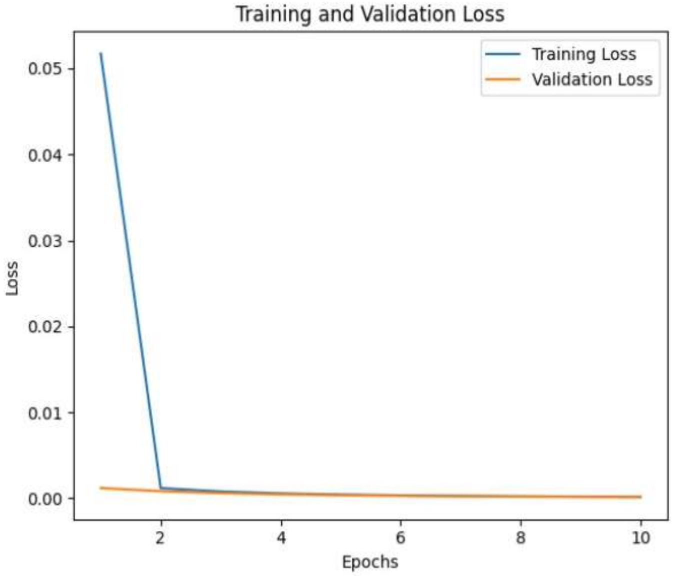

Figure 5.2 Training and validation loss for Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with GRU- GUJBRIJAPU Model

The training accuracy begins to increase right from the start of the training process and achieves 100% after only 2 epochs. Given overfitting, this implies that the model has rapidly acquired the capacity to forecast the training data; this may be explained by the accuracy rapidly reaching its maximum value. Like the training accuracy, the validation accuracy—shown by the orange line—beginners at a rather lower value but rapidly stabilizes around 1.000 (or 100%) after only two epochs. This shows that the model performs reasonably on the validation data, which is a positive indicator implying that it is not only recalling the training data but also able of performing on unknown data.

This hybrid model is tested in 100 sentences, with both correct and incorrect. Below are the results. The following accuracy statistics were achieved for correct and incorrect sentences:

Table 5.1 Performance analysis of Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with GRU – GUJAPUBRIJ Model

Peter Norvig with GRU – GUJAPUBRIJ Model		
	Incorrect sentences (in %)	Correct Sentences (in %)
Accuracy	71.00	85.00
Precision	71.00	84.00
Recall	71.00	85.00

Random Search is used as a hyperparameter tweaking technique to maximize the performance of the machine learning model. The settings or configurations—such as learning rate, number of layers, batch size, etc.—that are not learnt from data but rather must be defined before the training process starts define hyperparameters. Random Search picks random combinations of hyperparameter values from a specified range or distribution instead of exhaustively testing all conceivable combinations as in Grid Search.

Random Search is selected with this model to quickly investigate the hyperparameter space and find a combination that produces decent model performance without generating great computational expense. Especially when some hyperparameters have little influence on the model's output, it enables quicker convergence to an optimal or near-optimal solution. The table 3 shows the hyperparameters used for the proposed GUJAPUBRIJ model based on Peternorvig with GRU.

Table 2 Hyperparameter tuning for the Novel and hybrid Spelling and Grammar Error Correction approach for Gujarati Language using Peter Norvig with GRU- GUJAPUBRIJ Model

Hyperparameter	**Range / Value**	**Description**
embedding_dim	64 to 256 (step=32)	Size of embedding vectors, tuned using Keras Tuner
gru_units	32 to 128 (step=32)	Number of units in GRU layer, tuned using Keras Tuner
input_length	max_seq_length from data	Length of padded sequences
optimizer	Adam	Optimization algorithm used for training
loss	Binary Crossentropy	Loss function for binary classification
metrics	Accuracy	Evaluation metric during training and validation
tuning method	Random Search	Keras Tuner with 10 trials and validation accuracy goal

The refinement procedure and evaluation system for the GUJBRIJAPU model based on Peter Norvig with IndicBERT is utilized for Gujarati spelling and grammar correction. Here is a structured analysis of the process:

- A dataset consisting of 20,000 sentences has been created.
- Every sentence was classified as:
    - The correct sentence represents the ideal form, both grammatically and orthographically accurate.
    - The sentence presents common spelling and grammatical errors that are often found in Gujarati.
- Sentences were evaluated based on probability:

- Incorrect sentences received a score of 0.1, indicating a low level of correctness.
- Accurate sentences received a score of 0.9, which indicates a high level of correctness.

