DETAILED ANALYSIS ON LANGUAGE MODEL PERFORMANCE

A PROJECT REPORT

Submitted by

BL.EN.U4CSE19045 Gayatri Sanjana Sannala

BL.EN.U4CSE19115 Rohit R

BL.EN.U4CSE19116 S. A. Gandheesh

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF COMPUTING, BANGALORE

AMRITA VISHWA VIDYAPEETHAM

BANGALORE 560 035

May – 2023

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING, BANGALORE, 560035

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **"**Detailed Analysis on Language Model Performance**"** submitted by

BL.EN.U4CSE19045	Gayatri Sanjana Sannala
BL.EN.U4CSE19115	Rohit R
BL.EN.U4CSE19116	S. A. Gandheesh

in partial fulfillment of the requirements as part of **Bachelor of Technology** in "**COMPUTER SCIENCE AND ENGINEERING"** is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Computing, Bangalore.

11/05/2023

Dr. Peeta Basa Pati	Dr. Gopalakrishnan E. A.,
Professor	Professor and Chairperson,
Dept. of CSE	Dept. of CSE

This project report was evaluated by us on 3rd May, 2023.

J. Amudha	Niranjan D K
EXAMINER1	EXAMINER2
Dr. Amudha J. Professor	Mr. Niranjan D K Lecturer

ACKNOWLEDGEMENTS

The satisfaction that accompanies successful completion of any task would be incomplete without mention of people who made it possible, and whose constant encouragement and guidance have been source of inspiration throughout the course of this project work.

We offer our sincere pranams at the lotus feet of **“AMMA”, MATA AMRITANANDAMAYI DEVI** who showered her blessing upon us throughout the course of this project work.

We owe our gratitude to **Prof. Manoj P.**, Director, Amrita School of Engineering, Bangalore.

We thank **Dr. Sriram Devanathan,** Principal and **Dr. Gopalakrishnan E. A.,** Chairperson-CSE, Amrita School of Engineering, Bangalore for their support and inspiration.

It is a great pleasure to express our gratitude and indebtedness to our project guide **Dr. Peeta Basa Pati, Professor**, Department of Computer Science and Engineering, Amrita School of Computing, Bangalore for her/his valuable guidance, encouragement, moral support, and affection throughout the project work.

We would like to thank express our gratitude to project panel members for their suggestions, encouragement, and moral support during the process of project work and all faculty members for their academic support. Finally, we are forever grateful to our parents, who have loved, supported and encouraged us in all our endeavors.

ABSTRACT

The language models BART and MarianMT have been trained and tested, and their evaluation metric scores are presented in various papers. Only the performance of the models was analyzed based on correction without any specific stress test. Therefore, this project aims to stress test the model on various different percentages of errors. This study will help in identifying the breaking point of these models and employing them accordingly.

This work will be extended to various other models to learn their breaking point to gain an understanding about the strong points of the models. Based on these analyses, the appropriate models can be utilized on various erroneous percentage inputs and categories of errors accordingly. By doing so, already existing applications can be made more robust.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

S.No.	Acronym	Full Form
1	BART	Bidirectional Auto-Regressive Transformers
2	MarianMT	Marian Machine Translation
3	CER	Character Error Rate
4	WER	Word Error Rate
5	OCR	Optical Character Recognition
6	RNNLM	Recurrent Neural Network Language Model
7	LM	Language Models
8	LSTM	Long Short Term Memory
9	GPT	Generative Pre-trained Transformation
10	GEC	Grammatical Error Correction
11	BERT	Bidirectional Encoder Representations from Transformers
12	CSC	Chinese Spelling Correction
13	TL	Transfer Learning

CHAPTER - 1
INTRODUCTION

Text is a natural representation of all the existing languages in the world. Texts help one express and communicate with others. It is due to the existence of text that the existence and explanation of evolution is known to humans. Handwritten texts have been part of the history for ages, while digital texts have evolved to keep up with the rapidly growing technology in day to day lives. It is due to texts that one can extend from their knowledge and memory beyond their body into the environment around [1]. Text is available in various forms, from handwritten manuscripts to digitally written blogs, from stone carvings to printed posters. Texts can be utilized for personal reasons such as diary entry, blog, etc., as well as for professional purposes like advertising, surveying, etc. Right from the newspaper one reads in the morning to the social media scrolling before going to bed, people are surrounded by text.

It is human nature to categorize any kind of data they receive. As there is so much text available around, it is obvious that humans tend to inspect and review the text they require. Thus, the origin of text analysis. It is the process of scanning the textual data in order to derive some meaning and store information.

Most Businesses rely on text analysis to extract valuable insights from various raw sources. The feedback received from these sources such as emails, chat messages, social media posts, comments & statements and survey responses help them in their decision-making strategies. Text capturing can be complex and tend to introduce new errors based on its source and capture technology.

Besides digital native documents, text may be obtained from other forms of media such as images, video, speech or voice. OCR extracts text present in images, thereby enabling editing and reviewing of the content [2]. The captured textual information may contain errors due to a variety of reasons. Some of these errors are pre-existing in the input images due to incorrect typing or even inaccurate language knowledge. Moreover, the models used to recognize the text in images may introduce some errors, since all OCR engines are inherently error-prone. These recognition errors may occur due to noisy and unclear images, poor handwriting, formatting and spacing issues. Speech recognition [3] is a speech-to-text technology that recognizes spoken words and into its text equivalent. Auto-transcription of speech may introduce errors while phoneme to grapheme conversion is carried out. These errors may be attributed to factors such as substitution of a word with a different, yet similar sounding word, deletion of incoherent words and insertion of contextually irrelevant words. Technologies such as web scraping [4], email-readers and file handlers may also introduce errors while capturing text due to format incompatibility.

Thus, there are mainly two distinct error types available in text sentences, namely, spelling or typographic error and grammatical error. Both these kinds of error may be attributed to errors in symbol recognition, mis-typing and incorrect language knowledge. Existence of these errors may lead to undesirable consequences, especially in fields such as courtrooms & hospitals. For example, the sentences "The accused fled from the crime" and "The accused bled from the crime" can lead to very different outcomes for the case based on just one spelling mistake. The sentence "He are healthy" is grammatically incorrect, yet highly possible to be created by someone who lacks proper knowledge of English grammar.

