RAYA JOURNAL OF SCIENCE AND DEVELOPMENT

RJSD 1(2):2026

Development of a Hybrid AI-Based Tigrigna Grammar Checking and Error Correction System with an Interactive Graphical User Interface

Gebrhans Weldegebriel Weldegers

Department of Computer Science, Adigrat University, Adigrat, Ethiopia

Article History:
Received: 2026-03-15
Accepted: 2026-06-02
Published: 2026-07-07
DOI: https://doi.org/10.82489/rjsd.2026.1.2.75

Suggested citation:

Development of a Hybrid AI-Based Tigrigna Grammar Checking and Error Correction System with an Interactive Graphical User Interface. (2026). Raya Journal of Science and Development, 1(2). https://doi.org/10.82489/rjsd.2026.1.2.75

***Correspondence :**
Gebrhans Weldegebriel Weldegers
magelgebrhans@gmail.com

Abstract

Tigrigna is one of the morphologically and syntactically rich but low-resource languages with limited annotated datasets, linguistic resources, and computational resources. Due to these challenges, developing an automated grammar analysis and correction is hindered. Therefore, this research presents the development of an automated Tgrigna grammar analyzer and correction system using a combination of rule-based, machine learning, and hybrid approaches using the Python programming language. The model is trained with a total of 2000 sentences, including 500 correct sentences and 500 incorrect sentences per class of agreement fault, gender biased fault, and tense fault sourced from books, online articles, and academic materials. In addition to this, the model was trained with word tokens of 625 nouns, 545 verbs, 440 adjectives, 14 pronouns, and 90 noun-prepositions and evaluated with 80% for training and 20% for testing. Experimental results indicate that the hybrid approach achieves an accuracy of 94%, significantly outperforming the pure machine learning model with an accuracy of 92% for random forest and naive Bayes, 91% for logistic regression, and 90% for linear svm and the rule-based model with an accuracy of 81%. The model also integrated features of a graphical user interface (GUI) that enable it to identify errors and provide automated grammatical suggestions and corrections.

Keywords: *Error correction, grammar checking, hybrid artificial intelligence, machine learning, Tigrigna language*

1. Introduction

Natural Language Processing (NLP) is a field of artificial intelligence that has experienced exponential growth with the advent of large language models (LLMs), machine learning (ML), and deep learning (DL) (Cloudflare, 2026). These advancements have been made for high-resource languages like English, German, French, Spanish, Chinese, etc. (Ethnologue 2026; Demilie and Salau, 2022).

Tigrigna is one of the Semitic languages spoken in Ethiopia and Eritrea, with over 9.9 million speakers (Ethnologue, 2026), and it remains a low-resource language due to limited annotated corpora, linguistic datasets, and language technology resources (Ranathunga et al., 2021; Eberhard et al., 2022). Although preliminary Tgrigna grammar checkers have been developed using rule-based, statistical, and dependency-based methods (Gebremariam et al., 2023; Kiros and Aray, 2021; Abdelkadr, 2021; Gebrekiros, 2018), they remain limited in coverage, robustness, and correction quality for complex sentence structures. Existing work focuses mainly on detecting a small set of agreement and word-order errors, while there is still no comprehensive hybrid grammar checker and correction system that integrates morphological analysis, dependency parsing, rule constraints, and intelligent correction suggestions for low-resource Tgrigna text.

Therefore, this study aims to develop an automated grammar analyzer and correction provider for Tigrigna by leveraging rule-based, machine learning-based, and hybrid approaches that enable it to solve the problems of low-resource language processing. The system integrates a graphical user interface (GUI) to enhance usability and accessibility for students, teachers, writers, and language professionals who need support in producing accurate Tigrigna text. The integration of the GUI enables interactive input, real-time grammar checking, and automatic correction suggestions with Python.

The proposed system not only addresses the scarcity of Tgrigna linguistic resources but also provides a practical tool for language learning, digital communication, and academic applications. By bridging rule-based linguistic knowledge with data-driven learning, this research contributes to advancing NLP for low-resource languages and lays a foundation for future work in Tigrigna language processing.

