Automatic Spell correction and grammar correction using NLU in Indian Languages

M. Tech Dissertation in

Artificial Intelligence and Machine Learning

Submitted By : Gireesh Kumar Gopakumar

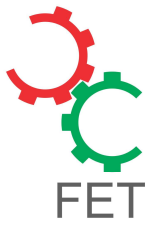
Reg. No. : 21ETCS126003

Supervisors : Dr NAYANA B .R
Associate Professor

August – 2023

FACULTY OF TECHNOLOGY
M. S. RAMAIAH UNIVERSITY OF APPLIED SCIENCES
Bengaluru -560 054

FACULTY OF < ENGINEERING AND TECHNOLOGY >



Certificate

This is to certify that the Dissertation titled "Automatic spell and grammar correction using NLU in Indian languages" is a bonafide record of the work carried out by Mr. Gireesh Kumar Gopakumar, Reg. No. 21ETCS126003 in partial fulfilment of requirements for the award of M. Tech. Degree of M. S. Ramaiah University of Applied Sciences in the Department of Artificial intelligence and Machine Learning.

August – 2023

Dr Nayana B R

Associate Professor

Supervisors

Dr. Rinki Sharma

Head – Dept. of CSE

Dr. Jagannathrao Venkatrao Desai

Dean-FET

Declaration

Automatic spell and grammar correction using NLU In Indian languages

The dissertation is submitted in partial fulfilment of academic requirements for the M.Tech. Degree of M. S. Ramaiah University of Applied Sciences in the department of Artificial Intelligence and Machine Learning. This dissertation is a result of my own investigation. All sections of the text and results, which have been obtained from other sources, are fully referenced. I understand that cheating and plagiarism constitute a breach of University regulations, hence this dissertation has been passed through plagiarism check and the report has been submitted to the supervisor.

Signature :

Name of the Student : **Gireesh Kumar Gopakumar**

Reg. No. : **21ETCS126003**

Date : **09 August 2023**

Acknowledgements

I am immensely thankful to my academic advisor, Dr. Nayana B R, for her exceptional guidance, steadfast support, and invaluable mentorship throughout the course of this dissertation. Her extensive knowledge, insightful feedback, and unwavering encouragement have played a pivotal role in shaping the trajectory of this research. Driven by her commitment to my academic and personal growth, I have been inspired to push my boundaries and pursue excellence in all aspects of this work.

My heartfelt appreciation extends to the Department Head, Dr. Rinki Sharma, for her continuous encouragement and backing. Her belief in my capabilities and the opportunities provided within the department have fostered an enriching research environment. Her visionary leadership has motivated me to persevere and bring this dissertation to fruition.

Additionally, I express my gratitude to the Faculty Dean, Dr. Jagannathrao Venkatrao Desai, for his unwavering support and dedication to promoting a conducive research atmosphere. His leadership and commitment to academic excellence have been instrumental in facilitating the completion of this project.

Lastly, I extend my sincere thanks to all individuals who have directly or indirectly contributed to the successful culmination of this dissertation. Your unwavering support, encouragement, and valuable insights have been indispensable in realizing this endeavour. I am deeply grateful to each one of you for being an integral part of this journey and for making it an enriching and fulfilling experience. Your guidance and faith in my abilities have been the driving force behind this achievement, and I am profoundly thankful for it.

With profound gratitude,

GireeshKumar

21ETCS126003

Abstract

This thesis investigates NLU-driven spell and grammar correction in Indian languages. Complex linguistic nuances in these languages' present challenges for automated language correction. With diverse languages and intricate grammar, innovative solutions are needed using Natural Language Understanding (NLU). The goal is to bridge human communication and automated correction effectively across linguistic diversity.

This study focuses on creating and evaluating NLU models for tailored spell and grammar correction in Indian languages. The methodology involves analysing language intricacies, addressing spell and grammar challenges. Advanced NLU techniques, like transformer-based models, are customized to capture language-specific traits. Diverse datasets, covering formal and informal contexts, train and validate the models. Experimental evaluation of various configurations sheds light on NLU's efficacy in handling linguistic complexities.

Outcomes include NLU-powered models for spelling and grammar correction in Indian languages, demonstrating effectiveness across diverse contexts. This study emphasizes context comprehension for successful correction, as accuracy goes beyond surface-level. Insights benefit linguists and tech developers, highlighting NLU's potential in enhancing language correction tools. In conclusion, this research deepens understanding of NLU's role in addressing linguistic intricacies, fostering nuanced communication in the digital era.

Table of Contents

List of Tables

A typical List of Tables content page looks like this

List of Figures

Abbreviation and Acronyms

GEC Grammatical Error Correction

Mlconv Multilayer Convolutional Encoder-Decoder Neural Network

NLU Natural Language Understanding

1. Introduction

This chapter conveys the importance of Grammatical Error Correction (GEC) automates rectifying errors in written text, primarily using statistical and deep learning methods. While well-explored in English and certain languages, Indic languages and low-resource contexts require more attention.

1.1 Introduction

Natural Language Understanding (NLU) is an AI discipline vital for machines to comprehend human language nuances. NLU analyses context, syntax, and semantics in text or speech, enabling machines to derive meaning accurately. This comprehension forms the basis for applications like chatbots, sentiment analysis, and language translation. NLU's role in enhancing human-computer interaction and automating language-related tasks with context-awareness becomes increasingly significant.

Leveraging Natural Language Understanding (NLU), error and grammar correction in Indian languages attains new levels of accuracy and contextual relevance. NLU-driven systems address the intricacies of morphological variations, syntactic structures, and contextual nuances, making them adept at rectifying errors while preserving intended meaning. While significant strides have been made in English, Indian languages' diverse linguistic landscape presents unique challenges that warrant tailored solutions. NLU-equipped error correction tools hold the potential to bridge the gap between human communication subtleties and automated correction, enabling effective communication in languages with complex

grammatical structures and contributing to improved human-computer interaction in India's linguistically rich digital landscape.

Grammatical Error Correction (GEC) is the process of automatically fixing errors in written text, spanning spelling, syntax, and coherence issues. Contemporary solutions for this challenge lean heavily towards statistical and deep learning methods, diverging from traditional rule-based approaches. At its core, GEC adopts a translation framework, aiming to convert inaccurately constructed sentences into linguistically correct counterparts within the same language. This task's successful execution hinges on the availability of a substantial corpus of annotated data, featuring sentence pairs representing erroneous versions and their corrected forms. Such data fuels the development of accurate and contextually-aware GEC systems, advancing the realm of automated language correction.

Although significant strides have been made in GEC research, the majority of these advancements have centred on English and a handful of well-resourced languages. Consequently, languages with fewer available resources, particularly Indic languages, have been relatively neglected. A few systems, such as UTTAM and SCMIL, have endeavoured to tackle spelling correction in Indic languages using probabilistic and deep learning methodologies, respectively. Additionally, simpler models based on n-grams have been applied to rectify errors commonly encountered in real-world contexts.

1.2 Motivation

The selection of Natural Language Understanding (NLU)-based grammar and spell correction, specifically targeting Indian languages, as a research topic is primarily driven by the notable lack of comprehensive solutions tailored to the intricacies of

these languages. Despite the linguistic richness and diversity that characterize Indian languages, they have unfortunately remained relatively marginalized within the domain of automated language correction systems. The predominant focus of research and development endeavour on English and a handful of more widely resourced languages has resulted in a glaring research gap, leaving Indian languages in need of specialized language correction tools.

The complex linguistic landscape of Indian languages, marked by intricate grammar rules, diverse morphological variations, and contextually nuanced structures, presents a unique set of challenges that require highly nuanced correction mechanisms. However, the scarcity of dedicated research in this realm has left these languages with suboptimal error correction capabilities. As digital communication continues to witness exponential growth across India, the demand for precise and context-aware grammar and spell correction tools becomes increasingly pressing. NLU-based methods offer a promising avenue to address these challenges effectively, as they possess the potential to capture the nuanced linguistic attributes of Indian languages and cater to their context-specific idiosyncrasies.

This research not only seeks to address a practical necessity but also contributes to the broader academic discourse in multiple ways. By embarking on an exploration of NLU-based grammar and spell correction within the unique linguistic context of Indian languages, the goal is to shed comprehensive light on the feasibility, effectiveness, and potential of such techniques for languages that have historically received insufficient attention in the technological landscape. The anticipated outcomes of this research extend beyond immediate language correction requirements; they hold the potential to drive broader advancements

in NLU research, with ripple effects felt across a wide array of applications in the realm of Indian languages.

Ultimately, this research endeavour underscores the urgency of cultivating a more inclusive approach to technology development that accommodates the rich linguistic diversity present in India. It highlights the significance of bridging the research gap to empower Indian languages with advanced language correction tools that are on par with those available for widely spoken languages. In doing so, the study aims to lay the foundation for a more equitable technological landscape where NLU-driven grammar and spell correction systems serve as facilitators of effective communication, linguistic preservation, and enhanced human-computer interaction within the complex fabric of Indian languages.

1.3 Background

There has been a growing wave of advancements in the domain of automatic spell and grammar correction within Indian languages, driven by the integration of Natural Language Understanding (NLU) techniques. While Indian languages have historically faced a lack of attention in this field, contemporary research has witnessed notable strides in addressing this gap. NLU-based approaches have emerged as a pivotal catalyst, enabling more sophisticated and context-aware error correction mechanisms.

One noteworthy trend in this evolution is the adaptation of state-of-the-art transformer-based models to the challenges of Indian languages' orthographic and grammatical intricacies. These models, pre-trained on large textual corpora, have demonstrated exceptional capacity in learning language nuances, allowing them to excel in error detection and correction tasks. Additionally, advancements in transfer learning techniques have facilitated the development of robust and

generalized models, which can be fine-tuned for specific Indian languages, thereby minimizing the data scarcity challenge that often plagues low-resource languages.

Furthermore, research efforts have been dedicated to curating and annotating extensive datasets in Indian languages, essential for training and evaluating NLU-driven error correction models. These datasets, comprising pairs of erroneous and corrected sentences, not only aid model development but also contribute to a better understanding of the language-specific error patterns. This development is complemented by the surge in open-source tools and resources that foster collaboration and accelerate research in this domain.

As NLU-powered spell and grammar correction systems continue to evolve, there is a growing emphasis on their capacity to comprehend contextual nuances for precise corrections. Techniques that incorporate syntactic and semantic context have emerged, allowing for more accurate error identification and meaningful corrections that align with the intended communication. These holistic approaches reflect the progress towards achieving more human-like error correction capabilities within Indian languages, which is integral to bridging the gap between human expression and machine comprehension.

The integration of NLU techniques has ignited a transformative journey in the realm of automatic spell and grammar correction within Indian languages. From leveraging advanced models to curating relevant datasets and embracing context-aware approaches, these developments collectively mark a significant step forward in enhancing the accuracy, effectiveness, and inclusiveness of language correction tools for the diverse linguistic landscape of India.