There are several methods to eradicate these errors. While one can utilize manual labor to individually pick out the errors, it can be time consuming and can add some human errors. Rule based error correction [5] is also a solution to eliminate error. In such a system, first a large number of documents are studied to identify the common error patterns based one which the rules are defined. These defined rules are, subsequently, employed to bring corrections to errors in future documents. Efficiency of such error correction is based on the quality of defined rules as well as ordering and properties of the rules. Though it is useful in reducing the errors in text, it is sensitive to newer language structures. Rule Based error correction is often time consuming, without any learning capacity and is very unidimensional, thereby, making it inefficient. To overcome these shortcomings, advanced deep neural network based Natural Language Processing (NLP) models are used for error correction. These NLP models use Artificial Intelligence (AI) and enable machines to read, understand & analyze the meaning as well as context of the text sentences. Since text is a sequential data consisting of a sequence of words, Recurrent Neural Network (RNN) architectures are utilized to analyze and capture the information storing them as language models. Long Short-Term Memory (LSTM) networks are advanced RNN architectures which allow information to persist for relatively longer duration in the model's network. The encoder-decoder architecture of the LSTM carries the capability to learn the context of a sentence and store information in the model, thereby making it a perfect tool for error correction [6]. This paper speaks about two such deep learning NLP models, Bidirectional Auto-Regressive Transformers (BART) [7] and MarianMT [8].

BART and MarianMT models have an encoder-decoder architecture, they work using sequence-to-sequence modeling of language data, and are memory-based. During training, these models pick up on the traits and rules of the grammar. The noisy text is fed as input to these language models, which produce accurate sentences as output. This helps to improve the accuracy of generated text. Both these models accomplish the stated task by comprehending the grammar and context of used words in a sentence. Token masking, token detection, text infill & sentence permutation are tasks that the BART and MarianMT decoders are trained to perform.

BART is an advanced language model and has been reported for its efficacy for various language processing tasks. MarianMT is yet to be thoroughly explored in these aspects. For the purpose of this work, both these models are employed to correct various errors present in the input sentence. Many researchers have already employed such language models for accuracy enhancements and reported success with the approach. However, most of these works deal with study of the aggregated quantitative impact, namely, accuracy improvement, error reduction etc. It's equally important to find the pattern of improvement brought in by these models. The outcome of such a study shall help in identification of error scenarios and find the right model to be employed.

In this work, an error category definition is developed & utilized to categorize all the input sentences in the dataset as well as the corresponding predicted outputs. While it's ideal to assume that all input sentences shall be predicted as fully corrected, in practice predicted sentences may also have errors. Thus, it's important to study the shift pattern of the error categories which is the main focus of this work. Observing and analyzing the shift in error categories provides insights into the model's behavior as well as capacity for capturing the context and meaning of the sentences. Such analysis also helps to develop effective strategies to engage such models for appropriate correction based on the types of error available in input sentences. This work aligns with United Nations' Sustainable Development Goals of "Quality Education (SDG-4)" and "Decent Work and Economic Growth (SDG-8)".

CHAPTER - 2

LITERATURE SURVEY

Language models have been employed in numerous research studies on a variety of documents to get a wide range of results. Following are some of the noteworthy research contributions.

Errors introduced by non-native English speakers are a big challenge in Grammatical error correction as there is change in the meaning of the sentences. Liang et al. [9] proposed a solution to overcome this problem. They first defined the categories of noise in an English sentence and then injected targeted noise into sentences to build training sets. Subsequently, they fine-tuned BERT with the training set which outperformed various state-of-the-art language models such as LSTM and CNN.

Saluja et al. [10] developed an LSTM-based model with a fixed latency that can learn, detect, and correct OCR errors. Three studies were carried out: error detection (using various evaluation), error correction, and suggestion generation. The results showed that their LTSM model provided the best error correction for Hindi and Malayalam compared to the prior reported models. Introduction of the fixed delay to LSTM is a novelty, different from the standard LSTM models, that enables learning from the subsequent sequence of characters.

Tan et al.'s [11] study focused mostly on BERT for Mandarin spelling error correction. The BERT model, which was trained on a Chinese dataset, is used in the paper to fix incorrect strings. The detection network and the correction network were kept as two distinct parts of the model during design. The task of the detection network is to identify incorrect text (such as misspelled words) while the correction network rectifies the incorrect text. The model's precision, recall, and F1 scores were recorded and compared with those of Kenlm, RNNLM, and BERT-Fintune. An average improvement of 13% for each of these evaluation metrics was observed.

The authors Alikaniotis and Raheja [12] propose that state-of-the-art language models can be used to achieve competitive performance on the task of grammatical error correction (GEC) without the need for annotated training data. The authors reported using pre-trained Transformer language models like BERT, GPT, and GPT-2 to perform GEC & demonstrated that these models create tough baselines to beat. Their work proved that transformer based language-models are effective and robust for the task of GEC, even in absence of annotated training data.

Bryant and Briscoe [13] discuss the use of language models (LM) in GEC. The work proposes a simple 5-step approach that relies on very little annotated data and can be used for any language. The approach involves calculating the probability of input sentences, building a confusion set for each token, rescoring the sentence, applying the best correction, and iterating. The work is concluded by highlighting the potential of LM-based approaches for GEC and their usefulness as a baseline for future research.