1.1. Statement of the Problem

Tigrigna is a low-resource language in the field of Natural Language Processing (NLP) in the field of Natural Language Processing (NLP) (Cloudflare, 2026; Desta and Lehal, 2023). This problems, leads to a challenge for the development of automated grammar checking and correction systems. Existing grammar analyzers and NLP tools were largely designed for high-resource languages (Tonja et al., 2021; Pakray et al., 2025) such as English, German, French, and Chinese, and these resources could not be directly applied to Tigrigna due to its unique linguistic morphology and the absence of sufficient corpora (Atnafu et al., 2024; Fitsum, 2025). This gap obstructs effective text analysis, error detection, and automatic correction for Tgrigna users, affecting digital communication, education, and academic writing.

To address the challenges of being a low-resource language, the researchers developed a grammar analyzer and correction provider for Tgrigna using rule-based, machine learning based and hybrid method which can support a graphical user interface (GUI) to ensure usability and accessibility for learners, professionals, and general users. Therefore, the specific objectives of this study were to analyze the morphological and syntactic characteristics of the Tigrigna language to identify common grammatical structures and patterns written in the Ge'ez script; to design and implement a rule-based morphological analyzer capable of decomposing complex agglutinated words and identifying their grammatical components; to integrate rule-based techniques with machine learning methods to build a hybrid framework for improved grammar analysis and detection; to develop a user-friendly graphical

user interface (GUI) that allows users to input Tigrigna text and receive real-time grammatical analysis and correction suggestions; and to evaluate the performance of the developed system in terms of grammar detection capability, accuracy, and usability.

2. Methodology

This research employed a hybrid-based model of methodology to develop a Tigrigna grammar analyzer and correction system. This methodology was conducted using a hybrid framework that takes some concepts of machine learning and rule-based approaches which support a user-friendly graphical user interface (GUI), as stated in the figure 1.

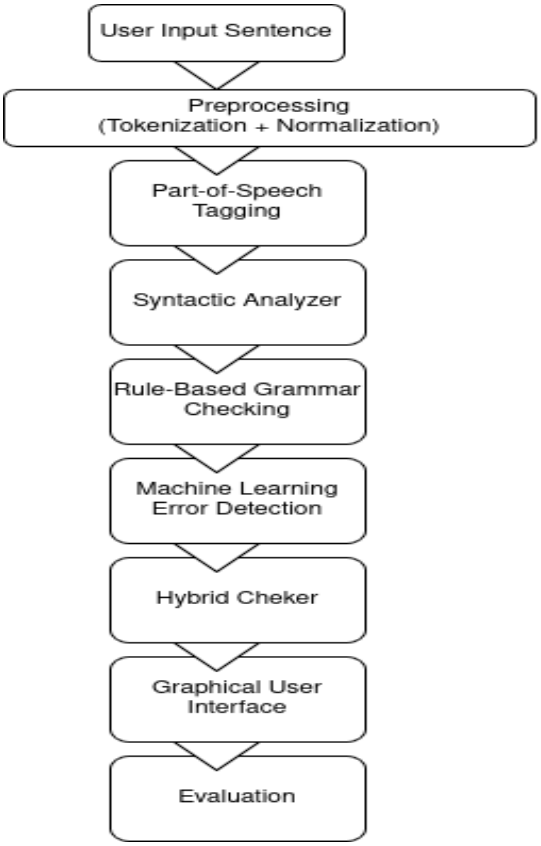


Figure 1

Proposed architecture

2.1. Data Collection and Text Preprocessin

In this study, the datasets were manually collected and annotated text data from books, online articles, academic materials, and user-contributed content with the collaboration of Adigrat University linguist researchers and lecturers, starting in 2025. During this study, the researchers employed a total sample of 2000 sentences, including 500 correct and 1500 incorrect sentences (500 agreement fault, 500 gender biased fault and 500 tense fault) with parts of speech such as a total sample of nouns of 625, verbs of 545, adjectives of 440, pronouns of 14, and prepositions of 90 tokens, as stated in the table 1 and 2 . In addition to this, the model was trained with ratios of 70:30 for training and testing, respectively; 70:20:10 for training, testing, and validation, respectively; and 80:20 for training and testing, respectively, and finally, the researchers found a high accuracy of 80:20 for training and testing, respectively. For cleaning and preparing the text for processing, the model employed whitespace tokenization. This preparation step involves removing any unnecessary spaces or line breaks found at the beginning and end of each sentence and splitting the Tigrigna text into tokens.