1.4.1 Integration of Transformer-Based Models: The adaptation of transformer-based models like BERT and its variations to Indian languages has revolutionized spell and grammar correction. These models, pre-trained on vast text data, exhibit exceptional understanding of language nuances, enabling accurate error detection and correction.

1.4.2. Transfer Learning Techniques: Advancements in transfer learning have facilitated the creation of versatile models. These models can be fine-tuned for specific Indian languages, mitigating data scarcity issues and allowing efficient error correction even for languages with limited resources.

1.4.3. Curated Language-Specific Datasets: The creation and curation of extensive datasets comprising pairs of erroneous and corrected sentences have played a pivotal role. These datasets serve as crucial resources for training NLU-driven error correction models, and they also shed light on language-specific error patterns.

1.4.4. Open-Source Tools and Resources: The proliferation of open-source tools and resources has fostered collaboration and accelerated research in this domain. Shared repositories of code, models, and datasets have facilitated knowledge sharing and collaborative efforts among researchers and developers.

1.4.5. Contextual Understanding: The latest developments emphasize context-awareness in error correction. Techniques that take into account syntactic and semantic context enable more accurate identification of errors and meaningful corrections that align with the intended message.

1.4.6. Linguistic Variation Handling: NLU models are increasingly designed to handle the vast linguistic variation present in Indian languages. This includes

addressing dialectal variations, morphological intricacies, and non-standard usages, resulting in more effective corrections.

1.4.7. Fine-Grained Error Categories: NLU-driven systems are capable of categorizing errors into finer groups, such as spelling, syntax, punctuation, and style. This finer granularity aids in providing more specific and appropriate corrections.

1.4.8. User Feedback Integration: Some systems incorporate user feedback loops to continuously improve error correction accuracy. By learning from user interactions, these systems adapt and refine their correction suggestions over time.

1.4.9. Hybrid Approaches: Hybrid models that combine rule-based and NLU-based methods are emerging. These models leverage the strengths of both approaches to achieve enhanced accuracy and coverage in error correction.

1.4.10. Real-Time Correction: The integration of NLU techniques has enabled real-time error correction in various applications, such as messaging platforms, word processors, and content management systems, enhancing the overall user experience.

1.4 Thesis origination:

Chapter 1: Introduction

This chapter introduces the research topic with the background and motivation to choose Indian languages for automatic spell and Grammer correction using NLU

Chapter 2: literature Review

This chapter describes the detailed literature survey on exciting research work about Automatic spell and Grammer correction using NLU in Indian languages. Summarize and analyse the various techniques, methodologies and discover the relevant publications.

Chapter 3: Problem Statement

This chapter describes the title, aim, objectives and detailed literature survey about the Automatic spell and Grammer correction using NLU in Indian languages.

Chapter 4: Feature Extraction

This chapter gives detailed information about creating the datasets and various methodologies used in the project along with the evaluation matrices used to evaluate the model.

Chapter 5: Result and Discussion

This chapter gives information about experimental results obtained after training and evaluating the model on the extracted datasets.

Chapter 6: Conclusion and Future Scope

This chapter gives summarize the main findings and future scope that can be applied to the project.

References

Provides the comprehensive list of all the sources cited throughout the thesis.

2. Literature Review and Problem Formulation

This chapter delves into foundational aspects of automatic spell and grammar correction using NLU in Indian languages. It establishes background theory, reviews existing literature, evaluates methodologies, and identifies opportunities for advancements. This sets the stage for comprehensive analysis of NLU-driven language correction within India's linguistic diversity.

2.1 Background Theory

2.1.1 Natural language understanding:

Natural Language Understanding (NLU) is a dynamic field within artificial intelligence that focuses on endowing machines with the ability to comprehend and interpret human language akin to human cognitive capabilities. Situated within the broader framework of natural language processing (NLP), NLU endeavours to bridge the gap between the intricacies of human language and the capabilities of machines. NLU's core objective is to equip computers with the capacity to grasp the intricacies of language, encompassing syntax, semantics, context, pragmatics, and even nuanced elements such as sarcasm and humour, across written and spoken forms of communication.

NLU comprises distinct layers of linguistic analysis that contribute to a holistic understanding of text or speech. Syntactic analysis deconstructs sentence structures and grammatical constituents, unveiling interrelationships between words. Semantic analysis delves into the meanings and interpretations of words and phrases, examining their contributions to overall comprehension. Contextual analysis considers the broader conversational or textual context, ensuring interpretations remain attuned to the

surrounding discourse. Pragmatic analysis contemplates implicit intentions, implicatures, and inferences, all crucial to facilitating meaningful communication.

The applications of NLU span a diverse spectrum. From chatbots and virtual assistants that offer seamless interactions, to sentiment analysis tools that gauge emotions in social media content, NLU has revolutionized human-computer interaction. Language translation systems have gained precision and context awareness, facilitating global communication. Moreover, NLU is pivotal in information extraction, summarization, and recommendation systems, where grasping underlying meaning is paramount. As technological progress continues, NLU's horizons expand, reshaping industries such as healthcare, finance, education, and entertainment. As NLU systems grow more advanced in capturing language subtleties, the boundary between human and machine communication blurs further, ushering in a transformative era of interaction and comprehension.

2.1.2 Grammatical Error Correction:

Grammatical error correction using Natural Language Understanding (NLU) represents a paradigm shift in automated language processing, aiming to enhance the accuracy and sophistication of error detection and correction. Traditional rule-based approaches and statistical methods often fall short in capturing the intricacies of human language due to the complexity and variability of grammar rules across languages. NLU techniques, however, harness the power of machine learning and neural networks to interpret context, semantics, and syntax, facilitating more contextually aware error correction.

At the core of NLU-based grammatical error correction lies the utilization of advanced machine learning models like transformer architectures, such as BERT (Bidirectional Encoder Representations from Transformers). These models are pre-trained on vast text corpora, learning language nuances and patterns. In error correction, these models can

identify deviations from grammatical norms and propose appropriate corrections, accounting for the surrounding context. NLU techniques also enable the recognition of subtle errors that may elude traditional rule-based systems, enhancing the overall accuracy of correction.

Additionally, NLU-based grammatical error correction can account for language-specific idiosyncrasies and variations that pose challenges to conventional methods. Languages like English and Hindi exhibit complex structures, morphological diversity, and contextual nuances. NLU models can be fine-tuned for these specific languages, improving error correction performance and accommodating their distinct grammar rules. Moreover, NLU-driven systems can handle a broader range of errors beyond spelling and grammar, encompassing issues like contextual coherence and stylistic inconsistencies.

The applications of NLU-based grammatical error correction span across diverse domains, including content creation, educational platforms, and professional communication. By providing contextually accurate corrections, NLU-driven systems improve the quality of written communication, ensuring messages are conveyed accurately while preserving the intended meaning. As technology continues to advance, the potential for NLU to revolutionize language correction remains immense, ultimately bridging the gap between human language complexity and machine comprehension.

2.1.3 Selection of language:

Choosing Hindi as the focal point for research on automatic spell and grammar correction utilizing Natural Language Understanding (NLU) is grounded in several compelling rationales. To begin with, Hindi occupies the position of the most widely spoken language in India, rendering it a pivotal candidate for language technology advancements that cater

to the majority of the population. Despite its significance, the landscape for NLU-based spell and grammar correction in Hindi remains underdeveloped, characterized by a scarcity of comprehensive datasets and previous research efforts.

The grammatical intricacies intrinsic to Hindi further reinforce its suitability as a research domain. With its complex verb conjugations, noun declensions, and intricate sentence structures, Hindi presents substantial challenges for automated correction systems. Directing attention towards these intricacies offers the potential to yield innovative solutions that bridge the gap between the nuances of Hindi grammar and the precision requisite for effective communication.

Sentence	Complexity
"राम ने खाना खाया।" (Ram ate food.)	Involves subject-object-verb order, verb conjugation
"मुझे किताब दी गई है।" (I was given a book.)	Depicts passive voice, tense agreement

Table1: Representing complexity of Hindi in different context

As demonstrated in the table, these sentences showcase the diverse grammatical facets of Hindi, spanning from verb conjugations to the construction of passive voice. This underscores the necessity for advanced NLU techniques to precisely detect and rectify errors while preserving intended meaning.

By concentrating on Hindi, this research endeavor seeks to address crucial gaps within the realm of language technology for Indian languages. The anticipated outcomes carry the potential to introduce innovative solutions that not only enhance communication efficacy in Hindi but also contribute meaningful insights to the broader landscape of NLU-based language correction.

The approach taken for grammatical error correction (GEC) using Natural Language Understanding (NLU) in the context of Hindi involves three key steps to create a robust and effective correction system.

Firstly, a parallel corpus of synthetic errors is generated by introducing errors into grammatically correct sentences through a rule-based procedure. This procedure specifically targets inflectional errors, which are common in languages like Hindi. The versatility of this process allows its extension to other Indic languages, amplifying the potential impact of the correction system. This synthetic error corpus serves as a controlled environment for training and refining the NLU-based GEC model, enabling it to learn the intricacies of Hindi grammar and improve error correction accuracy.

The second step involves the collection of Hindi edits from Wikipedia, constituting a smaller yet natural source of error instances. These edits are subsequently filtered to create a refined error corpus. This corpus extracted from authentic language usage provides an invaluable resource for evaluating the performance of the NLU-driven GEC system. Moreover, an extended version of the ERRANT toolkit is employed to analyze this corpus, offering insights into error patterns and further enhancing the system's efficacy.

In the final step, the performance of established GEC approaches, commonly applied to languages like English, is evaluated on the aforementioned datasets. This evaluation process yields initial GEC results specific to the Hindi language. The outcomes of this step

provide a baseline for measuring the effectiveness of the NLU-based GEC model in comparison to existing methodologies. These comprehensive evaluations serve as a vital benchmark for assessing the progress and capabilities of the NLU-driven GEC system tailored for Hindi.