Existing GEC systems suffer from not having enough labeled training data to achieve high accuracy. To overcome this problem, Zhang et al. [14] proposed a copy-augmented architecture for the GEC task by copying the unchanged words from the source sentence to the target sentence. The copy-augmented architecture was pre-trained with unlabeled One Billion Benchmark dataset. This is followed by comparisons between the fully transferred learnt model and a pretrained model. A copying mechanism was applied on the GEC system, which enables the model to copy tokens from the source sentence. The model was evaluated against CoNLL-2013, CoNLL-2014 and JFLEG dataset. The copy-augmented model had a precision of 68.48%, recall of 33.10%, f1-score of 56.42% with respect to CoNLL-2014 and GLEU score of 59.48 with respect to JFLEG. The authors reported an increase in the evaluation metrics when the model is combined with denoising auto-encoders. A precision of 71.57%, recall of 38.65%, f1-score of 61.15% with respect to CoNLL-2014 and GLEU score of 61.00 with respect to JFLEG was achieved with such a combination.

Chinese Spelling Check (CSC) is a challenging task due to the complex characteristics of Chinese characters. The most common errors present in Chinese language are phonological or visual errors. To overcome these problems, Huang et al. [15] proposed a novel end-to-end trainable model called PHMOSpell, which promotes the performance of CSC with multi-modal information. Pinyin and glyph graphical representations are derived which are then integrated into a pre-trained language model by a well-designed adaptive gating mechanism. The reported model significantly outperformed all previous state-of-the-art models on precision, recall and f1 score metrics.

Existing state-of-the-art methods either only use a pre-trained language model or incorporate phonological information as external knowledge. To overcome this drawback, Zhang et al. [16] proposed an end-to-end Chinese spelling correction (CSC) model that integrates phonetic features. Initially, the words were replaced with phonetic features and their sound-alike words. Then these words were jointly trained for error correction and detection. The transformer model was trained on SIGHANI15 dataset and significantly outperformed previous state-of-the-art methods with a precision of 77.5%, recall of 83.1% and f1-score of 80.2%.

Xu1 et al. [17] proposed a Chinese spell checker model called ReaLiSe by directly leveraging the multimodal information of the Chinese characters. The ReaLiSe model tackles the CSC task by first capturing the semantic, phonetic, & graphic information of the input characters and then selectively mixing the information in these modalities to predict the correct output. The performance of ReaLiSe model was reported with an accuracy of 84.7%, precision of 77.3%, recall of 81.3% and f1-score of 79.3%.

Based on the survey of the related works, it is inferred that models such as BERT & BART are used not only for various kinds of text processing works but has also been studied for their effectiveness in correction of errors in text. MarianMT is also an enhanced version of BERT and is demonstrated to be effective for spelling error correction. Therefore, BART and MarianMT are chosen for this work. Additionally, while most reported works have studied the effectiveness of various models for error correction of one type of error (either grammatical or spelling), this work tries to simultaneously handle both these error types. This work spotlights the error shifts occurring with the model predictions to understands the models' capabilities under differing scenarios of anomalies. This helps identify the models' strengths and weaknesses.

CHAPTER - 3
REQUIREMENTS

Software and Hardware Requirements:

- **Front End S/W**

  HTML5 , CSS3, React v18, Flutter 3, Javascript

- **Back End S/W**

  Node JS with Express, Dart, MongoDB, ReST API, Python 3.X

- **H/W Requirements**

  GPU (GeForce GTX Titan Black)

CHAPTER - 4

SYSTEM DESIGN

The following flowchart represents the work done earlier in a report given by Rohit et al. [18]. Fig. 4.1 explains about how the application works. In that report a system was built with Deep learning based language models for handwritten recognized text accuracy enhancement.