The model was trained and guided using these probability-based ratings to help distinguish between correct and incorrect sentences. A regression-based training approach was implemented to refine the IndicBERT GUJBRIJAPU model. The GUJBRIJAPU model provides a probability score for each candidate sentence.

- Sentence Ranking: The sentences are organized according to the scores they have been assigned.
- The sentence that receives the highest score is selected as the most likely correct version.

Through the application of regression-based scoring for fine-tuning, IndicBERT improves its capability to identify the most contextually accurate sentence. This method effectively distinguishes subtle differences between minor spelling errors and serious grammatical issues.

The ranking mechanism ensures that the most suitable candidate sentence is chosen, thereby enhancing the model's correction accuracy.

Training epochs and results for the GUJBRIJAPU Model based on Peter Norvig with IndicBERT are plotted as shown in Figure 5.3 and Figure 5.4.

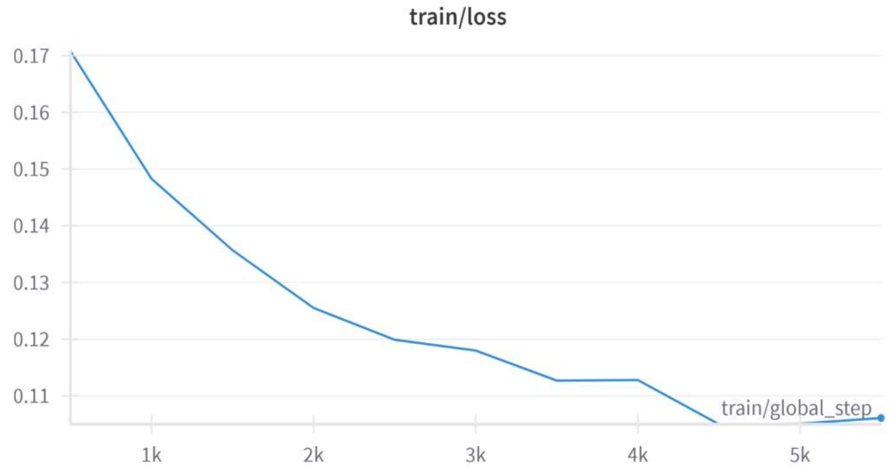

Figure 5.3 train/loss plot for Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with IndicBERT – GUJBRIJAPU Model

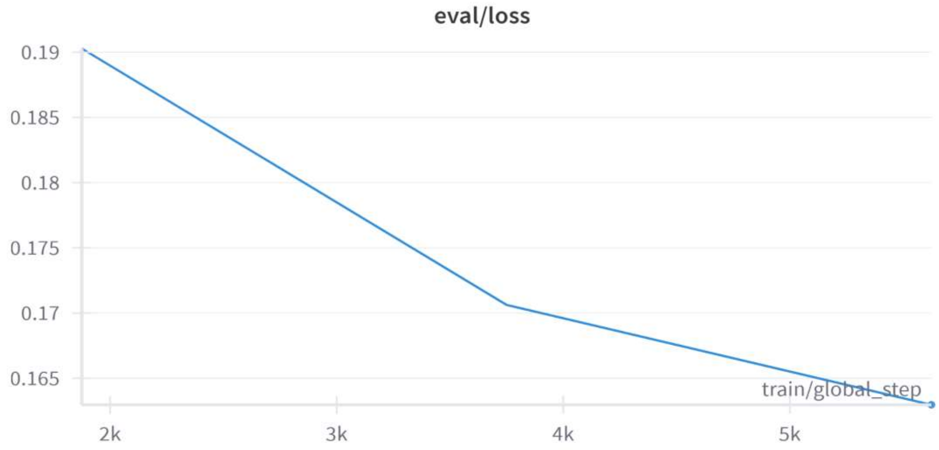

Figure 5.4 eval/loss plot for Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with IndicBERT – GUJBRIJAPU Model