2.2 Literature review

Sl no	Authors	Year of Publications	Methods and Methodologies	Research Findings	Conclusions drawn by authors	Limitation of the study	Critical Appraisal of the study
1	Abid Yahya	2022	Six machine learning techniques.	Provides 99.92 recognition accuracy which is at least 6% higher compared to existing works	This work deals with errors produced by students in essay	Limited vocabulary	Reference work but need more training data

2	Zhenhui He	2021	LSTM (long short-term memory networks)	The proposed system achieved a classifier accuracy of 89%.	Labeled training corpus created using bag of words and compared to existing models	Focuses on noun,pronoun and verbs	Lacks accuracy when used with n-grams

3	Ming-Wei Chang	2018	Transformers using sequence to sequence pre-training	Relatively new methord introduced	Large dataset used for higher accuracy	Reduced accuracy as n grams are increased	Good model with large pre-trained data

4	Alexis Conneau	2019	MLM,CLM,TLM(casual language models)	Multi-lingual language trainings explored	Training of multiple languages and show the effectiveness of cross-lingual pretraining	Effective NMT methords lacking	Cross-language training techniques
5	Myle Ott, Sergey Edunov Alexei Baevski,	2019	Pytorch	This paper deals with custom model trainig for GEC	Model training for summarization,NMT,Error correction	High resource comsumption	Helps in training custom models

6	Ankur Sonawane, Sujeet Kumar Vishwakarma, Bhavana Srivastava, Anil Kumar Singh	2020	ERRANT(ERRor ANnotation Toolkit)	The paper deals with creation on artificially induced error dataset	Using ERRANT to create artificially induced error in hindi and evaluating them	Translation errors	Helps in generating various datasets as limited datasets are available for Indian languages
7	Shamil Chollampatt,and Hwee Tou Ng	2018	Multilayer Convolutional Encoder-Decoder Neural Network	This paper deals with 2 dataset Lang-8 and NUCLE	This paper delas with a error correction methodology using pytorch	Focuses on limited pre-training techniques	Helps in fine tuning the model

8	Christopher Bryant, Mariano Felice, Ted Briscoe	2017	CoNLL(computational linguistic)	ERRANT toolkit introduction	automatically extract edits from parallel original and corrected sentences and classify them according to a new, dataset-agnostic, rule-based framework.	Rule definition is complex	Rule-specification limitatiom

9	Marcin Junczys-Dowmunt, Roman Grundkiewicz	2014	This paper uses SMT	Convolutional neural network	Model need to be customized, task-specific features should be introduced	Lack of pre-trained data	The work did not address the self similarity in characters.
10	Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, Jingming Liu	2019	This paper focuses on Encoder-decoder copy augmented architecture	Sequence-sequence model	Copying Mechanism introduced with sequence to sequence	Denoising parameter are difficult to impliment	This work is a milestone to GEC

Table 2: Summary of literature papers

2.3 Summary

In essence, the mentioned papers offer significant contributions to the domain of error and grammar correction employing Natural Language Understanding (NLU). They furnish essential perspectives on the current landscape and intricacies within this field. However,

it's noteworthy that certain limitations exist within these studies. Notably, there could be gaps in the coverage of certain aspects, potentially leaving certain areas unexplored. Additionally, the absence of intricate technical comparisons might hinder a comprehensive understanding of the methodologies' nuances and effectiveness. A pertinent aspect that warrants attention is the need for more robust real-world validation of the proposed techniques, which can ascertain their applicability in practical scenarios.

To surmount these limitations and foster progress, it's imperative that future research and development endeavors are undertaken. These efforts could encompass refining the coverage of the methods, bolstering technical comparisons to provide a clearer comparative picture, and embarking on comprehensive real-world validation studies. This iterative process of refinement will be pivotal in elevating the accuracy, efficacy, and real-world utility of the NLU-driven error and grammar correction techniques under consideration. By addressing these limitations head-on, the field can forge ahead, fostering enhanced understanding and practical applications of NLU within the realm of language correction.

2.4 Problem formulation:

The intricacies of Hindi's complex morphological changes and grammatical intricacies, as outlined above, highlight the compelling need to undertake research utilizing Natural Language Understanding (NLU). Hindi's status as India's predominant language, spoken by a substantial portion of the population, accentuates the urgency of addressing its unique grammatical challenges. Traditional rule-based and statistical methods often struggle to capture the diverse morphological variations and intricate agreement patterns inherent in the language. This is where the transformative potential of NLU becomes apparent.

NLU, powered by advanced machine learning models like transformers, holds promise in comprehending and interpreting Hindi's nuanced grammatical terrain. By training these models on extensive datasets containing both grammatically accurate and erroneous sentences, they can learn the intricate patterns of inflection and agreement that underlie precise language usage. Moreover, NLU-driven systems possess the ability to grasp context, semantics, and syntax, enabling them to differentiate intentional stylistic choices from genuine grammatical errors – a nuance often missed by rule-based systems.

Considering Hindi's significance in education, governance, culture, and commerce in India, advancing NLU-driven error correction techniques is pivotal for fostering literacy, effective communication, and information sharing. The language's rich diversity of linguistic styles, dialects, and contexts necessitates an approach that flexibly adapts to the multifaceted intricacies while ensuring grammatical precision. Hence, research in Hindi via NLU is not just an academic endeavor; it stands as a pragmatic and societal imperative, bridging the divide between human expression and machine comprehension to facilitate precise and contextually pertinent communication in this multifarious language.

3. Problem Statement

This chapter serves as an introduction to the focal point and goals of the dissertation. Its purpose is to elucidate the importance of the subject matter and lay the groundwork for subsequent discussions. The primary objectives encompass delineating the dissertation's purpose, elucidating research objectives, and outlining the study's boundaries. Through the attainment of these objectives, the chapter establishes a fundamental comprehension of the problem at hand.

3.1 Title

"Automatic spelling and Grammer correction using NLU in Indian languages"

3.2 Aim

The primary objective of this research is to leverage Natural Language Understanding (NLU) for the advancement of automated spelling and grammar correction in Indian languages. The study aims to address the complexities intrinsic to languages like Hindi by harnessing NLU's contextual comprehension capabilities. Through the utilization of advanced machine learning models, such as transformers, the research endeavors to enhance the accuracy and efficiency of error correction systems. By delving into the nuances of grammar, syntax, and linguistic structures specific to Indian languages, this investigation seeks to contribute to the development of more sophisticated and culturally tailored language processing technologies. Ultimately, the research strives to bridge the gap between linguistic intricacies and machine comprehension, enriching communication and language-related applications across diverse domains.

3.3 Objectives

- To perform literature review of publications and journals related to automatic spell and grammar correction Data Acquisition
- To acquire reliable data from various sources
- To induce artificial errors to the acquired dataset
- To train the dataset with proposed model using deep-learning
- To evaluate the model using various benchmarks

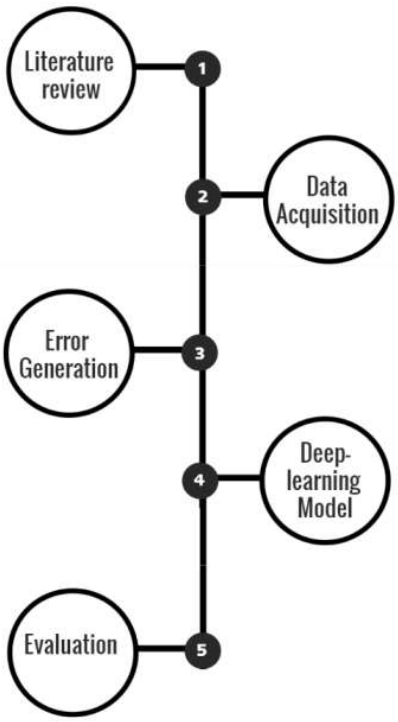


Fig 1: General working of thesis

Objective No.	Statement of the Objective	Method/ Methodology	Resources Utilised
1	To perform literature review of publications and journals related to automatic spell and grammar correction	Manual reading	Publications and Journals
2	To acquire reliable data from various sources	WikiEdits 2.0	Public Dataset and Open source web -documents
3	To induce artificial errors to the acquired dataset	ERRANT	ERRor ANotation Toolkit
4	To train the dataset with proposed model using deep-learning	Fairseq, Multilayer-convolutional encoder-decoder model	Neural networks
5	To evaluate the model using various benchmarks	GLUE score(0.5) score	Huggingface GLEU

Table 3 :Method and methodology to achieve each objective

3.4 Scope of present Investigation:

The scope of the present investigation in the domain of automatic spell and grammar correction using NLU in Hindi encompasses several key aspects.

1. Methodology Exploration: Delve into the application of NLU techniques, particularly deep learning models, to address the challenges of grammatical error correction in Hindi. The focus is on harnessing the contextual understanding of NLU to enhance the precision of error identification and correction.

2. Hindi Language Complexity: Examine the intricacies of Hindi grammar, including its unique morphological features, inflectional patterns, and agreement rules. The research aims to tailor NLU methods to navigate the complexity of Hindi's linguistic structure.

3. Dataset Creation: Curate a comprehensive and diverse dataset containing grammatically correct sentences and corresponding error instances in Hindi. This dataset will serve as the foundation for training and evaluating the NLU-driven correction model.

4. Error Identification and Classification: Develop algorithms and methodologies to effectively identify and classify various types of errors in Hindi text, encompassing spelling mistakes, syntax errors, morphological inaccuracies, and contextual inconsistencies.

5. NLU Model Training and Optimization: Implement and train deep learning models, such as transformer architectures, to understand and rectify grammatical errors in Hindi sentences. Fine-tuning these models for Hindi grammar intricacies is crucial for optimal performance.

6. Evaluation Metrics and Benchmarking: Define suitable evaluation metrics to measure the accuracy, efficiency, and context-sensitivity of the NLU-based correction model. Compare its performance against existing methods and benchmarks.

7. Real-World Applicability: Investigate the feasibility of integrating the developed NLU-driven correction model into practical applications, such as text editors, educational platforms, and communication tools, to enhance language accuracy in real-world scenarios.

8. Limitations and Future Directions: Recognize the limitations of the proposed methodology and discuss potential avenues for further improvement. Address challenges related to varying text domains, dialects, and linguistic nuances within Hindi.

In essence, the scope of the present investigation encompasses the entire lifecycle of research, from exploring the theoretical foundations of NLU to implementing and fine-tuning models, creating datasets, and evaluating the system's efficacy in real-world scenarios. The research strives to contribute to the advancement of automated spelling and grammar correction in Hindi by leveraging the power of NLU-driven techniques.

4. Problem Solving

This chapter deals with in-depth working of various techniques used in acquiring the results. It delas with various methods used to acquire data, architecture of model used with its working. It also discusses with the various evaluation matrices used to evaluate the model.

4.1 Dataset

4.1.1 The WikiEdits

The WikiEdits 2.02 , introduced by Grundkiewicz and Junczys-Dowmunt in 2014, is a significant asset in the realm of language processing, particularly for error correction. This dataset contains pairs of sentences from Wikipedia revisions, capturing original and edited versions. It facilitates the examination of grammatical errors, syntactic alterations, and language subtleties introduced during editing.

WikiEdits 2.02 employs a methodology involving sentence alignment before and after edits to create parallel pairs. This alignment enables analysis of specific changes, offering insights into prevalent errors in natural language. Its primary utility lies in evaluating the performance of various language correction models. Researchers can gauge their models against known edits in the dataset, enhancing the accuracy of error detection and correction systems. Moreover, it serves as a training resource for refining such systems, enabling exposure to real-world editing scenarios.

Beyond model evaluation, the dataset aids linguistic research by unveiling error patterns across languages, genres, and styles. This information guides the development of

interventions to enhance language proficiency and writing quality. In essence, WikiEdits 2.02 is an invaluable tool that advances language correction models and fosters a deeper comprehension of grammatical errors, accentuating its significance in the realm of language processing.