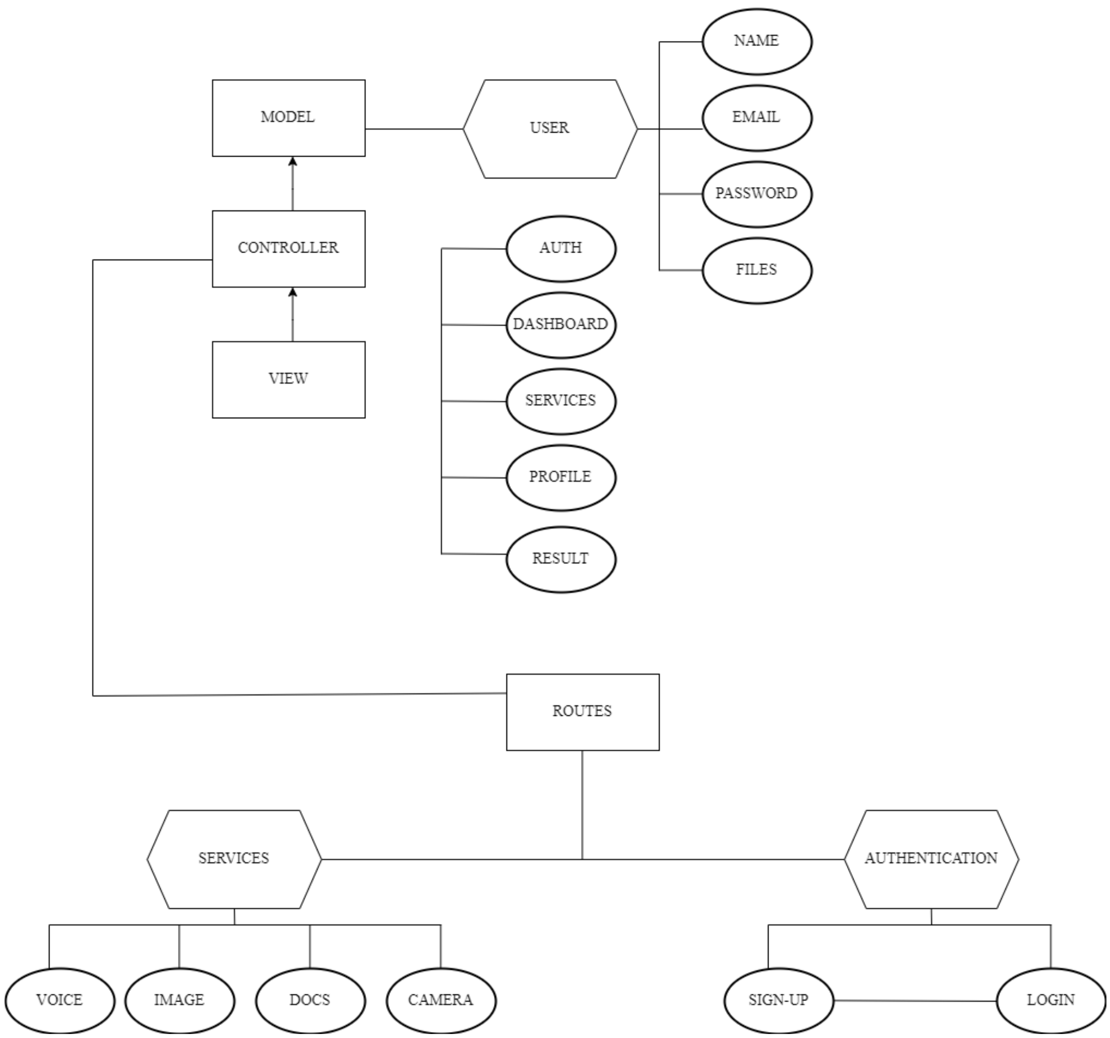


Fig. 4.1 Flowchart for Phase 1

The work carried out in earlier report is as follows:

4.1 Full Stack Development of a working Application

A full stack mobile application and web application was developed to complement the output generated by the models. This user friendly interface was built for easy text recognition using OCR and to view the effective error correction of the proposed model. A patent submission for the same was carried out.

4.2 Model Building

BART and MarianMT are deep learning language models that use sequence to sequence encoder decoder architecture. These models were built by fine tuning the already existing pre trained language models to correct both grammatical and spelling errors.

4.3 Error Category Definition

Hossain et al. [19] studied the different errors in textual documents and presented the list of eight error types. These error types are: Typographic error, Cognitive error, Visual error, Run-on error, Split-word error, Non-word error, Real-word error. Subsequently, Mounika et al. [20] studied the error types present in automatically speech recognized (ASR) text in their work. Their work provided a list of errors observed in sentences. Both the works of Hossain et al. & Mounika et al. dealt with errors at word level. In this study, these categories of errors are abstracted to sentence level and four distinct categories of errors are proposed. The detailed definitions of these four proposed error categories, namely, Cat A, Cat B, Cat C and Cat D, are provided below. These error categories, each with a short description and example sentences, are tabulated in Table 4.1.

4.3.1 Cat A:

This is a no-error category. Here, the input sentence matches perfectly with the target sentence.

4.3.2 Cat B:

This category deals with sentences containing errors such as grammatical, word omission, capitalization. The input sentences do not match the target perfectly but all constituent words of the input sentence are valid words. Here, the erroneous words are of real-word error types as described by Hossain et al [19]. Additionally, new words may be added or some words may be missing in the sentence. The various caused which leads to formation of this error category are listed below.

- Some words of the input sentence do not match to words in target; all such words in the input sentence are valid dictionary words.

- New words (valid dictionary words) are added or some words are missing in the input sentence.

- Case change effect in the input sentence. Examples are all characters in upper or lower case, case changes, initial upper case while not needed, initial uppercase missing etc.

- Change in position of words leading to improper & grammatically inaccurate sentences.

- Changing the sentence formation from direct to indirect speech or vice-versa; change of speech in sentences from first person to third person.

4.3.3 Cat C:

The constituent words present in sentences contain spelling errors or typographic errors. The input sentences mostly contain non-word errors. This leads to the formation of words which are neither proper nouns nor are found in the dictionary. This category may contain words from languages other than English as well.

4.3.4 Cat D:

This category or error consists of input sentences containing both Cat B & Cat C type errors. Thus, the sentences contain both non-word and real-word errors. For a language model, this is the most complex type of error to deal with due to the presence of both Cat B & C type errors. Such errors are very confusing to humans as well.

Table 4.1: Different Categories of Errors.

Error Category	**Description**	**Example Sentence**	**Example Target**
Cat A	No errors detected in the text	These are cars.	These are cars.
Cat B	Grammatical errors detected in the text.	These are car.	These are cars.
Cat C	Spelling errors detected in the text.	These are rars.	These are cars.
Cat D	Sentence formation errors are introduced.	Thess are car.	These are cars.

In this report, the focus is drawn towards addressing important research questions to ultimately find the model's capabilities and limitations. As a pre-processing step to answer the research questions, some computations and analysis were done. There are various reasons for a model to make changes to the input text in order to correct it, which in turn causes a category shift. It is essential to learn all these reasons, so a better understanding of the model's ability can be drawn.

Additionally, two input-target sets were created using the existing dataset where each set contained different percent error in input was present. The two sets with synthetic and natural errors were created to observe the model's behavior towards both the types of error, as explained in Fig. 4.2. Although synthetic errors are easier to generate and test the model quickly, natural errors are closer to real life scenarios and it is important for the model to learn and correct these errors. Hence, models were tested for both the error types.