It was tested on a fully independent test set comprising 100 sentences, with the results shown below. The following accuracy statistics were achieved for incorrect and correct sentences:

Table 5.3 Performance analysis of Novel and hybrid Spelling and Grammar Error Correction approaches for Gujarati Language using Peter Norvig with IndicBERT – GUJBRIJAPU Model

Peter Norvig with Indic BERT – GUJBRIJAPU Model		
	Incorrect sentences (in %)	Correct Sentences (in %)
Accuracy	84.79	93.49
Precision	86.21	94.46
Recall	83.54	90.13
F1 Score	85.74	91.59

Table 4 Hyperparameter tuning for the Novel and hybrid Spelling and Grammar Error Correction approach for Gujarati Language using Peter Norvig with IndicBERT – GUJBRIJAPU Model

Parameter	**Value**	**Description**
Model	ai4bharat/IndicBERTv2-SS	Pretrained Indic language transformer model
Number of Labels	1	Binary classification setup
Tokenizer	AutoTokenizer (from model)	Handles subword tokenization using the same model
Optimizer	Adam	Adaptive Moment Estimation (default in Trainer)
Learning Rate	2e-5	Fine-tuning rate for BERT parameters
Batch Size	16	Per-device mini-batch size
Number of Epochs	3	Number of full passes over training data
Weight Decay	0.01	L2 regularization strength
Evaluation Strategy	epoch	Evaluate after every epoch
Loss Function	Binary Cross Entropy (via	Used for sentence-level scoring

	Trainer)	
Compute Metrics	Custom scoring (e.g., accuracy or ranking)	Contextual ranking of candidates

Advantages of the GUJBRIJAPU model based on IndicBERT

- Accuracy: Transitioning to a BERT-based Sentence Scoring Model significantly improved the spell checker's performance.
- Flexibility: The model adapts to various types of spelling errors.
- Scalability: The methodology allows for seamless integration of additional data for further improvements

5.3. Comparative analysis of both the proposed GUJAPUBRIJ and GUJBRIJAPU Model

Two models for Gujarati spelling and grammar checking are compared in the Figure 5.5 graph on the basis of Accuracy, precision, recall and F1 score with the metric values displayed on the y-axis (percentage scale) and the metric types arranged on the x-axis for incorrect sentences:

- Peter Norvig along with GRU – GUJAPUBRIJ Model, depicted in blue with a solid line and triangle markers.
- Peter Norvig along with IndicBERT – GUJBRIJAPU Model, which is represented in green with a dashed line and square markers.

