Grammar Feedback for Non-Native Hindi Learners

Uddip Yalamanchili
Dept. of Computer Science
George Mason University
uyalaman@gmu.edu

Aditya Shah
Dept. of Computer Science
George Mason University
ashah49@gmu.edu

Nikhil Chukka
Dept. of Computer Science
George Mason University
nchukka@gmu.edu

1 Introduction

1.1 Task / Research Question Description

The task is to develop a tool that provides grammar feedback for non-native Hindi learners. The goal is to help learners identify their grammatical errors, facilitating their journey toward fluency.

1.2 Motivation and Limitations of existing work

Existing resources primarily focus on vocabulary and basic phrases, but they lack personalized grammar feedback. While others have addressed general Hindi language learning, this project aims to offer real-time, individualized feedback that enhances learners' understanding of grammar, which existing solutions do not sufficiently provide. Prior efforts have not been able to cater to personalized grammar refinement, especially for self-directed learners.

1.3 Proposed Approach

The proposed approach*(tentative approaches)* includes integrating rule-based and pre-trained NLP models to detect grammatical errors and provide suggestions. Initial ideas involve using existing public datasets, pre-trained models and machine translation tools. The development of an API that allows users to input sentences and receive corrections and explanations is also planned.

1.4 Likely challenges and mitigations

Challenges include accurately detecting complex grammatical errors, handling diverse sentence structures, and ensuring the tool's performance. To mitigate these, the project plans to use a combination of rule-based systems and machine learning models. Additionally, there are contingency plans to involve human evaluations and real-world user testing to refine and improve the accuracy of the feedback tool.

2 Related Work

2.1 Vyakranly: Hindi Grammar & Spelling Errors Detection and Correction System:

Vyakranly focuses on addressing the growing need for automated Hindi language tools by providing a system that detects and corrects both spelling and grammar errors. It employs a combination of rule-based and statistical approaches, utilizing morphological analysis and part-of-speech tagging. The system is primarily designed for simple sentences but can also handle some compound sentences. Vyakranly integrates modules for grammar checking, spell correction, and translation between Hindi and English, aiming to create a holistic language processing tool for Hindi learners.

The existing grammar correction systems mainly address errors like adjective-noun and noun-verb agreements for simple sentences, as seen in prior works. However, they often lack the capacity to handle complex sentence structures, which Vyakranly attempts to overcome by employing hybrid methods that combine statistical techniques with rule-based corrections. Additionally, while many spell-checking tools are available, they often do not integrate grammar correction, making Vyakranly distinct in its combined approach. (S. et al., 2023)

2.2 Detection and Correction of Grammatical Errors in Hindi Language Using Hybrid Approach:

This research developed a grammar checking system for Hindi using a hybrid approach that combines statistical and rule-based methods. The system addresses four types of grammatical errors, in-

cluding adjective-noun and noun-verb agreement errors in terms of number and gender. By using a combination of morphological analysis, part-of-speech tagging, and pattern-based bigram and trigram methods, it effectively identifies and corrects grammatical issues in simple Hindi sentences. The system achieved a precision of 0.83, recall of 0.91, and an F-measure of 0.87. However, the approach mainly focuses on simple sentences, and its application to more complex structures remains limited. (**?**)

3 Methodology

This section must explain your approach to solving the problem. Remember, that this should ensure that your work is potentially reproducible, so try to be as detailed as possible. You don't need to describe established models/architectures from previous work as long as you appropriately cite the relevant papers. If you use prompt-based techniques, make sure to include your prompts (e.g. in an Appendix) along with the generation configuration.

3.1 Evaluation

How will you evaluate your system? Will you write unit tests? Will you perform human evaluation, or will you use automatic references, and why?

4 Experiments

4.1 Datasets

Please list which datasets you plan to use , whether or not you have access them, and whether or not they are publicly available with the same preprocessing and train / dev / tests as the previous work you will be comparing to (if applicable). If you plan to collect your own dataset, please describe clearly the data plan (the data source, how you plan to collect it, how you would preprocess it for the task, etc.).

4.2 Implementation

Please provide a link to a repo of your implementation (if applicable) and appropriately cite any resources you have used.

4.3 Results

Provide a table with your results.

4.4 Discussion

Analyze the performance of your model. Discuss any issues you faced. Did you do a sensitivity analysis (e.g. multiple runs with different random seeds)?

4.5 Resources

Discuss the cost of your solution in terms of resources: computation, time, people, development effort.

4.6 Error Analysis

Perform an error analysis on the system. Include at least 2-3 instances where your system fails and 2-3 where it succeeds.

5 Conclusion

Summarize your contribution in three sentences.

References

Rachel S., Vasudha S., Shriya T., Rhutuja K., and Lakshmi Gadhikar. 2023. Vyakranly : Hindi Grammar & Spelling Errors Detection and Correction System. In *2023 5th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*, pages 1–6.