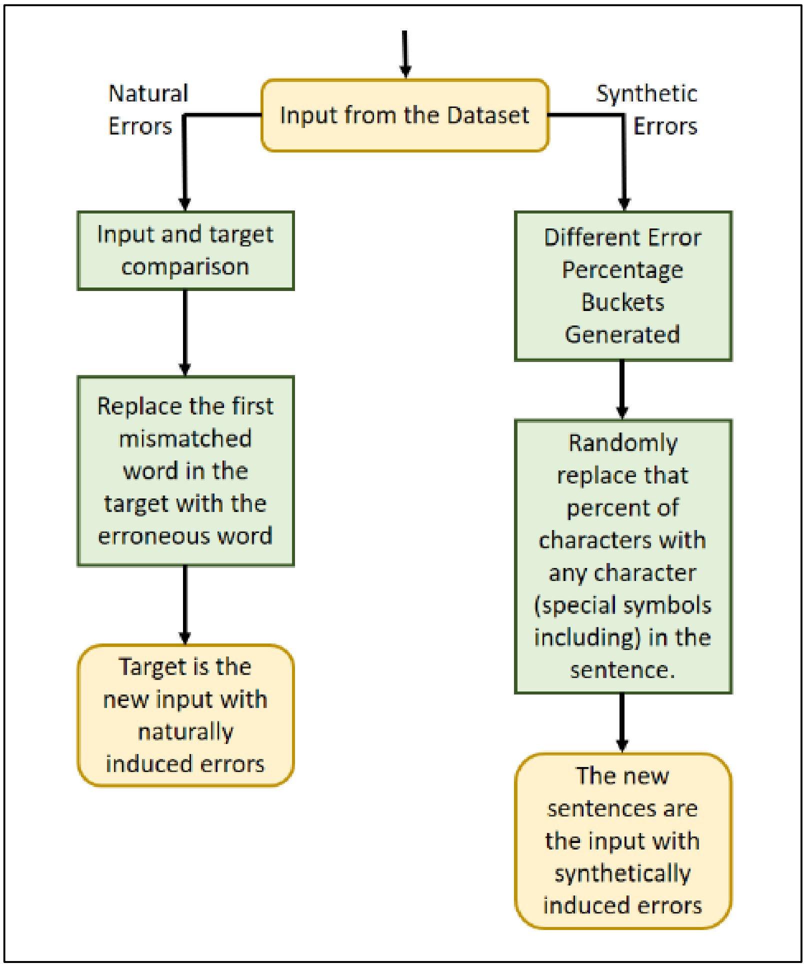


Fig. 4.2 Flowchart on generating the Error Set

4.3 Research Questions

The Research questions to be addressed are as follows:

i. Do the models distinguish between natural error and synthetic error?

ii. Do the BART and MarianMT models behave differently for Synthetic errors?

iii. Do the models behave differently for CER and WER metric?

iv. What is the Break Even Point of BART model?

v. Does Transfer Learning models provide better prediction to the input sentences than pre-trained models?

CHAPTER – 5
SYSTEM IMPLEMENTATION

5.1 Dataset

C4 dataset is an Open-Source dataset obtained with Common Crawl web scrape [21]. This dataset contains millions of collected sentences along with their target sentences. The collected statements, referred to as input sentences in this work, belong to one of the categories defined in Section 4.3 of this report. The input sentences are manually typed chat messages sent by users across the internet. These target sentences are manually inspected corrected input sentences by human experts. These target sentences act as the ground truth for this work. Due to constraints of available computing infrastructure, only a million sentences from this dataset are chosen for this work.

These one million sentences were analyzed for the error category distribution. Fig. 5.1 represents the distribution pattern of the dataset chosen for this work. It may be observed that most of these input sentences are error free and belong to Cat A.

These one million sentences were split into train, validation, and test sets with 50:20:30 ratio. 500,000 sentences were used for training of the models while 200,000 sentences were used as validation sets (refer Table 5.1). A closer inspection of the test set sentences revealed that some of the target sentences contained errors (improper job in correcting the input sentences by human experts). Since the existence of such error affects the evaluation of the models, an exercise to inspect and correct the targets was undertaken. Such an exercise demands huge investment of time. Hence, a limited set of 25,738 sentences were inspected and corrected in this work. This set, called as 'UpdTest' is used for all testing and result reporting in this work.

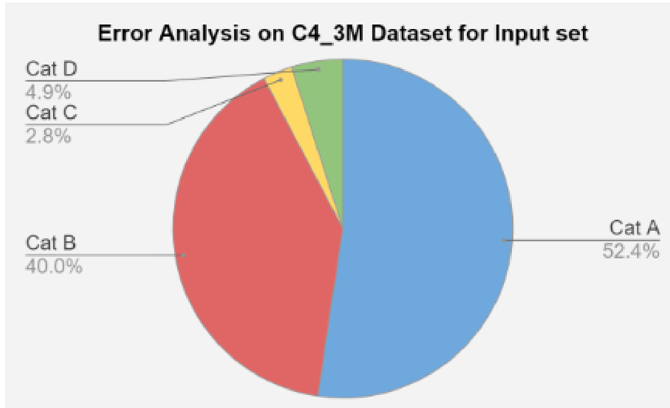


Fig. 5.1: Error category distribution of the input sentences in the dataset used for this work.

To understand the impact of target sentence inaccuracy on the model training, a study was conducted on the 25,738 samples in the UpdTest set. The study involved identification of sentences which passed as error free during the quality inspection process. It was observed that ~90% target sentences were error free. Since, most of the target sentences are error-free, it was assumed that only a negligible influence shall be exerted by the rest 10% of the erroneous sentences during the model training. So, a decision to continue with the training set (500,000 sentences) and validation set (200,000 sentences) was taken.

Table 5.1: Count of sentence records in Train, Test & Validation sets.

Parameter	**Value**
Train	500,000
Validation	200,000
Test	300,000
UpdTest	25,738

5.2 Preprocessing

An error category shift analysis was performed where every potential reason causing the shift from one category to another was listed out to understand why the model could have possibly correct the sentence and made the shift. In Table 5.2 these reasons are listed out for each category shift.

Table 5.2: List of causes leading to error category shift of sentences.

		Input			
		Cat A	Cat B	Cat C	Cat D
Predicted	Cat A	• No Change	• Grammatical errors are corrected	• Spelling errors are corrected	• Both, spelling & grammatical errors are corrected
	Cat B	• New words introduced • Words deleted • Sequence change • Existing correct word altered but no spelling error	• No change • New words introduced • Words deleted • Sequence changes of existing words • Existing correct word altered but no spelling error	• New words introduced • Words deleted • Sequence changes of existing words • Existing correct word altered but no spelling error • Spelling errors corrected to non-matching words	• New words introduced along with correction of erroneous words. • Erroneous words deleted • Sequence changes with spelling correction. • Existing correct word altered but no spelling error • Spelling errors corrected to non-matching words • Spelling errors corrected
	Cat C	• New words with spelling errors • Existing correct word altered but no	• New words with spelling errors introduced with corrected grammar • Sequence changed with correct	• No change. • New spelling errors introduced. • Existing spelling errors corrected with new spelling errors introduced.	• New words with spelling errors with corrected grammar • Existing errors corrected but new words spelling errors

		grammatical error	grammar but spelling error • Existing words corrected but misspelled		• Existing grammatical errors got corrected. • Existing errors corrected but other existing words misspelled
	Cat D	• New misspelt words and grammatical errors • Existing words altered for spelling and grammatical errors • Word deletion and Existing correct words altered with spelling errors • Sequence changed & existing correct words altered with spelling errors	• New misspelt words • Some existing words misspelled • Existing grammatical error corrected but new grammatical and spelling error introduced	• New words with grammatical and / or spelling errors • Existing correct words sequence changed • Existing spelling error corrected but new grammatical & spelling error introduced	• No change. • New spelling OR / AND grammatical error added • New misspelt words with corrected existing spelling error • New grammatical error with corrected existing grammatical error