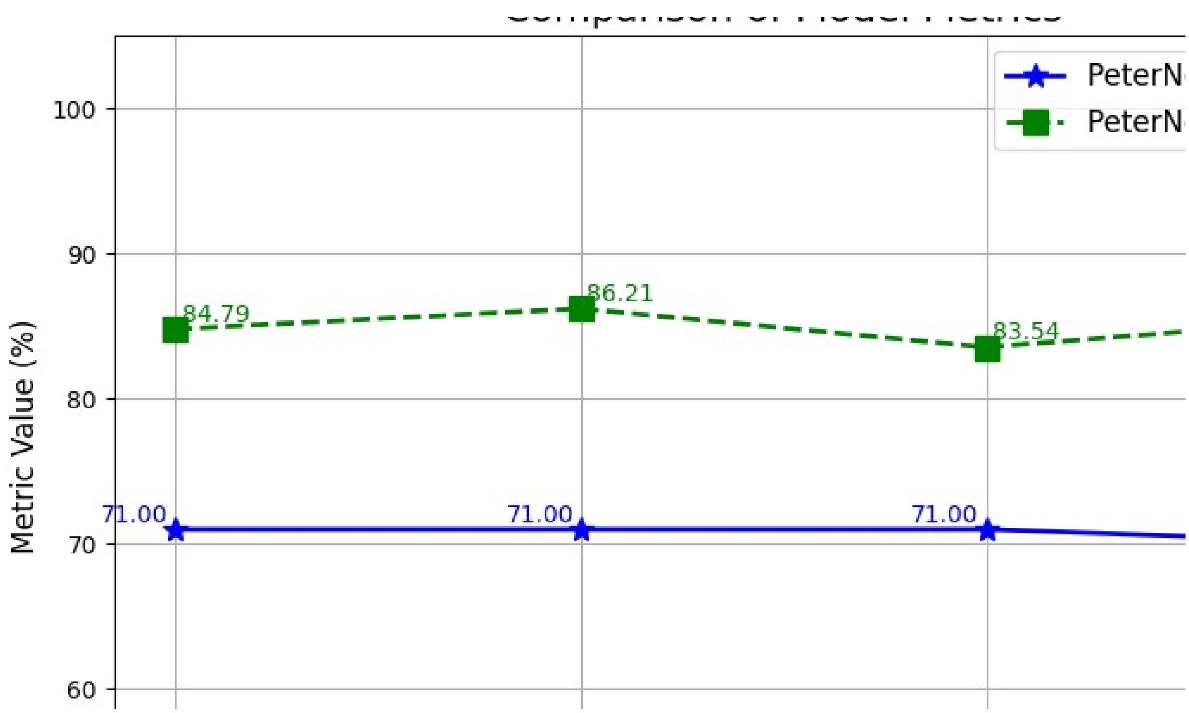

Figure 5.5 Comparative analysis of both the proposed model on incorrect sentences

Overall Analysis

- IndicBERT based GUJBRIJAPU Model has better ability in precisely identifying erroneous sentences, hence reducing the quantity of misclassified incorrect sentences. Her accuracy is 84.79%, substantially greater than that of GRU based GUJAPUBRIJ Model 71.00%.
- Enhanced accuracy lets IndicBERT avoid misclassification of accurate sentences and help to lower the false positives by precisely recognizing erroneous sentences.
- Given that the F1-score represents a balance between precision and recall, a higher F1-score suggests that IndicBERT based GUJBRIJAPU Model demonstrates greater reliability in identifying incorrect sentences when compared to GRU based GUJAPUBRIJ Model.
- IndicBERT based GUJBRIJAPU demonstrates a higher recall, indicating its superior ability to detect incorrect sentences while missing fewer compared to GRU based GUJAPUBRIJ Model.

The graph shown in Figure 5.6 presents a comparison of two models for context aware checking spelling check in Gujarati language.

- Peter Norvig with GRU based GUJAPUBRIJ Model (represented by a blue solid line with triangle marks).
- Peter Norvig along with IndicBERT based GUJBRIJAPU Model, represented by the green dashed line featuring square markings.

Accuracy, Precision, Recall, and F1 Score serve as metrics for evaluating models, with the metric values represented on the y-axis (% scale) and the types displayed on the x-axis.