Table 1

Sentence Dataset Distribution

Category	Total	Training (80%)	Testing (20%)
Correct Sentences	500	400	100
Agreement Fault	500	400	100
Gender Fault	500	400	100
Tense Fault	500	400	100
Incorrect Sentences	1,500	1,200	300
Total Sentences	2,000	1,600	400

Table 2

POS Dataset Distribution

Category	Total	Training (80%)	Testing (20%)
NOUNS	2,080	1,664	416
Verbs	1,820	1,456	364
Adjectives	1,470	1,176	294
Pronouns	1,430	1,144	286
Prefix Nouns	300	240	60
Total Word	**7,100**	**5,680**	**1,420**

2.2. Rule-Based Classifier

In this phase, with the collaboration of Adigrat University lecturers and Tigrigna researchers, we developed explicit Tigrigna grammatical rules, covering syntax, morphology, and common sentence structures. These linguistic experts at Adigrat University helped to define error patterns, agreement rules, and correction suggestions.

This rule works by checking each word token and sentence against a fixed dictionary and a set of predefined grammar rules. If a word and sentence are found in the dictionary, they are labeled accordingly. If neither condition is met, the token is marked as unknown. This method struggles with new or unseen words and sentences that are not in its dictionary.

2.3. Machine Learning Classifier

In the Machine Learning Classifier, when encountering an out-of-dictionary sentences, rather than returning an unclassifiable state, the model makes a probabilistic guess by randomly selecting from the available word categories, namely noun, verb, adjective, pronoun, and prepositional noun, as defined in Equation (1). for this model, logistic regression, random forest, linear svm, and naive bayes were used.

$$\begin{equation}\smallf(t)_{Oov}\sim \text{Choice}\left(\{\texttt{NOUN}, \texttt{VERB}, \texttt{ADJ},\texttt{PRON},\texttt{NOUN\_PREP}\} \right) \end{equation}$$

2.4. Hybrid Classifier

This combines the strengths of both rule-based and machine learning method into a single layered system. It first checks the dictionary, then applies the prefix rules for sentences and tokens not found in the dictionary, and finally uses the probabilistic estimator for any sentences and tokens that still remain unclassified. By working through these steps in order, the hybrid model avoids the weaknesses of each individual approach of rule based and machine learning and achieves stronger and more consistent classification performance across the entire dataset.

2.4. Correction Recommendation Engine

Once grammatical errors are identified, the model generates context-aware correction suggestions based on rule-based (lexical information and predefined grammar rules) and machine learning-based (contextual analysis and predictions of probable corrections based on learned patterns) methods. This engine suggests alternative words or sentence structures that follow the grammatical standards of the Tigrigna language.

2.5. Graphical User Interface (GUI) Implementation

This model is implemented in Python using Tkinter for GUI development. The GUI allows users to input text, receive real-time grammar analysis, and view correction suggestions interactively.

2.6. performance Evaluation

The performance of the developed model was evaluated using standard metrics, including accuracy, precision, recall, and F1-score. In addition to these, confusion matrices were generated to compare the performance of rule-based, machine learning, and hybrid methods. The hybrid approach consistently demonstrated superior performance, validating the effectiveness of combining linguistic rules with data-driven learning.

3. Experimental Results

3.1. Experimental Setup

This model was conducted using the Python programming language using a set of Tigrigna sentences that hold both grammatically correct and incorrect structures. For the rule-based model, predefined grammatical rules were generated using a dictionary with the consultation of lecturers and researchers from Adigrat University, who are experts in linguistics. For the machine learning model, after extensive experiments of 70:30 (training: testing orderly), 70:20:10 (training:

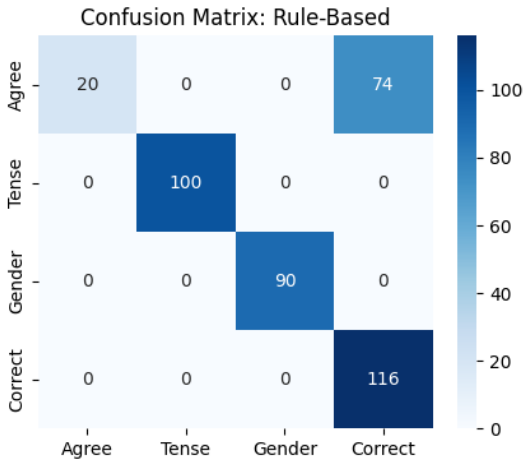
testing: validation respectively), and 80:20 (training: testing orderly) with naive Bayes, random forest, logistic regression and linear svm, the model was finally selected with a ratio of 80:20 for training and testing respectively since this provides the highest accuracy. Finally, the model integrated a rule-based approach with the machine learning method to enhance the detection and correction of grammatical errors.