In order to stress test model's ability and understand its limitations, two sets of input-target sets were prepared, Synthetic Errors and Natural Errors. In Synthetic Errors category various error buckets were created in order to have 2%, 4%, 6%, 8%, 10%, 20%, 30%, 40%, 50% and 60% errors to observe with which percent error induced input does the model lose its efficiency. It is very complicated to create different percentage error buckets for Natural Errors. Therefore, if the models do not differentiate between the two types of error, then it is safe to conduct stress test on Synthetic Errors and assume the same for Natural Errors.

Synthetic errors were introduced by replacing characters at random with respective percentage. For example, in 10% error data, around 10% of the characters were replaced randomly.

Natural errors are likely to occur when the models are utilized in real world scenarios. Hence, it is important to generate natural errors and test them. In order to create this, input and target sentence are compared. The first error encountered is retained and the rest of the sentence is replaced with the target. For example, input: "I am gone to the Palace, to seen the Parade" target: "I am going to the Palace, to watch the Parade" and natural error input: "I am gone to the Palace, to watch the Parade". While the input has two errors, 'gone' and 'seen' as supposed to 'going' and 'watch', only 'gone' is retained as an error and the rest of the sentence is corrected by taking from the target. Hence, this is how natural error was induced in the entire dataset.

5.3 Language Models

Sequence to Sequence (Seq2Seq) [22] models are an industry standard & used for various kinds of text analytics and natural language processing (NLP) tasks. Seq2Seq model employs advanced deep learning algorithms with elaborate encoder-decoder architectures to perform such tasks. The encoder in Seq2Seq models reads the input sequence and summarizes it into a fixed length representation called state or context vector. The decoder uses this context vector to generate an output which is task specific. The choice of architecture for each Seq2Seq model depends on the task at hand.

Seq2Seq models achieve their intended target tasks by conversion of textual data from one domain to another. Such conversion operations include machine translation [23], text summarization [24], image captioning [25], chatbot interactions [26] & query engines [27]. Generating error-free sentences from erroneous text is another such task which may be accomplished by Seq2Seq modeling [22]. This error correction is achieved by analysis of the context, neighborhood and meaning of the sentence. Two such Seq2Seq models, namely, Bidirectional Auto-Regressive Transformers (BART) and MarianMT, are utilized in this work.

BART is a self-supervised denoising autoencoder that aids in the recognition of words and phrases. It is a pre-trained model that combines the advantages of auto-regressive transformers with bidirectional traversal. BART's working consists of two sequential steps: (i) adding noise to the input text, & (ii) reconstructing the correct output. The first step is achieved either by corrupting a predetermined character or sequence, or by using appropriate noise-generating functions. The reconstruction of the correct output sequence is achieved with usage of a language model with a Seq2Seq architecture that aids in learning and reconstruction of the output string by substituting valid tokens for noise [28]. BART's architecture allows for selection of different nosing functions for effective learning. The BART model from the SimpleTransformers library [29] is utilized for this work.

MarianMT [30], a sophisticated neural translation framework that uses a sequence-to-sequence model, is mostly employed for text translation. They are C++ translational frameworks that are effective and independent. The foundation of MarianMT is also a typical encoder-decoder design. In the current work, MarianMT is used to transform incorrect English sentences into their corrected equivalents. It's assumed that such a transformation is equivalent to language translation task where incorrect sentences form a variation of the English language. The MarianMT model employed in this work uses pre-trained models from the SimpleTransformers library [29]. The decoder, "Helsinki-NLP/opus-mt-NORTH EU-NORTH EU," is trained with transfer learning in this work for converting sentences from erroneous English language back to corrected English, thereby, reducing textual error.

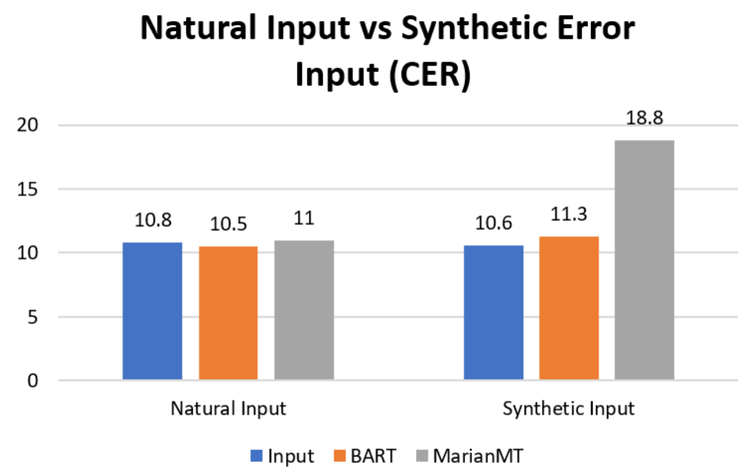
Both the pretrained models, BART and MarianMT, are trained on a train set of 500,000 records with a batch size of 20 sentences in each batch. The model employs beam search with a beam width of 5 for the prediction of tokens from sequence of words. A fivefold cross validation was also employed during the training process. The training lasted for ~27 hours on a GTX Titan Black GPU with 12 GB RAM to complete one epoch.