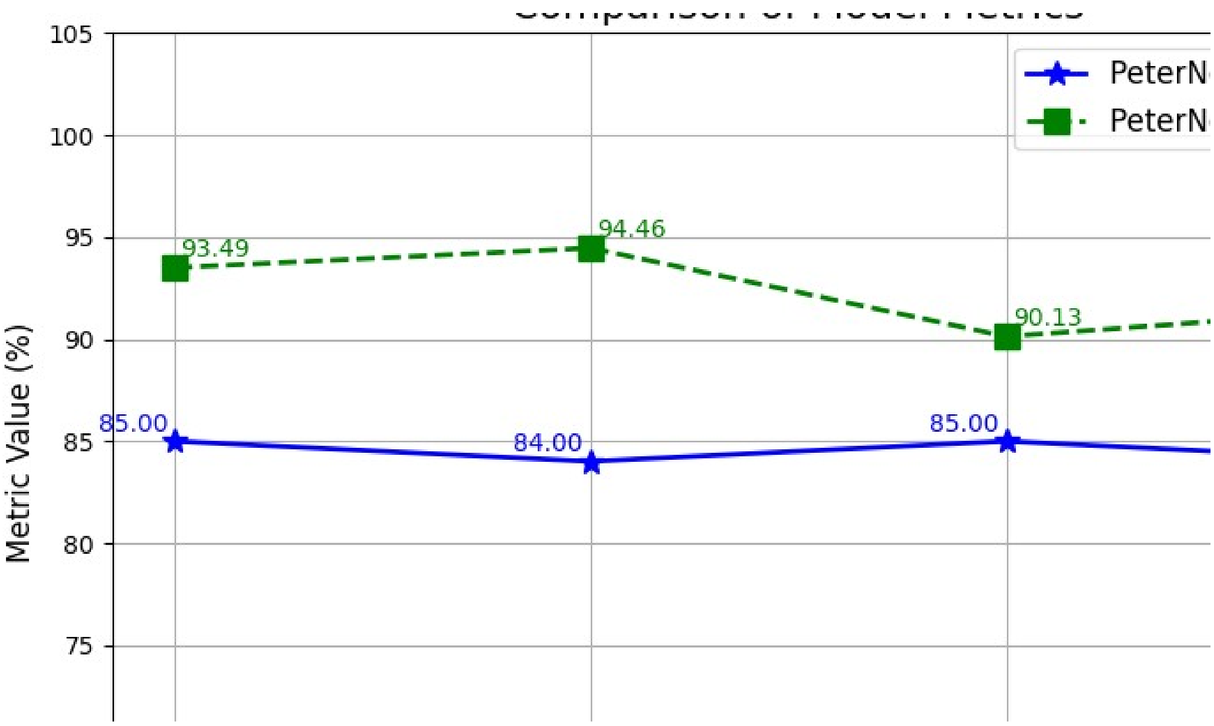

Figure 5.6 Comparative analysis of both the proposed models for correct sentences

Overall Analysis

- Peter Norvig combined with IndicBERT based GUJBRIJAPU Model demonstrates superior performance compared to Peter Norvig with GRU based GUJAPUBRIJ Model across all key metrics for identifying correct sentences.
- It demonstrates higher accuracy (93.49% compared to 85.00%), improved precision (94.46% versus 84.00%), greater recall (90.13% in contrast to 85.00%), and a superior F1-score (91.59% against 84.00%).
- This suggests that IndicBERT based GUJBRIJAPU Model demonstrates greater accuracy, confidence, and reliability in identifying correct sentences, resulting in fewer false positives and missed correct sentences.

5.4. Comparative analysis of proposed model with Jodani: Gujarati Spell Checker & Suggestion System

The Novel Peter Norvig and GRU/IndicBERT based GUJAPUBRIJ/GUJBRIJAPU Model context aware spelling checker for Gujarati Text is compared with Jodani which a spellchecker tool for Gujarati that finds mistakes and fixes them while also suggesting ways to be more accurate. Jodani boosts accuracy by combining rule-based and statistical methods and uses a predetermined Gujarati vocabulary to identify misspelled words using edit distance and phonetic comparableness. It deals with orthographic mistakes such as missing characters, transposition, addition, and substitution using phonetic similarities, N-gram models, and Peter Norvig's technique for correcting spelling to enhance precision.

Table 5.5 Comparative analysis of proposed GUJAPUBRIJ/GUJBRIJAPU model with Jodani [3]

Feature	**Peter Norvig with GRU / IndicBERT– GUJAPUBRIJ/GUJBRIJAPU**	**Jodani** [3]
Spelling Correction	Deep- learning based approach	Rule based
Grammar Checking	Yes	No
Contextual Understanding	Yes	No
Efficiency & Speed	More Computation power	Faster and lightweight
Scalability & Learning	Can be fine-tuned	Predefined rules are used.
Use case	NLP applications (Chatbots, Translation, Gujarati Grammar Checking)	Word Processing, Typo Correction

Table 5.6 Performance analysis of proposed model with Jodani [3]