3.2. Confusion Matrices

For visualizing detection accuracy and common classifications, confusion matrices for the rule-based model, the machine learning models, and the hybrid framework were developed. The hybrid framework showed the highest rate of true positives, indicating it successfully detects errors, suggests correct alternatives, and resolves ambiguities that arise from low-resource data limitations when we gave unseen datasets, as stated in the figures 4.

The machine learning methods such as random Forest, Naive Bayes, Logistic Regression and linear svm also demonstrated moderate detection accuracy, as stated in the figures 3 , even with limited datasets .

The rule-based technique provided excellent detection accuracy when it was given predefined rules, but it struggled when it was given new datasets, as stated in the figures 2 .



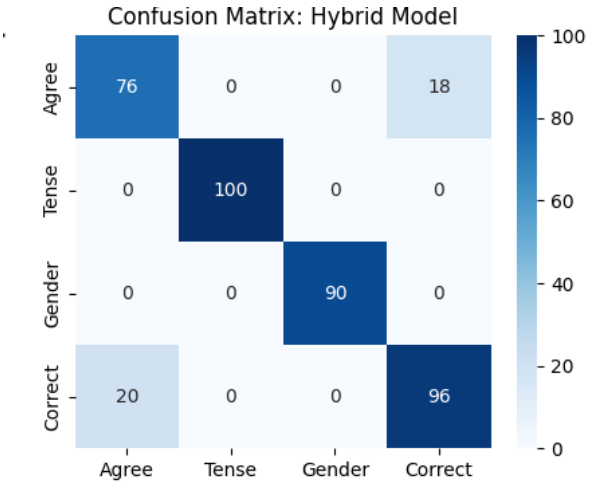
Figures 2

Confusion Matrix of the Rule-Based Grammar Analysis Model

Random Forest *Naive Bayes* *Logistic Regression* *Linear SVM*

Figures 3

Performance Evaluation of Machine Learning Grammar Analysis Models using Confusion Matrix

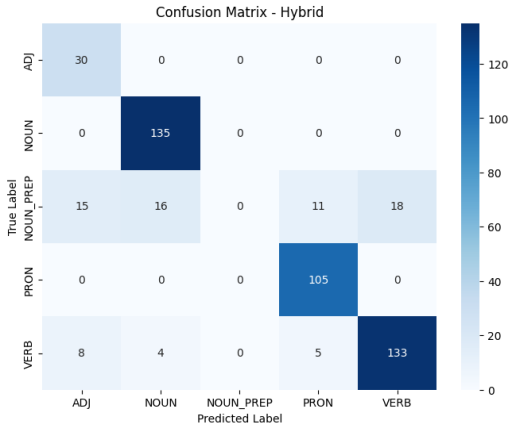


Figures 4

Confusion Matrix of the Hybrid Grammar Analysis Model

3.3. Confusion Matrix of Syntactical Analyzer

For effective learning of the hybrid model, in addition to correct and incorrect sentences from the dictionary, the model was integrated with the syntactical analyzer. Integrating the syntactical analyzer, which includes nouns, adjectives, verbs, pronouns, and prepositions, with the hybrid model allows it to learn from both correct sentences and incorrect sentences generated by the rule-based and machine learning models, as illustrated in figure 5.



Figures 5

Hybrid Model of Syntactical Analyzer using Confusion Matrix

3.4. Classification Reports

In addition to the confusion matrices, the models of rule-based approaches, machine learning models, and hybrid models were evaluated using accuracy, recall, precision, and F1-scores, as depicted in figures 6, 7, and figures 8.

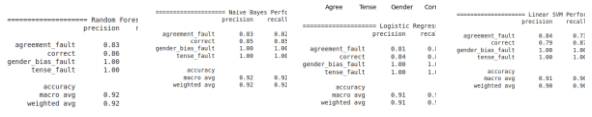
```
==================== Rule-Based ====================
                    precision    recall  f1-score   support

  agreement_fault       1.00      0.21      0.35        94
          correct       0.61      1.00      0.76       116
gender_bias_fault       1.00      1.00      1.00        90
      tense_fault       1.00      1.00      1.00       100

         accuracy                           0.81       400
        macro avg       0.90      0.80      0.78       400
     weighted avg       0.89      0.81      0.78       400
```