CHAPTER – 6
RESULTS AND ANALYSIS

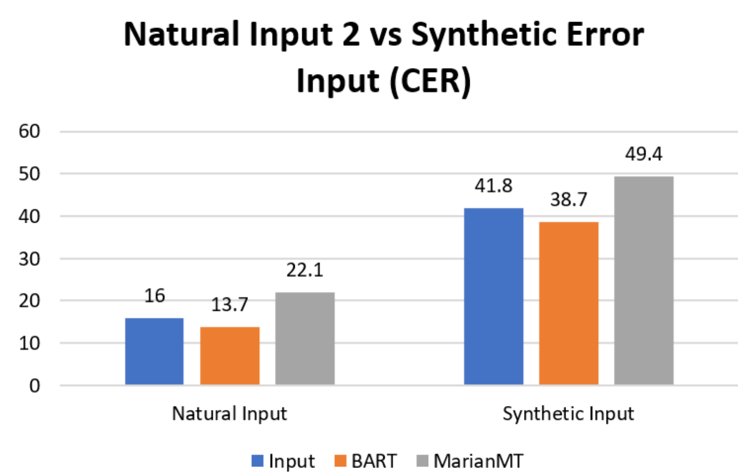
The models were tested on both Natural Errors, Synthetic Errors and original Input. The Character Error Rate (CER) and Word Error Rate (WER) for each out generated were computed. CER and WER scores of all the results obtained was analyzed and understood to address the research questions presented in Section 4.4.

6.1 RQ1 - Do the models distinguish between natural error and synthetic error?

BART and MarianMT were tested with the above mentioned data to check if they perform similarly with naturally occurring errors and synthetically introduced errors. It was observed that BART performed uniformly with both the types of errors with respect to CER scores, while MarianMT had higher CER score with synthetically modified errors as seen in Fig. 6.1.



(a)



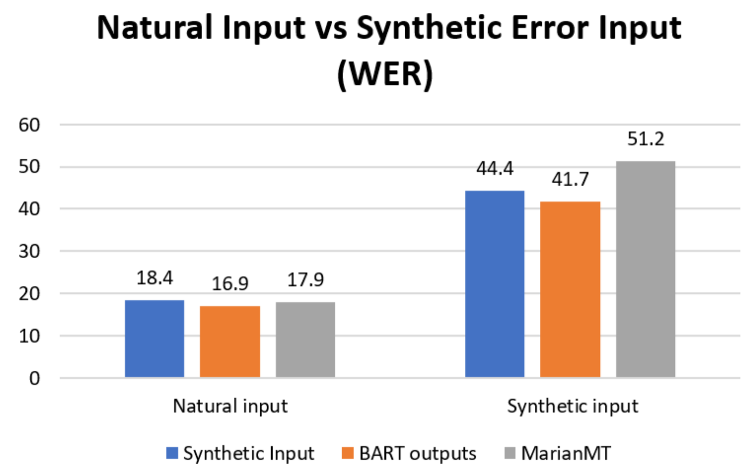
(b)

Fig. 6.1 (a) Models CER Scores on Synthetic and Original Natural Input (b) Models CER Scores on Synthetic and Natural Error Data

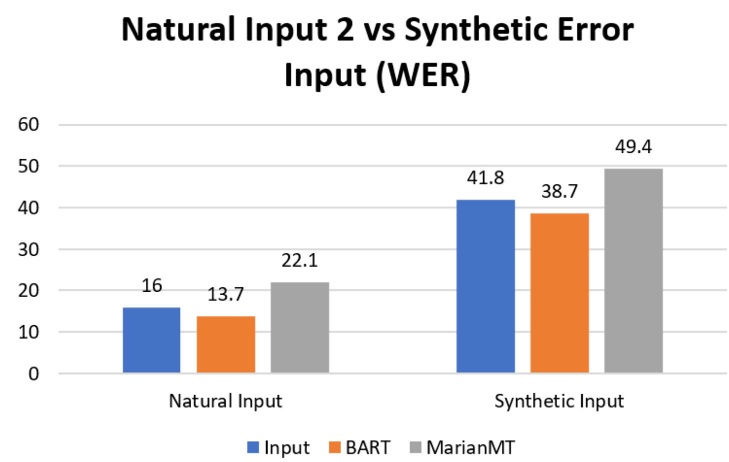
This could have occurred due to the training of the models being different. Since these models are pre-trained, MarianMT might have been trained highly on naturally occurring erroneous dataset and was biased towards it.

Although BART had a higher CER score for the synthetic error input by a margin of 0.8%, which might have occurred due to training on biased dataset, it is safe to conclude that BART performed similarly on both the datasets.

With respect to WER scores, though BART's results are varied with natural and synthetic errors, it coincides with the Input WER scores. Whereas MarianMT's WER scores, similar to CER scores, are much higher than Input data error rate as shown in Fig. 6.2.



(a)



(b)

Fig. 6.2 (a) Models WER Scores on Synthetic and Original Natural Input (b) Models WER Scores on Synthetic and Natural Error Data

Therefore, a study on BART's capability of correcting various levels of erroneous inputs on synthetic dataset can be beneficial on the study of break-even point of the model.

6.2 RQ2 - Do the BART and MarianMT models behave differently for Synthetic errors?

A study on WER data was conducted for the model outputs on various error rate levels for synthetic error datasets as shown in Fig. 6.3. It is observed that BART has lower WER score up to 20% erroneous input data. Above 20% error input, BART produced results with much higher WER score, more than input.