Metric	Peter Norvig with GRU- **GUJAPUBRIJ**	Peter Norvig with IndicBERT- **GUJBRIJAPU**	*Jodani* [3]
Accuracy	85.00%	93.49%	~87-90%
Precision	84.00%	94.46%	~89.00%
Recall	85.00%	90.13%	~86.00%
F1 Score	81.00%	91.59%	~87.00%

From the above comparison, it can be observed that *'Jodani'* works well for spelling correction based on rules, but it doesn't analyze grammar while the novel approach based on Peter Norvig and IndicBERT – GUJBRIJAPU Model can more accurately detect context aware grammatical mistakes which can be used in variety of applications where context of text plays vital role. Jodani offers a comprehensive spell-checking solution for the Gujarati language, utilizing lexicon-based, statistical, and phonetic similarity methods. The system effectively corrects spelling errors and enhances the quality of written Gujarati text, which makes it a valuable tool for language processing applications. It frequently encounters challenges with context-dependent errors and does not possess a profound understanding of linguistics. In contrast, the Peter Norvig + IndicBERT based GUJBRIJAPU model utilizes deep learning techniques, providing enhanced accuracy and contextual analysis that allow it to effectively correct context aware spelling errors in Gujarati text. For applications that necessitate a comprehensive comprehension of language, the proposed approach based on Peter Norvig and IndicBERT based GUJBRIJAPU is the optimal choice. Nonetheless, in situations that demand quicker and more efficient processing, Jodani continues to be a suitable choice.

Conclusion

Spell and grammar checkers are vital to natural language processing (NLP) because they detect and fix mistakes in textual input. Although a lot of study has been done on English and other generally spoken languages, the regional languages of India provide special difficulties because of their complicated morphology, sophisticated phonetic patterns, and different scripts. The study carried out in this article emphasizes how difficult context aware spell checking in Indian regional languages—especially Gujarati—is given their intricate linguistic systems. The novel and hybrid error detection and correction approach based on Peter Norvig's spelling correction algorithm in collaboration with GRU (GUJAPUBRIJ Model) neural networks and IndicBERT (GUJBRIJAPU Model) are proposed and assessed. With outstanding accuracy, precision, recall, and F1-score, IndicBERT regularly exceeded GRU among the evaluated models. For all important criteria, Peter Norvig's method combined with IndicBERT means GUJBRIJAPU Model showed the best performance; thus, it is the most dependable method for context aware grammar correction in Gujarati literature. Jodani, rule-based spell checker for Gujarati, suffers with grammatical precision and lacks contextual knowledge even if it offers a quick rule-based fix for spelling errors. On the other hand, the Peter Norvig with IndicBERT GUJBRIJAPU Model is perfect for uses needing excellent contextual understanding since it uses deep learning to precisely identify spelling and grammatical mistakes. Still, Jodani is a good choice in situations requiring speedier processing with reduced computing burden. With improved accuracy and contextual awareness, the Peter Norvig with IndicBERT GUJBRIJAPU Model eventually seems to be the most solid approach for Gujarati language processing. Its capacity to lower false positives, increase classification accuracy, and offer thorough error correction makes it a useful tool for many NLP uses where linguistic accuracy is crucial. The computation overload can be reduced in future to improve the performance of the Peter Norving with IndicBERT GUJBRIJAPU model.

References

[1] N. G. Patel and D. D. B. Patel, "Research review of Rule Based Gujarati Grammar Implementation with the Concepts of Natural Language Processing (NLP)," *Journal of Emerging Technologies and Innovative Research (JETIR),* vol. 5, no. 9, 2018.

[2] N. P. Desai and V. K. Dabhi, "Resources and components for Gujarati NLP systems: a survey.," *Artificial Intelligence Review ,* vol. 55, pp. 1-19, 2022.

[3] H. Patel, B. Patel and K. Lad, "Jodani: A spell checking and suggesting tool for Gujarati language," in *11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 2021.

[4] S. Singh and S. Singh., "HINDIA: a deep-learning-based model for spell-checking of Hindi language," *Neural Computing and Applications,* vol. 33, no. 8, pp. 3825-3840, 2021.

[5] M. Gokani and R. Mamidi, "GSAC: A Gujarati Sentiment Analysis Corpus from Twitter," in *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, Association for Computational Linguistics*, 2023.