Figures 6

Classification Report of the Rule-Based Grammar Analysis Model



Random Forest Classification Reports

Naive Bayes Classification Reports

Logistic Regression Classification Reports

Linear SVM Classification Reports

Figures 7

Performance Evaluation of Machine Learning Grammar Analysis Models using Classification Reports

```
==================== Hybrid Model ====================
                    precision    recall  f1-score   support

  agreement_fault       0.79      0.81      0.80        94
          correct       0.84      0.83      0.83       116
gender_bias_fault       1.00      1.00      1.00        90
      tense_fault       1.00      1.00      1.00       100

         accuracy                           0.91       400
        macro avg       0.91      0.91      0.91       400
     weighted avg       0.91      0.91      0.91       400
```

Figures 8

Classification Report of the Hybrid Grammar Analysis Model

3.5. Performance Comparison

For this research study, a comparison of the rule-based, machine learning, and hybrid approaches was conducted as stated in Table 3. As stated in table 3, the experimental results indicate that the highest performance with an accuracy of 94% among the three methods was the hybrid approach. On the other hand, the machine learning model as depicted in Table 3 demonstrated improved performance with an accuracy of 92% for random forest and naive bayes, 91% for logistic regression, and 90% for linear svm. The rule-based model was effective in detecting grammatical errors defined by linguistic rules, but it struggled to capture contextual variations in sentence structures. So its accuracy was achieved at 81%, the lowest compared to the three models, as stated in Tables 3.

Tables 3

Accuracy Comparison of Rule-Based, Machine Learning, and Hybrid Approaches

Model	Accuracy
Rule-Based	81%
Machine Learning Models	
Random Forest	92%
Naïve Bayes	92%
Logistic regression	91%
Support Vector Machine(SVM)	90%
Hybrid Model	94%

3.6. Demonstration of Grammar Analysis and Recommendations

In this study, we developed the model using Python with the integration of GUI to demonstrate how the model successfully detects errors, suggests correct alternatives, and resolves ambiguities that arise from the low resources of the language. Figures 9, 10, and 11 demonstrate how to analyze each sentence, identify grammatical errors, and provide correction recommendations to handle both correct and incorrect sentences in real-time, and in addition to the figures, we are also demonstrating by table 4.

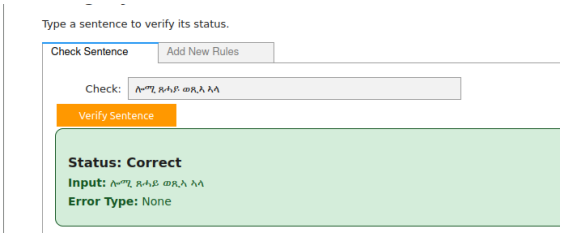


Figure 9

Correct Sentence Sample

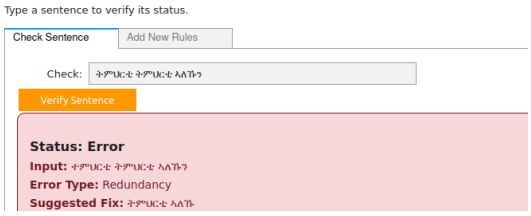


Figure 10

Incorrect Sentence Sample

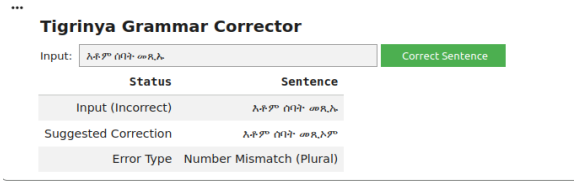


Figure 11

Incorrect Sentence Sample

Table 4

Examples of Tigrinya grammatical error correction generated by the proposed system

Input (Incorrect)	Suggested Correction	Error Type
	ትምህርቲ ኣለኹ	Redundancy
	ኣነ ሓላፍነት ኣሎኒ	Conjugation
	መንግስቲ ትምህርቲ ክፍሊ ኣለዎ	Agreement
	ንሳ ጽብቕቲ ሰብ እያ	Gender (F)
	ኣነ ሻሂ ክሰቲ እደሊ	Person (1st)
	እቶም ሰባት መጺኦም	Number (Pl.)
	ንስኻ ብስራት ኢኻ	Redundancy

4. Discussion

For this model, a rule-based approach for the identification of grammatical patterns of morphological structures was applied with the collaboration of linguist experts of Adigrat University. This model works well with predefined rules, whereas it struggles with unseen sentence structures.

The machine learning method captures patterns in user-generated text that may not be covered by a rule-based approach and provides flexibility in detecting contextual errors and variations in sentence formation.