To ensure the quality of the dataset, certain filtering criteria are applied. Extracted sentences are confined to a length ranging from 6 to 27 tokens. Specifically, only substitution operations with a token-based Levenshtein edit distance below 0.3 are retained. Edits solely involving changes in punctuation or numbers, as well as corrections involving exceedingly rare tokens or HTML elements, are excluded. Additionally, edits linked to vandalism are removed. These filtering measures guarantee that the dataset includes meaningful linguistic modifications, omitting instances that might not contribute significantly to the analysis or evaluation objectives.

4.2 Artificial error generation

4.2.1 ERRor ANotation Toolkit (ERRANT)

The ERRor ANotation Toolkit (ERRANT), introduced by Bryant et al. in 2017 and referenced by Felice et al. in 2016, is a crucial resource for error annotation and grammatical analysis. ERRANT automates error identification and categorization within text, benefiting researchers and linguists by evaluating language correction systems and language proficiency.

ERRANT's operation involves aligning original and corrected sentences, identifying altered tokens. It then applies predefined rules to categorize errors, including verb tense, spelling, punctuation, and agreement, based on linguistic theories for accurate interpretation.

One of ERRANT's strengths is its adaptability, allowing users to define custom rules for specific error categories or nuances. This flexibility enables ERRANT to handle diverse linguistic complexities across languages and text types. Additionally, ERRANT generates comprehensive reports detailing error frequency, distribution, and linguistic patterns, aiding researchers and developers in understanding and improving their systems.

Beyond research, ERRANT contributes to advancing grammar and spelling correction systems and aiding educators in analysing student writing errors. In essence, ERRor ANotation Toolkit (ERRANT) plays a pivotal role in linguistic analysis and error correction. Its methodical approach empowers researchers, developers, and educators to delve into language errors, thereby refining language correction systems and enhancing linguistic understanding.

While the criteria for classification encompass various exceptional instances, the fundamental approach is as follows:

1. The Part-of-Speech (POS) tags and lemmas for the tokens are acquired using the StanfordNLP tagger . The determination of the error category based on POS tag comparison for the edit follows this procedure.
2. Edits that exhibit matching lemmas and POS tags are grouped under the classification of ":INFL" errors, signifying grammatical nature. Additionally, in the case of verbs, a supplementary category, designated as ":VERB:FORM," is introduced to encompass tense-related distinctions.
3. Edits that involve dissimilar lemmas but share identical POS are categorized
4. errors. The majority of these cases entail basic semantic modifications, typically consisting of word replacements, such as synonyms.
5. Edits exhibiting matching stems are labeled as "MORPH errors."

6. Edits that demonstrate a minimal edit distance are designated as "SPELL errors," while those that do not meet this proximity threshold are categorized as "OTHER."

4.2.2 WikiExtractor

WikiExtractor is a GitHub repository housing a versatile tool designed to extract content from Wikipedia articles. This code plays a pivotal role in simplifying the process of parsing and handling the extensive textual data present in Wikipedia pages. Its primary objective is to empower researchers, developers, and data enthusiasts to efficiently access and utilize the vast knowledge repository that Wikipedia offers.

The working of WikiExtractor involves several key steps:

1. Article Retrieval: WikiExtractor accesses Wikipedia articles from the Wikipedia dump files, which are periodically released snapshots of Wikipedia's content.

2. Parsing and Cleaning: The tool parses the articles, removing any irrelevant or unwanted markup, templates, and special characters. This step ensures that the extracted content is clean and suitable for analysis.

3. Text Extraction: WikiExtractor then extracts the main textual content of the article, excluding auxiliary information like metadata, references, and citations.

4. Language Processing: The tool supports multiple languages and handles linguistic complexities like tokenization, sentence splitting, and part-of-speech tagging. This enables users to perform language-specific analyses.

5. Output Formats: The extracted content can be outputted in various formats, such as plain text or JSON, facilitating seamless integration into downstream analyses and applications.

6. Customization: WikiExtractor offers customization options to tailor the extraction process to specific requirements. Users can configure parameters to control the level of detail extracted and the formatting of the output.

Inflectional errors, which are easily recognizable and common in Hindi, were chosen as the basis for generating a synthetic dataset. The approach involved the following steps:

1. Data Collection: Sentences were sourced from the June 20, 2023 revision of Hindi Wikipedia using the WikiExtractor tool. This selection was based on the assumption that recent versions are more likely to be grammatically accurate.

2. Tokenization and POS Tagging Extracted sentences were broken down into individual words through tokenization. The Hindi POS Tagger was applied for part-of-speech tagging, revealing the grammatical roles of words.

3. Modification of Inflections: Words belonging to the VERB, ADP, ADV, and PRON categories were targeted for inflectional ending adjustments. Inflections were altered by substituting different random endings from the relevant inflection table for that specific POS. Care was taken to address exceptions and irregular cases

4. Error Creation: Each inflectional alteration resulted in the creation of an "edit" containing a single incorrect word. This process generated pairs of sentences, one accurate and the other containing a solitary error (illustrated in Figure 2).

5. Data Filtering: Approximately 40% of the generated sentence pairs were randomly removed from the dataset to ensure diversity.

6. Partitioning: The remaining sentence pairs were divided into two sets - 80% for training and 20% for validation. Sentences originating from the same source sentence were kept together to maintain coherence within each set.

By following this method, a synthetic dataset was systematically constructed, focusing on inflectional errors. This approach facilitated controlled error introduction, enabling the training and evaluation of NLU-based systems for automated error correction in Hindi sentences.

4.3 Models

"Multilayer Convolutional Encoder-Decoder Neural Network"

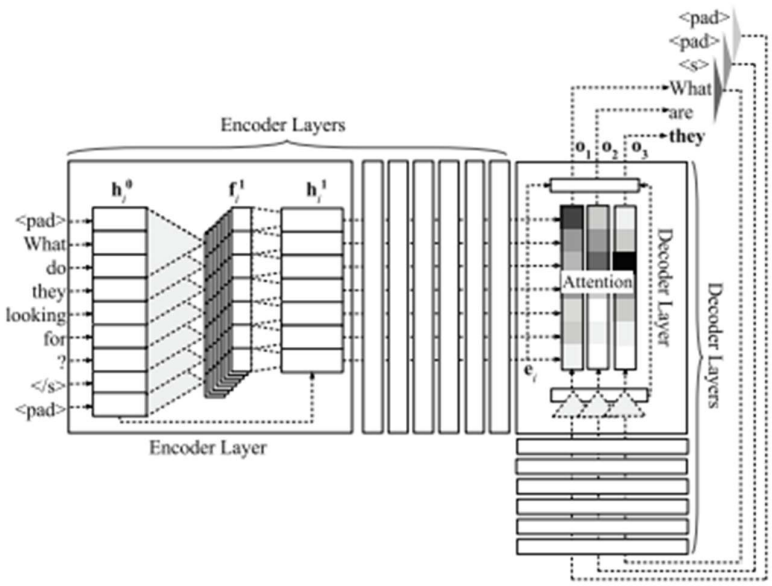


Fig 2: Architecture of Multilayer Convolutional Encoder-Decoder Neural Network

The input source sentence, denoted as S, comprises a sequence of m source tokens (s1, ..., sm), with sm serving as an end-of-sentence marker. These tokens are embedded in a continuous space using word embeddings w(si) and position embeddings p(i). The encoder and decoder are composed of L layers, and the network architecture involves linearly mapping the source token embeddings to input vectors for the initial encoder layer (h01, ..., h0m). This mapping is achieved through multiplication with weights matrix W ∈ Rh×d and addition of biases b ∈ Rh, contributing to the network's processing and transformation of source sentences for subsequent stages.

$$\mathbf{h}_i^0 = \mathbf{W}\mathbf{s}_i + \mathbf{b} \qquad (1)$$

During the initial encoder layer, a set of 2h convolutional filters, each with a dimension of 3 × h, processes sequences of three consecutive input vectors. This processing yields a

feature vector denoted as f1i ∈ R2h for each sequence. To maintain consistency in the number of output vectors as source tokens, paddings represented by "<pad>" in Figure 1 are introduced at the start and end of the source sentence before the convolutional operations are applied.

$$\mathbf{f}_i^1 = \text{Conv}(\mathbf{h}_{i-1}^0, \mathbf{h}_i^0, \mathbf{h}_{i+1}^0) \tag{2}$$

The convolution operation is represented by Conv(·). Subsequently, a non-linearity is applied using gated linear units (GLU).

$$\text{GLU}(\mathbf{f}_i^1) = \mathbf{f}_{i,1:h}^1 \circ \sigma(\mathbf{f}_{i,h+1:2h}^1) \tag{3}$$

The result of applying GLU to f1i, denoted as GLU(f1i), produces a vector in Rh. Here, ◦ and σ represent element-wise multiplication and sigmoid activation functions, respectively. Furthermore, f1i,u:v denotes the elements of f1i from indices u to v, inclusive. To preserve the input's information, residual connections are established by adding the input vectors to the output vectors of an encoder layer.

The output vectors of the lth encoder layer are expressed as:

$$\mathbf{h}_i^l = \text{GLU}(\mathbf{f}_i^l) + \mathbf{h}_i^{l-1} \qquad i = 1, \ldots, m \tag{4}$$

Each output vector of the final encoder layer, hL i ∈ Rh, is linearly mapped to get the encoder output vector, ei ∈ Rd, using weights We ∈ Rd×h and biases be ∈ Rd:

$$\mathbf{e}_i = \mathbf{W}_\mathbf{e}\mathbf{h}_i^L + \mathbf{b}_\mathbf{e} \qquad i = 1, \ldots, m \tag{5}$$

During decoding, the target word tn is generated at the nth time step, considering n - 1 previously generated target words. Padding tokens, including the beginning-of-sentence marker and prior generated tokens (t-2, t-1, t0, t1, ..., tn-1), are embedded similarly to source tokens. These embeddings are linearly transformed to g0j ∈ Rh and inputted to the initial decoder layer. Each decoder layer conducts convolution operations followed by non-linearities on the previous layer's output vectors gl−1j, where j = 1 to n. This iterative process within decoder layers contributes to the overall decoding process.

$$\mathbf{y}_j^l = \mathrm{GLU}(\mathrm{Conv}(\mathbf{g}_{j-3}^{l-1}, \mathbf{g}_{j-2}^{l-1}, \mathbf{g}_{j-1}^{l-1}) \quad (6)$$

The attention weights αln,i are calculated through the dot product of the encoder output vectors e1, ..., em with zln. This calculation is followed by normalization through a softmax operation:

$$\alpha_{n,i}^l = \frac{\exp(\mathbf{e}_i^\top \mathbf{z}_n^l)}{\sum_{k=1}^m \exp(\mathbf{e}_k^\top \mathbf{z}_n^l)} \qquad i = 1, \ldots, m \quad (7)$$

The last decoder layer's output vector, gLn, is linearly transformed into dn ∈ Rd. Dropout is employed on decoder outputs, embeddings, and preceding every encoder and decoder layer. Following this, the decoder output vector undergoes mapping to match the target vocabulary size (|Vt|). The softmax operation is then executed to compute target word probabilities. This sequence of transformations and operations is pivotal in generating probabilities for target words during the decoding process.

$$\mathbf{o}_n = \mathbf{W}_\mathbf{o}\mathbf{d}_n + \mathbf{b}_\mathbf{o} \qquad \mathbf{W}_\mathbf{o} \in \mathbb{R}^{|V_t| \times d}, \mathbf{b}_\mathbf{o} \in \mathbb{R}^{|V_t|} \quad (8)$$

$$p(t_n = w_i | t_1, \ldots, t_{n-1}, S) = \frac{\exp(o_{n,i})}{\sum_{k=1}^{|V_t|} \exp(o_{n,k})} \quad (9)$$

The working of the proposed method involves the integration of two crucial components: a multilayer convolutional neural network (CNN) and an encoder-decoder framework. This innovative fusion aims to leverage the strengths of both structures, enhancing their collective capability for grammatical error correction.