[6] J. Baxi and B. Bhatt., "GujMORPH-ADatasetforCreatingGujaratiMorphological Analyzer," in *ProceedingsoftheThirteenthLanguageResourcesandEvaluationConference*, 2022.

[7] A. A. Desai, "Gujarati handwritten numeral optical character reorganization through neural network.," *Pattern recognition,* vol. 43, no. 7, pp. 2582-2589, 2010.

[8] S. Antani and L. Agnihotri, "Gujarati character recognition," in *Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR '99*, Bangalore, India,, 1999.

[9] C. P. B. Tailor, "Chunker for Gujarati Language Using Hybrid Approach," in *Rising Threats in Expert Applications and Solutions. Advances in Intelligent Systems and Computing*, 2021.

[10] K. Suba, D. Jiandani and P. Bhattacharyya, "Hybrid inflectional stemmer and rule-based derivational stemmer for gujarati.," in *Proceedings of the 2nd workshop on south southeast Asian natural language processing (WSSANLP)*, 2011.

[11] B. K. Y. Panchal and A. Shah, "Spell Checker Using Norvig Algorithm for Gujarati Language," in *nternational Conference on Smart Data Intelligence. Singapore*, Singapore, 2024.

[12] N. Patel and D. Patel, "Implementation Approach of Indian Language Gujarati Grammar's Concept “sandhi” using the Concepts of Rule-based NLP," in *8th International Conference on Computing for Sustainable Global Development (INDIACom).*, 2021.

[13] J. Sheth and B. C. Patel., "Gujarati phonetics and Levenshtein based string similarity measure for Gujarati language.," in *5th National Conference on Indian Language Computing.*, 2015.

[14] T. A. Gal, "Natural Language Processing(NLP) Pipeline," Medium, 23 Oct 2023. [Online]. Available: https://medium.com/@theaveragegal/natural-language-processing-nlp-pipeline-e766d832a1e5. [Accessed 22 02 2025].

[15] P. Patel, K. Popat and P. Bhattacharyya, "Hybrid stemmer for Gujarati," in *Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing*, 2010.

[16] M. Parikh and A. Desai, "Recognition of Handwritten Gujarati Conjuncts Using the Convolutional Neural Network Architectures: AlexNet, GoogLeNet, Inception V3, and ResNet50," in *Advances in Computing and Data Sciences: 6th International Conference, ICACDS2022,*, Kurnool,India, 2022.

[17] B. K. Y. Panchal and A. Shah, "NLP-Based Spellchecker and Grammar Checker for Indic Languages.," in *Natural Language Processing for Software Engineering*, Scrivener Publishing LLC, 2025, pp. 43-70.

[18] C. Tailor and B. Patel, "Sentence Tokenization Using Statistical Unsupervised Machine LearningandRule-BasedApproachforRunningTextinGujaratiLanguage," in *Emerging Trends in Expert Applications andSecurity.AdvancesinIntelligent SystemsandComputing*, 2018.

[19] S. Sooraj, K. Manjusha, M. A. Kumar and K. P. Soman, "Deep learning based spell checker for Malayalam language," *Journal of Intelligent & Fuzzy Systems,* vol. 34, no. 3, pp. 1427-1434, 2018.

[20] S. Murugan, T. A. Bakthavatchalam and M. Sankarasubbu, "Symspell and lstm based spell-checkers for tamil," in *Tamil Internet Conference,*, 2020.

[21] N. Hossain, M. H. Bijoy, S. Islam and S. Shatabda, "Panini: a transformer-based grammatical error correction method for Bangla," *Neural Computing and Applications,* vol. 36, pp. 3463-3477, 2024.

[22] R. Phukan, M. Neog and N. Baruah, "A Deep Learning Based Approach For Spelling Error Detection In The Assamese Language," in *14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, 2023.

[23] S. S. Jamwal and P. Gupta., "A Novel Hybrid Approach for the Designing and Implementation of Dogri Spell Checker," in *Data, Engineering and Applications: Select Proceedings of IDEA 2021*, Singapore, 2022.