The hybrid framework combines the strengths of both approaches and allows the model to achieve higher accuracy and more effective grammar analysis and recommendations compared to using either rule-based or machine learning techniques independently, as stated in the table 3.

Besides these models, a GUI was developed to enhance the practical usability of the system as a writing assistance tool for Tigrigna language users.

5. Conclusion

Due to the rich morphology of the Tigrigna language and the limited availability of annotated corpora and linguistic datasets, developing automated grammar analyzer and

error corrector tools has been a challenging task. To address this problem, the researchers implemented a hybrid model that integrates rule-based and machine learning approaches using Python. In this study, the experimental results demonstrated that the hybrid approach outperformed the individual rule-based and machine learning models. The hybrid model obtained an accuracy of 94%, which indicates that combining linguistic knowledge with data-driven learning significantly improves grammar error detection and correction for low-resource languages. This model also was integrated with a GUI to input text, receive grammar analysis, and receive correction suggestions interactively.

6. Future Work

For the development of more robust deep learning models such as transformers, the researcher recommended collecting larger and more diverse annotated datasets. The researcher also recommended supporting multilingual grammar analysis with low-resource languages in the region of Tigray, Ethiopia. To increase usability and accessibility for users, the researcher recommended integrating with a mobile application.

REFERENCES

Abdelkadr, Mehammedbrhan. 2021. Dependency-Based Tigrinya Grammar Checker. Master's thesis, Addis Ababa University. http://etd.aau.edu.et/handle/123456789/27003.

Atnafu Lambebo Tonja et. al., 2024. *LREC-COLING 2024*, 6341–52.

Cloudflare. 2026. *What Is NLP in AI? | Natural Language Processing*. https://www.cloudflare.com/learning/ai/natural-language-processing-nlp/.

Demilie, Wubetu Barud, and Ayodeji Olalekan Salau. 2022. "Automated All in One Misspelling Detection and Correction System for Ethiopian Languages." *Journal of Cloud Computing: Advances, Systems and Applications* 11 (1): 1–14. https://doi.org/10.1186/s13677-022-00299-1.

Desta, Solomon Gebremariam, and Gurpreet Singh Lehal. 2023. "Automatic Spelling Error Detection and Correction for Tigrigna Information Retrieval: A Hybrid Approach." *Bulletin of Electrical Engineering and Informatics* 12 (1): 387–94. https://doi.org/10.11591/eei.v12i1.4209.

Eberhard, David M., Gary F. Simons, and Charles D. Fennig. 2022. *Ethnologue: Languages of the World*. 25th ed. SIL International. http://www.ethnologue.com.

Ethnologue. 2026. *Tigrigna Language (TIR)*. https://www.ethnologue.com/language/tir/.

Fitsum Gaim, Jong C. Park. 2025. *arXiv:2507.17974*.

Gebrekiros, Abrha. 2018. "Development of Tigrigna Grammar Checker Using Hybrid Approach." Master's thesis, Addis Ababa University. http://etd.aau.edu.et/handle/123456789/18716.

Gebremariam, Solomon D et al. 2023. "Automatic Spelling Error Detection and Correction for Tigrigna Information Retrieval: A Hybrid Approach." *Bulletin of Electrical Engineering and Informatics*.

Kiros, Atakilti Brhanu, and Petros Ukbagergis Aray. 2021. "Tigrigna Language Spellchecker and Correction System for Mobile Phone Devices." *International Journal of Electrical and Computer Engineering (IJECE)* 11 (3): 2307–14.

Pakray, Partha, Alexander Gelbukh, and Sivaji Bandyopadhyay. 2025. "Natural Language Processing Applications for Low-Resource

Languages." *Natural Language Processing* 31: 183–97. https://doi.org/10.1017/nlp.2024.33.

Ranathunga, Surangika, En-Shiun Annie Lee, Marjana Prifti Skenduli, Ravi Shekhar, Mehreen Alam, and Rishemjit Kaur. 2021. "Neural Machine Translation for Low-Resource Languages: A Survey." *ACM Computing Surveys*.

Tonja, Atnafu Lambebo, Michael Melese Woldeyohannis, and Mesay Gemeda Yigezu. 2021. "A Parallel Corpora for Bi-Directional Neural Machine Translation for Low Resourced Ethiopian Languages." *2021 International Conference on Information and Communication Technology for Development for Africa (ICT4DA)*, 71–76. https://ieeexplore.ieee.org/document/9595930.