Project Proposal Template

Uddip Yalamanchili
Department of Computer Science
George Mason University, USA
uyalaman@gmu.edu

Second Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

1 Introduction

1.1 Task / Research Question Description

The task is to develop a tool that provides personalized 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 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 from platforms like Hugging Face, 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.
should also talk about the challenges about the dataset and also the

2 Related Work

Include 3-4 sentence descriptions of no less than 4 relevant papers (as applicable). Also mention how your work differs from these. Note that prior work should be properly cited in References, e.g., when you use the BERT model (Devlin et al., 2019) you could cite it in this way; if you want to refer to the authors of a certain paper, you should use `citet`, e.g., "Devlin et al. (2019) proposed the BERT model." See `https://acl-org.github.io/ACLPUB/formatting.html` for instructions.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.