[24] M. Das, S. Borgohain, J. Gogoi and S. Nair, "Design and implementation of a spell checker for Assamese," in *Language Engineering Conference, 2002. Proceedings*, 2002.

[25] S. Iqbal, W. Anwar, U. I. Bajwa and Z. Rehman., "Urdu spell checking: Reverse edit distance approach," in *In Proceedings of the 4th workshop on south and southeast asian natural language processing*, 2013.

[26] A. A. Lawaye and B. S. Purkayastha, "KASHMIRI SPELL CHECKER AND SUGGESTION SYSTEM," *THE COMMUNICATIONS,* vol. 21, no. 2, p. 123, 2012.

[27] B. Kaur and H. Singh, "Design and implementation of HINSPELL—Hindi spell checker using hybrid approach," *International Journal of scientific research and management,* vol. 3, no. 2, pp. 2058-2062, 2015.

[28] R. Sankaravelayuthan, "Spell and grammar checker for Tamil.," *Developing computing tools for Tamil,* vol. 5, no. 23, pp. 52-64, 2015.

[29] A. A. Lawaye and B. S. Purkayastha, "Design and implementation of spell checker for Kashmiri," *International Journal of Scientific Research,* vol. 5, no. 7, 2016.

[30] R. Sakuntharaj and S. Mahesan, "A novel hybrid approach to detect and correct spelling in Tamil text," in *2016 IEEE international conference on information and automation for sustainability (ICIAfS)*, 2016.

[31] U. M. G. Rao, A. P. Kulkarni and a. P. K. Christopher Mala, "Telugu Spell-Checker," *Vaagartha,* 2012.

[32] S. Saha, F. Tabassum, K. Saha, Akter. and Marjana, "Bangla Spell Checker and Suggestion Generator," (Doctoral dissertation, United International University)., 2019.

[33] S. Singh and S. Singh, "Systematic review of spell-checkers for highly inflectional languages,"

Artificial Intelligence Review , vol. 53, no. 6, pp. 4051-4092, 2020.

[34] B. Bhagat and M. Dua, "Enhancing performance of end-to-end gujarati language asr using combination of integrated feature extraction and improved spell corrector algorithm," in *ITM Web of Conferences*, 2023.

[35] D. Kakwani, A. Kunchukuttan, S. Golla, G. NC, A. Bhattacharyya, M. M. Khapra and P. Kumar., "IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages," *In Findings of the association for computational linguistics: EMNLP 2020,* pp. 4948-4961, 2020.

[36] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, B. Gopalan, D. Margam, P. Aggarwal, R. Nagipogu, S. Dave and S. Gupta, "Muril: Multilingual representations for indian languages.," *arXiv preprint arXiv:2103,* p. 10730, 2021.

[37] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer and V. Stoyanov., "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116,* 2019.

[38] J. A. R. C. P. Pfeiffer, A. Kamath, I. Vulić, S. Ruder, K. Cho and I. Gurevych, "Adapterhub: A framework for adapting transformers," *arXiv preprint arXiv:2007.07779,* 2020.

[39] S. Deode, J. Gadre, A. Kajale, A. Joshi and R. Joshi, "L3Cube-IndicSBERT: A simple approach for learning cross-lingual sentence representations using multilingual BERT."," *arXiv preprint arXiv:2304.11434,* 2023.

[40] M. Nejja and A. Yousfi., "The context in automatic spell correction," *Procedia Computer Science,* vol. 73, pp. 109-114, 2015.

[41] A. K. Ingason, S. B. Jóhannsson, E. Rögnvaldsson, H. Loftsson and S. Helgadóttir., "Context-sensitive spelling correction and rich morphology.," in *Proceedings of the 17th Nordic Conference of Computational Linguistics (NODALIDA 2009)*, 2009.

[42] A. Yunus and M. Masum., "A context free spell correction method using supervised machine learning algorithms," *International Journal of Computer Applications,* vol. 176, no. 27, pp. 36-41, 2020.

[43] P. Gupta, "A context-sensitive real-time Spell Checker with language adaptability," in *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*, 2020.