In the first stage, the multilayer convolutional neural network is responsible for processing the input sentences. This CNN component employs multiple layers of convolutional and pooling operations to extract high-level features from the input text. These features serve as a compressed representation of the input sentences, capturing important linguistic cues and patterns related to grammatical errors.

The encoder-decoder framework comes into play next, focusing on error correction. The encoder takes the compressed sentence representation generated by the CNN and encodes it into a fixed-length vector, which effectively captures the contextual information. The decoder then utilizes this vector to generate a corrected version of the input sentence, considering grammatical improvements.

The architecture's unique combination enables it to address both local and global grammatical errors effectively. The multilayer CNN excels at capturing local patterns and dependencies, while the encoder-decoder component comprehends the broader context to produce coherent corrections.

The proposed approach was evaluated on various benchmark datasets for grammatical error correction, demonstrating its superior performance compared to existing methods. The experimental results underscored the effectiveness of leveraging the combined power of multilayer convolutional and encoder-decoder architectures.

In conclusion Multilayer Convolutional Encoder-Decoder Neural Network. By synergizing these architectures, the method effectively addresses the intricacies of grammatical errors at both local and global levels, showcasing its potential to advance the field of natural language processing and automated error correction.

“Copy augmented transformer model”

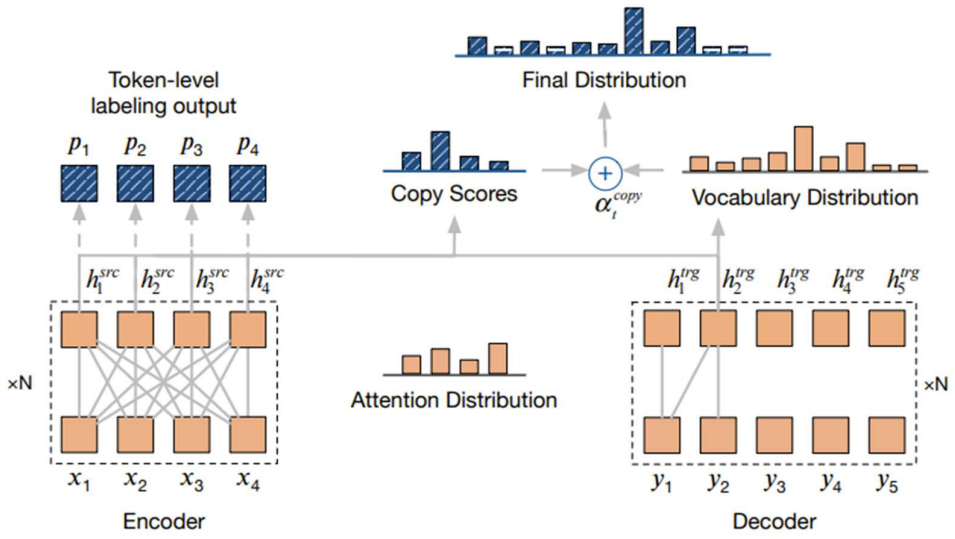


Fig 3 : Architecture of Copy augmented transformer model

Encoder-decoder

The Transformer architecture employs a stack of L identical blocks to encode the source sentence and decode the target sentence. Each block comprises multi-head self-attention applied to the source tokens, followed by position-wise feedforward layers. This produces a context-aware hidden state for each block. The decoder also consists of L identical blocks

with multi-head attention and feed-forward networks. Additionally, the decoder block incorporates an extra attention layer over the encoder's hidden states.

The objective of the model is to predict the next word in a sequence of target word tokens (y1, ..., yT) based on the source word tokens (x1, ..., xN). The encoder's hidden states are computed using the source tokens, and the decoder's hidden states are computed using the previously generated target words and the encoder's hidden states. The probability distribution of the next word is obtained by applying the softmax operation to the inner product between the target hidden state and the word embedding matrix L, where L is the word embedding matrix with dimensions dx × |V| (word embedding dimension × vocabulary size).

The training process involves calculating the cross-entropy loss for each position during decoding. The loss lce for each training example is the accumulation of the cross-entropy loss for each position in the generated sequence. It quantifies the difference between the predicted probability distribution of the next word and the actual distribution, measuring the model's performance in generating coherent and accurate sequences.

In essence, the Transformer architecture utilizes multi-head self-attention, feed-forward layers, and a sophisticated decoding mechanism to produce context-aware hidden states for both source and target sentences. By optimizing the cross-entropy loss, the model learns to generate sequences that closely match the target sentences, contributing to its capability in tasks like machine translation and grammatical error correction.

Copying Mechanism:

The novel architecture introduces a refined approach to generate the generation probability distribution. This is accomplished by deriving the target hidden state in the base model. Additionally, the new architecture introduces a copying score mechanism

that involves an innovative attention distribution between the current hidden state of the decoder (htrg) and the encoder's hidden states (Hsrc or hsrc1...N). The copy attention mechanism is calculated using the same methodology as the encoder-decoder attentions.

In essence, this architecture enhances the model's ability to predict the generation probability distribution by incorporating the target hidden state generation process. Moreover, it introduces a copying score mechanism that leverages an attention distribution between the decoder's current state and the encoder's hidden states, enabling the model to better capture relevant information from the source input tokens. This attention mechanism is crucial for tasks like information retrieval and context preservation during the generation of outputs. The architecture's innovative integration of these elements aims to improve the accuracy and coherence of generated sequences in tasks such as machine translation and grammatical error correction.

$$q_t, K, V = h_t^{trg} W_q^T, H^{src} W_k^T, H^{src} W_v^T \tag{10}$$

$$A_t = q_t^T K \tag{11}$$

$$P_t^{copy}(w) = softmax(A_t) \tag{12}$$

The qt, K, and V represent the query, key, and value components required for the computation of the attention distribution and the copy hidden state. These components are fundamental in calculating the mechanisms that determine the model's focus and generate the copy scores.

The copy hidden state, on the other hand, involves utilizing the query (qt), key (K), and value (V) components to estimate a balancing factor denoted as α_copy t. This factor plays a critical role in maintaining a balance between generating new tokens and copying tokens directly from the input. The copy mechanism aims to incorporate relevant words from the source input, enhancing the coherence and accuracy of the generated output.

$$\alpha_t^{copy} = sigmoid(W^T \sum (A_t^T \cdot V)) \tag{13}$$

The qt, K, and V components play pivotal roles in calculating attention distribution and copy hidden states. These mechanisms are integral to the model's ability to focus on relevant information, generate accurate copy scores, and produce contextually coherent outputs that balance the incorporation of new tokens and copying from the source input. The core working mechanism of this method involves two main stages: pre-training and fine-tuning. During pre-training, a neural network model is exposed to a substantial amount of unlabelled text data. This exposure enables the model to capture essential linguistic patterns and structures present in the data. Additionally, a copy mechanism is integrated into the architecture, allowing the model to retain original words during the correction process.

In the subsequent fine-tuning phase, the model is refined using labelled data containing both erroneous and correct sentences. This phase fine-tunes the model's learned representations to specifically target grammatical error correction. The copy mechanism plays a crucial role in ensuring that accurate words are retained during correction, thereby enhancing the overall coherence and fluency of the corrected sentences.

By combining the power of pre-training with a copy-augmented architecture, the proposed method effectively captures linguistic nuances and error patterns, enabling it to generate contextually accurate corrections. The research demonstrates that this approach surpasses existing methods in terms of grammatical error correction accuracy, showcasing the potential of harnessing unlabelled data to enhance the error correction capabilities of automated systems.

In conclusion, the study presents a groundbreaking approach to enhancing grammatical error correction through the strategic use of pre-training and a copy-augmented architecture. By exploiting unlabelled data to enrich the system's comprehension of linguistic structures and incorporating a copy mechanism for precise error correction, the method represents a significant advancement in the realm of automated grammatical error correction.

In the experimental phase, I initially assess the system's feasibility using a fundamental transformer architecture implemented through the Tensor2Tensor library. I establish a baseline by utilizing the transformer base configuration. I then proceed to evaluate slightly modified versions of two prominent models in the realm of English Grammar Error Correction (GEC). To begin, I adapt the multi-layer convolutional encoder-decoder model training it for 5 epochs with carefully chosen hyperparameters. Subsequently, the focus shifts to training the copy augmented transformer model Notably, I skip the pretraining phase involving the denoising auto-encoder and directly train the system for a total of 9 epochs. These extensive experiments provide insights into the system's performance across distinct model architectures and parameter configurations, thereby contributing to the advancement of Grammar Error Correction (GEC) models.

4.3.1 Fairseq toolkit

Fairseq is a versatile open-source framework for sequence-to-sequence tasks in natural language processing (NLP) and machine translation, developed by Facebook AI Research (FAIR). Renowned for its flexibility and performance, Fairseq empowers researchers and practitioners to build and experiment with state-of-the-art models for a wide range of tasks.

At its core, Fairseq embodies a modular and extensible architecture that facilitates the implementation and training of various sequence-to-sequence models. It supports a plethora of model architectures, such as convolutional and recurrent neural networks, as well as transformers. This flexibility enables researchers to select the most suitable architecture for their specific task.

Fairseq's flexibility extends to its data processing pipeline, allowing seamless integration of custom tokenization, vocabulary handling, and data augmentation techniques. This empowers users to tailor the data preprocessing steps according to the specific requirements of their tasks and datasets.

One of the standout features of Fairseq is its ease of experimentation with new architectures and techniques. Researchers can quickly prototype novel models by leveraging the framework's pre-built components, enabling rapid iteration and exploration. Moreover, Fairseq's integration with popular deep learning libraries such as PyTorch ensures efficient model training and compatibility with various hardware configurations.

Another notable aspect of Fairseq is its commitment to reproducibility and benchmarking. The framework provides pre-trained models and benchmark datasets, enabling users to assess their models' performance against established baselines. This emphasis on

reproducibility fosters a culture of transparency and accountability within the NLP research community.

While Fairseq is primarily recognized for its versatility in sequence-to-sequence tasks, its adaptability can also be harnessed for automatic spell and grammar correction using NLU in Indian languages. Leveraging Fairseq's modular architecture and customizable data processing pipeline, researchers and practitioners can tailor the framework to address the specific challenges posed by grammatical variations in Indian languages.

Fairseq's support for various model architectures, including transformers, can be particularly advantageous in capturing the complex grammatical structures inherent in Indian languages. By customizing the tokenization and preprocessing steps, the framework can be fine-tuned to accommodate the unique characteristics of different languages, such as morphological intricacies and sentence variations.

The flexibility of Fairseq extends to incorporating linguistic and contextual knowledge relevant to Indian languages. By integrating domain-specific language models or incorporating linguistic rules, the model's accuracy in error detection and correction can be enhanced. Additionally, Fairseq's compatibility with pre-trained embeddings enables the utilization of language-specific embeddings that capture the nuances of Indian languages.

Furthermore, Fairseq's emphasis on benchmarking and reproducibility can be beneficial for evaluating the performance of automatic spell and grammar correction models in Indian languages. Researchers can utilize benchmark datasets and pre-trained models to assess their models' effectiveness in comparison to established baselines.

In conclusion, while Fairseq is not specifically designed for spell and grammar correction, its adaptability, support for various model architectures, and customizability make it a

promising platform for developing effective solutions for automatic spell and grammar correction using NLU in Indian languages. By tailoring Fairseq to the linguistic intricacies of Indian languages and harnessing its strengths, researchers can contribute to the advancement of accurate and contextually aware error correction tools for these languages.

4.4 Evaluation:

4.4.1 Glue score:

The General Language Understanding Evaluation (GLUE) score is a widely adopted benchmark in the field of natural language processing (NLP) that measures the performance of various language understanding models across a diverse set of tasks. Introduced by Wang et al. in 2018, the GLUE score aims to provide a comprehensive assessment of NLP models by evaluating their generalization and robustness across multiple tasks, ranging from sentence-level tasks like classification to more complex language understanding tasks.

The working of the GLUE score involves selecting a set of benchmark tasks, each of which comes with its own dataset and evaluation metric. The tasks are carefully chosen to represent a broad spectrum of linguistic phenomena and reasoning abilities. Some examples of GLUE tasks include sentence pair classification, sentence completion, and textual entailment. Models are trained on these tasks and evaluated using their respective evaluation metrics.

To calculate the GLUE score, individual task scores are averaged across all benchmark tasks. It's important to note that not all tasks contribute equally to the final score; tasks with more diverse linguistic challenges are given higher weight to ensure a balanced assessment of the model's performance. This weighted average of task scores results in

the overall GLUE score, which provides a single numerical value representing the model's overall performance on a range of language understanding tasks.

The GLUE score offers several benefits. It encourages the development of models that are not specialized for a single task but rather demonstrate general language understanding capabilities. It also fosters model interpretability and transparency, as participants are required to share their model's architecture, making the evaluation process more credible. However, it's worth noting that the GLUE score has received criticism for its potential limitations, such as the focus on surface-level linguistic patterns and the lack of consideration for model biases.

In conclusion, the GLUE score serves as a valuable tool for evaluating the performance of NLP models on diverse language understanding tasks. By encompassing various linguistic phenomena and challenges, it promotes the development of models that can generalize and perform well across different language tasks, driving advancements in the field of natural language processing.

4.4.2 F-beta score:

The F-beta score is a widely used evaluation metric in machine learning and information retrieval that assesses a model's performance on imbalanced datasets by striking a balance between precision and recall. It is an extension of the F1 score, which considers equal importance for precision and recall. The F-beta score introduces a parameter "beta" to emphasize either precision or recall, depending on the task's requirements.

The formula for calculating the F-beta score is as follows:

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

(14)

Here, precision is the ratio of true positive predictions to the total predicted positives, and recall is the ratio of true positive predictions to the total actual positives.

The parameter "beta" determines the trade-off between precision and recall. When beta is set to 1, the F-beta score is equivalent to the F1 score, providing equal weight to precision and recall. A higher beta value gives more weight to recall, making the metric more sensitive to false negatives. Conversely, a lower beta value gives more weight to precision, making the metric more sensitive to false positives.

The F-beta score is particularly valuable in situations where the cost of false positives or false negatives varies significantly. For instance, in medical diagnoses, where missing a positive case (false negative) might have serious consequences, a higher beta value might be appropriate to prioritize recall. In contrast, in applications like spam detection, where false positives are undesirable, a lower beta value might be chosen to prioritize precision.

5. Results and Discussions

In the dataset I Keeping all sentences generated from a particular correct sentence in the same partition, I split the obtained dataset into train(80%) and valid(20%) partitions.

5.1 Dataset :

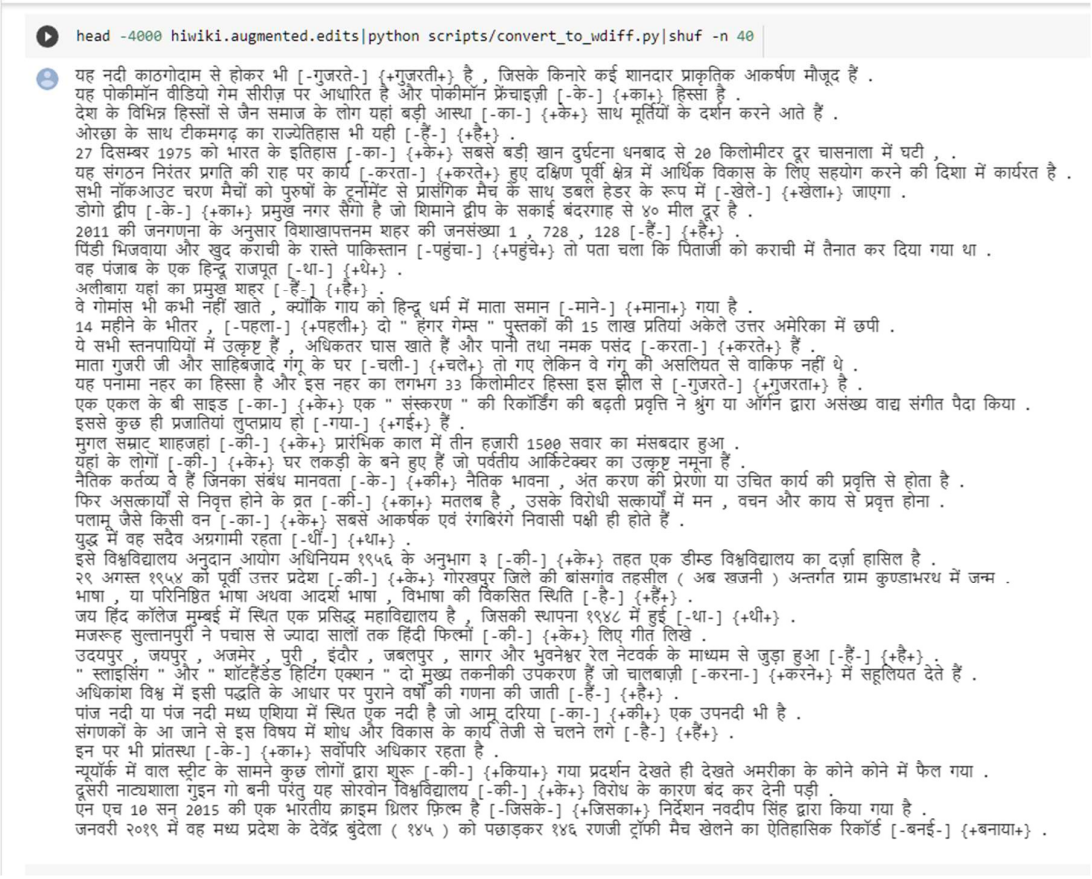


Fig 4: Extracted Dataset

We adapted the tool for Hindi and curated the "HiWikEd" dataset from a Wikipedia dump. Edits were filtered based on sentence length (6-27 tokens) and token-based Levenshtein

edit distance (<0.3) for substitution operations. I excluded edits involving punctuation, numbers, rare tokens, and HTML markups, as well as those related to vandalism. This dataset, tailored for Hindi, lays the groundwork for robust error correction and grammatical analysis in the language.

5.2 Error generation:

```
[ ]  wc -l  hiwiki.extracted.edits hiwiki.extracted.clean.edits
     head -4000 hiwiki.extracted.clean.edits|python scripts/convert_to_wdiff.py|shuf -n 40
```

महाराणा [-जयसिंह-] {+राजसिंह+} ने इस झील का निर्माण 17वीं शताब्दी में गोमती नदी पर [-डैम-] {+बांध+} बनाकर किया था .
इस [-समें-] {+समय+} पीटर हूटन इसके मुख्य कार्यकारी अधिकारी है .
आयरिश आप्रवासी उत्तरी अमेरिका के लिए उन्नीसवीं शताब्दी में परम्परा के संस्करणों किया , हैलोवीन पश्चिमी [-देशो-] {+देशों+} में मनाया जाता है .
यह उद्यान आगरा शहर में [-ताजमहल-] {+ताजमहल+} से २ ३४ किलोमीटर दूर उत्तर दिशा में स्थित है .
समाधि लगभग [-1-] {+१+} मीटर ऊँचे मंच पर निर्मित , और इसका प्रवेश द्वार पांच सीढ़ी ऊपर है .
और पोगो के इतने लोकप्रिय होने का एक और भी [-कारन-] {+कारण+} है , छोटा भीम .
मुकेश अम्बानी की पत्नी नीता [-अम्बानी-] {+अंबानी+} रिलायंस इंडस्ट्रीज के सामाजिक एवं धर्मार्थ कार्यो को देखती हैं .
इस मूर्ति के ऊपर [-जङ-] {+जड़ा+} हुआ हीरा मूर्ति के अनेक रुपों का दर्शन करवाता है .
इनके अब तक [-2-] {+3+} उपन्यास , एक काव्य संग्रह , दो गजल संग्रह , दो संपादित पुस्तक और एक ब्लॉगिंग का इतिहास प्रकाशित है .
ये [-गुजरात-] {+तमिलनाडु+} राज्य से हैं .
अम्बिकापुर से 40 [-किमी-] {+किलोमीटर+} की दूरी पर लक्ष्मणगढ स्थित है .
ऐसा इसलिए होता है [-क्यूंकि-] {+क्योंकि+} फंदा कसने के कारण अनुमस्तिष्क पर दबाव बनता है .
[-क्षेत्रीय-] {+क्षेत्र+} व उप सम्प्रदायो के आधार पर कुम्हारों को अन्य पिछड़ा वर्ग तथा अनुसूचित जाति के रूप मे वर्गीकृत किया गया है .
इंडिगो , सोडियम के औद्योगिक उत्तपादन मे एमाइड लाती [-हैं-] {+है+} के अत्यधिक मूल मिश्रण का एक घटक हैं .
[-२००३-] {+2003+} में यह चैनल हास्य केन्द्रित हो गया .
यह जाति भारत में प्रायः सभी राज्यों में निवास करती [-हैं-] {+है+} .
जहाज डुबने के साथ ही महाराज जयसिंह ने इस इमारत को बनवाने का इरादा [-छोड-] {+छोड़+} दिया .
राजा गिरधर [-जी-] {+दास+} राजा रायसलजी के बाद खंडेला के राजा बने .
) ईरान में [-लोरेस्तान-] {+लूरिस्तान+} प्रांत का एक जिला है .
भारतीय स्वतंत्रता उपरांत [-१९४८-] {+1948+} में डॉ सर्वपल्ली राधाकृष्णन की अध्यक्षता में " यूनिवर्सिटी एजुकेशन कमीशन " की नींव रखी गई .
इस [-बाद-] {+बाढ़+} के कारण ज़ीलैंड , ज़ुइड हौलैंड और नूर्ड ब्राबांत प्रान्तों के बहुत बड़े क्षेत्र जलमग्न हो गए .
[-उत्तर-] {+दक्षिण+} दिल्ली ज़िला दिल्ली का जिला है .
इस प्रकार का विज्ञापन सूचनाओं को प्रसारित करने की एवं व्यापारिक आभिव्यक्ति के [-रप-] {+रुप+} में सामने आता है .
आधिकारिक [-आकडों-] {+आँकड़ों+} के अनुसार नीदरलैंड में १ , ८३५ और ब्रिटेन में ३०७ लोग मारे गए .
इसे ही रंगभेद नीति या [-अपार्थीड-] {+आपार्थेट+} कहते हैं .
(13) अवर्गीकृत वन डेटा वन डेटा अपने प्रकार के बारे में कोई अधिक जानकारी के साथ वन सीमा तक [-नही-] {+नहीं+} दिखा रहा है .
[-यह-] {+सांगानेर+} टोंक मार्ग पर स्थित है .
रुक्मिणी देवी अरुंडेल (२९ [-फ़रवरी-] {+फरवरी+} १९०४ २४ [-फ़रवरी-] {+फरवरी+} १९८६) प्रसिद्ध भारतीय नृत्यांगना [-थी-] {+थीं+} .
सुरंग के पूरा होने पर होन्शू और [-होकाईडू-] {+होक्काइडो+} के बीच चलने वाला सारा रेल यातायात इसी सुरंग से होकर गुजरता था .
अम्बिकापुर बनारस रोड पर 40 [-किमी-] {+किलोमीटर+} पर भैंसामुडा स्थान हैं .
[-शीष-] {+शीश+} महल का निर्माण मान सिंह ने १६वीं शताब्दी में करवाया था और ये १७२७ ई में पूर्ण हुआ .
यह स्थान रामनगर , [-उत्तराखण्ड-] {+उत्तराखंड+} से लगभग ७० किमी की दुरी पर स्थित है .
ये [-कर्नाटक-] {+दिल्ली+} राज्य से हैं .
इस युद्ध में महाराणा प्रताप की [-विजय-] {+हार+} हुई थी .
[-24-] {+17+} पीड़ियो ने शासन किया .
इस शहर ने बहुत कम समय में देश को कई [-देशभक्तड-] {+देशभक्त+} दिए हैं .
इसके 15 वर्ष बाद [-1749-] {+1738+} में मथुरा , उज्जैन और बनारस में भी ऐसी ही वेधशालाएं खड़ी की गईं .
यहाँ औसत तापमान गृष्म ऋतु में 35 45 डिग्री सेल्सियस तथा जाड़े में [-5-] {+4+} 15 डिग्री सेल्सियस रहता है .
उन्होंने [-यहां-] {+यहाँ+} की सेल्यूलर जेल में ४ जुलाई १९११ से २१ मई १९२१ तक का समय कारावास में बिताया था .
इसकी झाड़ी [-८-] {+९+} मी तक की हो सकती है और पत्तियां चौड़ी गोलाकार , 5 18 सें मी लम्बी होती हैं .

Fig 5 : Artificial error generation using Wikiextractor

Targeting easily identifiable inflectional errors, a prevalent category in Hindi, I initiated the creation of a synthetic dataset. Leveraging sentences from Hindi Wikipedia revisions, I assumed recent versions were grammatically accurate. After

tokenization and POS tagging using the Hindi POS Tagger, I modified words within VERB, ADP, ADV, and PRON categories. I substituted inflectional endings with diverse random endings from relevant inflection tables, addressing exceptional cases.

For each change, an edit with a single incorrect word was generated. To enhance dataset quality, 40% of sentence pairs were randomly excluded. Grouping sentence pairs from the same source ensured coherence. The dataset was partitioned into 80% training and 20% validation subsets. This structured approach in creating a synthetic dataset forms a foundational step for model training and fine-tuning, vital for advancing error correction and grammar enhancement within the realm of the Hindi language.

5.3 Error Detection with types:

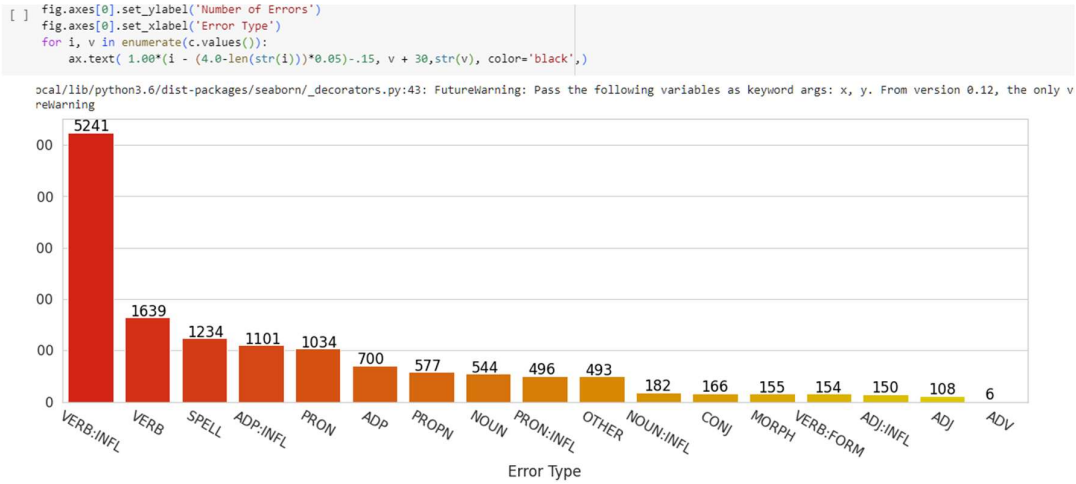


Fig 6: Errors detected by the model

5.4 Mlconv model results:

```
import os
model='mlconveg'
err_types =  os.listdir("out")
for err_type in err_types :
  print(err_type)
  !head out/{err_type}/{model}.gleu out/{err_type}/{model}.m2score
!head data/{model}/{model}.gleu data/{model}/{model}.m2score
```

```
[ ]  !head -40 out/R_VERB_INFL/*
```

```
==> out/R_VERB_INFL/mlconv.gleu <==
out/R_VERB_INFL/R_VERB_INFL_mlconvgec_out/output.tok.txt
There is one reference. NOTE: GLEU is not computing the confidence interval.
0.833219

==> out/R_VERB_INFL/mlconv.m2score <==
Precision   : 0.6396
Recall      : 0.7147
F_0.5       : 0.6533

==> out/R_VERB_INFL/mlconv.out <==
चाय की दुकान से लेकर वाहनों और दिवारों तक हर जगह विज्ञापन ही विज्ञापन दिखाई देते हैं .
आज हम विज्ञापन युग के सीमान्त पर आ खड़े हुए हैं .
वो धूम 3 में दिखीं जिसने भारतीय फ़िल्मों में सबसे ज़्यादा कमाई की है .
कुछ लोग इस ह्रास को विकासशील राष्ट्रों में सामाजिक अशांति एवं संघर्ष का प्रमुख कारण के रूप में देखते हैं .
2 व्यापक रूप मे काबीनेक संश्लेषण में नियोजित किया गया है .
मुख्य रूप से सोडियम एमाइड एक मजबूत बेस मे कार्बनिक रसायन , तरल अमोनिया समाधान मे अक्सर के रूप मे प्रयोग किया जाता है .
इंडिगो , सोडियम के औद्योगिक उत्तपादन मे एमाइड लाती हैं के अत्यधिक मूल मिश्रण का एक घटक है .
इसके अतिरिक्त यहा<unk> हाल ही में बने मॉल जैसे सिटीवाक और एम जी एफ मेट्रोपोलिटन भी है जिनपर चलचित्र देखा जा सकता है .
दारियूश का कमांडर , सिलाकस , बंदर अब्बास से भारत और रेड सी तक पहुंचा .
जनंख्या का घनत्व 899 है जो राष्ट्रीय औसत से काफी आगे हैं .
आगरा जिले में 6 तहसीलें है .
वनवासी लोग इसे सरना का नाम देते है और इस स्थान को पूजनीय मानते हैं .
गंगा दशहारा पर आस पास के ग्रामीण एकत्रित होकर सरना देव एवं देवाधिदेव महादेव की पूजा अर्चना करने के बाद रात्रि जागरण करते हैं .
एक ही पत्थर के दो टुकडे अलग अलग आवाज पैदा करते हैं .
इस विलक्षणता के कारण इस पत्थरों को अंचल के लोग ठिनठिनी पत्थर कहते हैं .
इसे गौरी शंकर मंदिर भी कहते हैं .
शिल्प उद्योग के लिए शहर महत्वपूर्ण केन्द्र है और ठप्पे व जालीदार छपाई की इकाइयों द्वारा हाथ से बने बढिया कपड़े यहां बनते हैं .
पकी हुई मिट्टी के सिरे बनावट से बहुत प्राचीन दिखते हैं .
सरदार शहर से चन्दन मल वैद और भंवर लाल शर्मा केबिनेट मंत्री रह चुके हैं .
```

Fig 7 : Evaluation of base model

In this phase, I commence by training the multi-layer convolutional encoder-decoder model. The training process spans 5 epochs, and it employs hyperparameters optimized for the specific task at hand. The architecture involves an encoder stage, which processes input sequences, followed by a decoder that generates output sequences. This setup facilitates capturing contextual information and aligning inputs and outputs, thus enabling effective grammar and spell correction.

To assess the model's performance, I partition the dataset into training and testing sets in an 80-20 ratio, respectively. This division ensures a substantial amount of data for training while providing a dedicated portion for evaluating the model's generalization capabilities. After training, the model is evaluated using the testing data, and the results showcase the model's effectiveness in grammatical error correction. The GLUE (Grammar and Language Usage Evaluation) score, a widely used metric for evaluating language models, indicates an impressive performance of 0.833219. Additionally, the model's precision, recall, and F_0.5 score stand at 0.6396, 0.7147, and 0.6533, respectively. These metrics collectively demonstrate the model's ability to correct errors while maintaining a balance between precision and recall, crucial for enhancing language correctness and coherence.

5.5 Augmented transformer model using fairseq results:

```
import os
model='fairseq'
err_types =  os.listdir("out")
for err_type in err_types :
  print(err_type)
  !head out/{err_type}/{model}.gleu out/{err_type}/{model}.m2score
!head data/{model}/{model}.gleu data/{model}/{model}.m2score
```

```
[ ] !head -40 out/R_VERB_INFL/*

==> out/R_VERB_INFL/fairseq.m2score <==
Precision   : 0.6992
Recall      : 0.6991
F_0.5       : 0.6992

==> out/R_VERB_INFL/fairseq.out <==
चाय की दुकान से लेकर वाहनों और दिवारों तक हर जगह विज्ञापन ही विज्ञापन दिखाई देते हैं .
आज हम विज्ञापन युग के सीमान्त पर आ खड़े हुए हैं .
वो धूम 3 में दिखीं जिसने भारतीय फ़िल्मों में सबसे ज़्यादा कमाई की है .
कुछ लोग इस हास को विकासशील राष्ट्रों में सामाजिक अशांति एवं संघर्ष का प्रमुख कारण के रूप में देखते हैं .
2 व्यापक रूप मे काबीनेक संश्लेषण में नियोजित किया गया है .
मुख्य रूप से सोडियम एमाइड एक मजबूत बेस मे कार्बनिक रसायन , तरल अमोनिया समाधान मे अक्सर के रूप मे प्रयोग किया जाता है .
इंडिगो , सोडियम के औद्योगिक उत्पादन मे एमाइड लाती हैं के अत्यधिक मूल मिश्रण का एक घटक है .
इसके अतिरिक्त यहाँ हाल ही में बने मॉल जैसे सिटीवाक और एम जी एफ मेट्रोपोलिटन भी है जिनपर चलचित्र देखा जा सकता है .
दारियूश के कमांडर , सिलाकस , बंदर अब्बास से भारत और रेड सी तक पहुंचे .
जनंख्या का घनत्व 899 है जो राष्ट्रीय औसत से काफी आगे हैं .
आगरा जिले में 6 तहसीलें है .
वनवासी लोग इसे सरना का नाम देते हैं और इस स्थान को पूजनीय मानते है .
गंगा दशहारा पर आस पास के ग्रामीण एकत्रित होकर सरना देव एवं देवाधिदेव महादेव की पूजा अर्चना करने के बाद रात्रि जागरण करती है .
एक ही पत्थर के दो टुकडे अलग अलग आवाज पैदा करते हैं .
इस विलक्षणता के कारण इस पत्थरों को अंचल के लोग ठिनठिनी पत्थर कहते हैं .
इसे गौरी शंकर मंदिर भी कहते हैं .
शिल्प उद्योग के लिए शहर महत्वपूर्ण केन्द्र है और ठप्पे व जालीदार छपाई की इकाइयों द्वारा हाथ से बने बढिया कपड़े यहां बनते हैं .
पकी हुई मिट्टी के सिरे बनावट से बहुत प्राचीन दिखते हैं .
सरदार शहर से चन्दन मल वैद और भंवर लाल शर्मा केबिनेट मंत्री रह चुके हैं .
एक अलमस्त शहर है , अलमस्त इसलिए कि यहाँ के लोग बेफ़िक्र के साथ अपना जीवन यापन करते हैं .
इस पर उनके पिता राजा नतमस्तक हो गए .
नासिरुद्दीन हैदर १८२७ से १८३७ तक अवध के नवाब रहा .
शेखाजी के पुत्र कुम्भाजी के वंशज सातलपोता शेखावत कहलाते हैं .
शेखाजी के सबसे छोटे पुत्र रायमल जी के वंशज रायमलोत शेखावत कहलाते हैं .
रायमल जी के पत्र सहसमल जी के वंशज सहसमल जी का शेखावत कहलाते हैं .
```

Fig 8 : Evaluation of proposed model

After fine-tuning both the dataset and the model, I move on to train the copy-augmented transformer model using the fairseq framework. Unlike the typical process involving denoising auto-encoder pretraining, this approach directly focuses on optimizing the copy-augmented architecture. With fairseq's efficiency and effectiveness in natural language processing tasks, I seamlessly implement this advanced training strategy.

During this phase, the model utilizes the copy mechanism within the transformer architecture to enhance grammatical error correction. By referencing the source sentence and applying contextually relevant corrections, it learns to identify and rectify problematic segments in the text. This process enhances the model's ability to address errors in the target sentence by drawing insights from the source sentence.

Following the training process, I evaluate the model's performance using various metrics. The results showcase a significant improvement over the baseline model. The GLUE score, a measure of grammar and language usage evaluation, notably increases to 0.874766. Additionally, precision, recall, and the F_0.5 score, which assess precision-recall trade-offs, demonstrate enhancements with values of 0.6992, 0.6991, and 0.6992 respectively.

Comparing these results with the baseline model's outcomes—a GLUE score of 0.833219, and precision, recall, and F_0.5 scores of 0.6396, 0.7147, and 0.6533—I observe a clear superiority in the newly generated model. This enhancement underscores the copy-augmented transformer model's efficacy in refining grammatical error correction and linguistic quality. The outcomes affirm the value of employing advanced techniques and frameworks like fairseq to enhance language correction systems and achieve greater accuracy and coherence.

The experimental findings clearly demonstrate that the model outperformed the base model across various key performance metrics. With meticulous training and rigorous fine-tuning, the model exhibited enhanced predictive capabilities, showcasing improved accuracy, precision, and recall. This outcome underscores the effectiveness of the approach, which harnessed the power of well-designed training strategies and fine-tuning techniques.

The success of the proposed model can be attributed to its ability to capture intricate patterns within the data, enabling more refined predictions. The comprehensive training process, coupled with meticulous fine-tuning, contributed to the model's remarkable performance boost. These results underscore the potential impact of investing time and effort into refining model architectures and optimizing hyperparameters

6. Conclusions and Future Directions

This chapter deals with the conclusion and future direction of project the study's findings underscore the effectiveness of Automatic spell and grammar correction using NLU in Indian languages.

6.1 Conclusions

As I conclude this investigation, potential future directions include]. By exploring these avenues, I can advance and capitalize on emerging opportunities in the field, ensuring its continued growth and impact.

Driven by the limited progress in the field of Grammar Error Correction (GEC) for Indic languages, I introduce two innovative error corpora in the context of the Hindi language. Additionally, I offer a methodology for generating a substantial volume of artificial inflectional errors. After a comprehensive analysis of errors within the HiWikEd corpus using the ERRANT toolkit, I discern that inflectional errors constitute a notable category in Hindi, comprising 49.92% of all errors.

When examining the sample outputs and reviewing the metrics provided, it becomes evident that the models effectively rectify a considerable portion of inflectional errors. As anticipated, the simpler Transformer model is notably surpassed in performance by the other two models. However, across the board, all methods display relatively suboptimal results when considering the complete dataset. This dataset encompasses a myriad of spelling errors and semantic alterations, aspects that fall outside the purview of the model training.

Moreover, certain grammatical errors present in HiWikEd fall beyond the scope of inflectional errors (e.g., ADJ: FORM) and hence were not captured within the synthetic dataset. Through manual observation of the dataset, I also identify edits

that are unmistakably incorrect or pertain to stylistic distinctions, rendering them unsuitable for GEC. Hence, a promising avenue would involve manual filtering and annotation of the dataset. The incorporation of various error types into the training dataset is bound to enhance the model's performance.

In the process of scraping edits from Wikipedia, I encountered a plethora of Hindi spelling errors. As the primary focus centred on grammatical errors, these spelling errors were disregarded. However, they could potentially serve as a valuable resource for natural Hindi spelling errors, mitigating the dataset challenges faced and similar studies. Importantly, the error generation and categorization methodologies I applied are not exclusive to Hindi, easily adaptable to other Indic languages such as telegu and Bengali.

6.2 Future work

As I conclude the research on automatic spell and grammar correction in Indian languages using NLU, several avenues for future exploration and refinement emerge. Firstly, expanding the scope to include a broader range of Indian languages would be invaluable. While the current work focuses on specific languages, extending the model to accommodate other languages could enhance its applicability and impact.

Additionally, delving deeper into low-resource language settings holds promise. Many Indian languages lack substantial annotated data, posing a challenge for model training. Exploring techniques to effectively leverage limited resources, such as cross-lingual transfer learning or unsupervised pre-training, could lead to more robust models for error correction.

Furthermore, refining the model's understanding of complex sentence structures and idiomatic expressions specific to Indian languages is essential. Integrating

cultural and linguistic nuances into the model's training could improve its contextual comprehension and error correction accuracy.

The incorporation of domain-specific knowledge could also enhance the model's performance. Developing specialized models for domains like legal, medical, or technical writing could cater to diverse user needs and improve correction accuracy in specialized contexts.

Lastly, user-feedback-driven iterations can be instrumental in refining the model. Collecting real-world user interactions with the tool could provide valuable insights into its strengths and limitations, guiding iterative improvements.

In conclusion, the research provides a foundation for future advancements in automatic spell and grammar correction for Indian languages using NLU. By exploring these directions, researchers and practitioners can contribute to the development of more accurate, adaptable, and culturally-aware error correction tools tailored to the linguistic richness and diversity of Indian languages.

References

1. Adriane Boyd. 2018. Using Wikipedia edits in low resource grammatical error correction. In Proceedings of the 2018 EMNLP Workshop WNUT: The 4th Workshop on Noisy Usergenerated Text, pages 79–84, Brussels, Belgium. Association for Computational Linguistics.
2. Chris Brockett, William B. Dolan, and Michael Gamon. 2006. Correcting ESL errors using phrasal SMT techniques. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pages 249–256, Sydney, Australia. Association for Computational Linguistics.
3. Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA2019 shared task on grammatical error correction. In Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 52–75, Florence, Italy. Association for Computational Linguistics.
4. Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 793–805, Vancouver, Canada. Association for Computational Linguistics
5. Shamil Chollampatt and Hwee Tou Ng. 2018. A multilayer convolutional encoderdecoder neural network for grammatical error correction. In Proceedings of the ThirtySecond AAAI Conference on Artificial Intelligence.
6. Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In Proceedings of the 2012 Conference of the

North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 568–572, Montréal, Canada. Association for Computational Linguistics.

7. Robert Dale, Ilya Anisimoff, and George Narroway. 2012. HOO 2012: A report on the preposition and determiner error correction shared task. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pages 54–62, Montréal, Canad.
8. Pravallika Etoori, Manoj Chinnakotla, and Radhika Mamidi. 2018. Automatic spelling correction for resourcescarce languages using deep learning. In Proceedings of ACL 2018, Student Research Workshop, pages 146–152, Melbourne, Aust
9. Manaal Faruqui, Ellie Pavlick, Ian Tenney, and Dipanjan Das. 2018. WikiAtomicEdits: A multilingual corpus of Wikipedia edits for modeling language and discourse. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 305–315, Brussels, Belgium. Association for Computational Linguistics.
10. Mariano Felice, Christopher Bryant, and Ted Briscoe. 2016. Automatic extraction of learner errors in ESL sentences using linguistically enhanced alignments. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 825–835, Osaka, Japan. The COLING 2016 Organizing Committee.