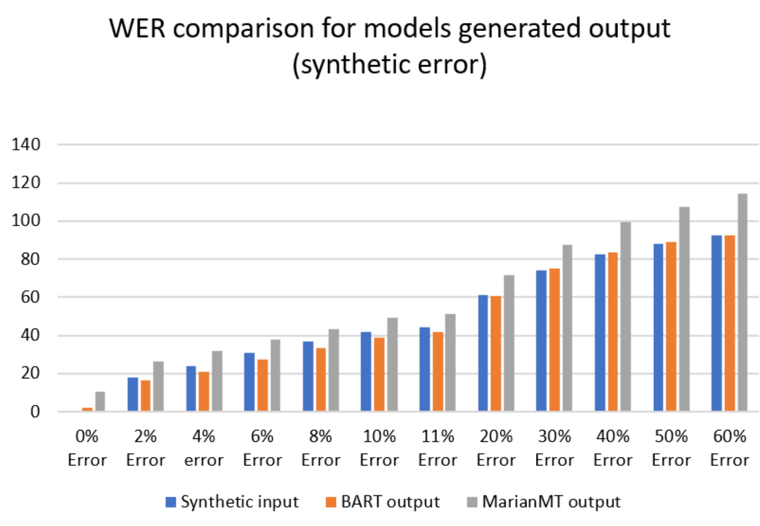


Fig. 6.3 WER scores for BART and MarianMT models trained on Synthetic Data

Initially MarianMT started off producing poor results with higher WER scores in contrast to the input. The model performance tends to be the same throughout all the error rate input data. While BART portrays a clear break-even point, MarianMT needs to be trained further to draw any conclusions.

Interesting results were generated by CER comparison for BART and MarianMT. BART results had higher CER score than the input CER score throughout, but the difference is miniscule. MarianMT's CER score, though higher than input CER score, was growing on par with input, but after a while it shoots up drastically as the input error rates increase as shown in Fig. 6.4.

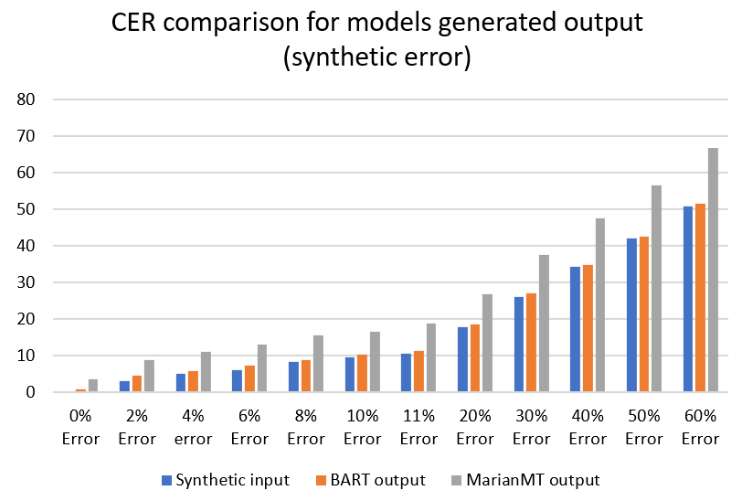
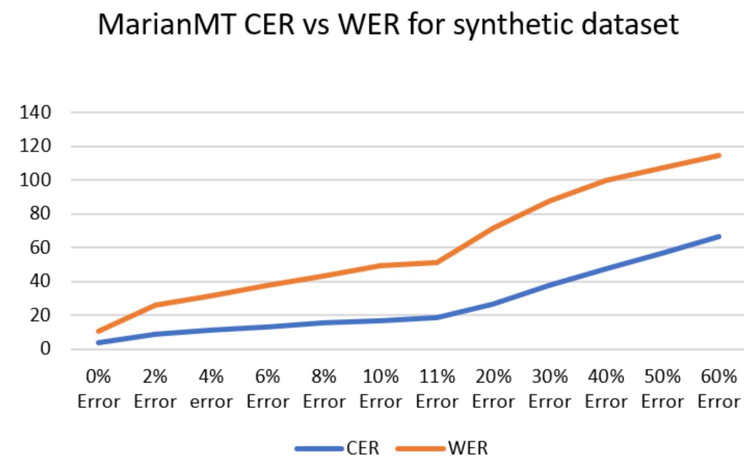


Fig. 6.4 CER scores for BART and MarianMT models trained on Synthetic Data

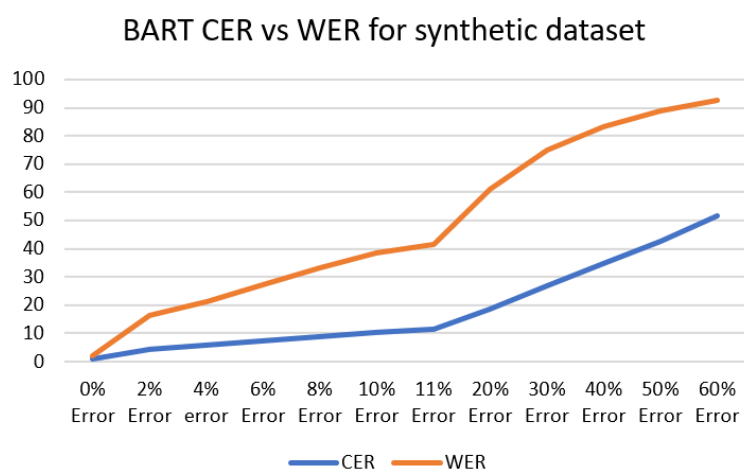
Upon analyzing the above figures, it is observed that the results are pointing towards BART having better correction ability and hence a break even analysis for BART is conducted. MarianMT requires further analysis and study to draw conclusions.

6.3 RQ3 - Do the models behave differently for CER and WER metric?

With a closer look on the graph trends, it was noticed that the input CER score uniformly increases with the increase in error percentages, forming a linear trend, as shown in Fig. 6.5. On the contrary, WER scores for the input data increases non uniformly throughout the error percentages.



(a)



(b)

Fig. 6.5 (a) CER vs WER comparison of MarianMT (b) CER vs WER comparison of BART

BART generated output CER follows a trend very similar to the Input CER whereas the MarianMT generated CER can be observed to be slightly higher than that of Input and BART generated CER throughout all the error percentages. But it should be noted that MatrianMT CER also follows a linear trend. Similarly, BART generated WER also follows a similar trend to the input WER whereas MarianMT generated

WER can be observed to be slightly higher than the Input and BART generated WER throughout all the error percentages, as shown in Fig. 6.6.

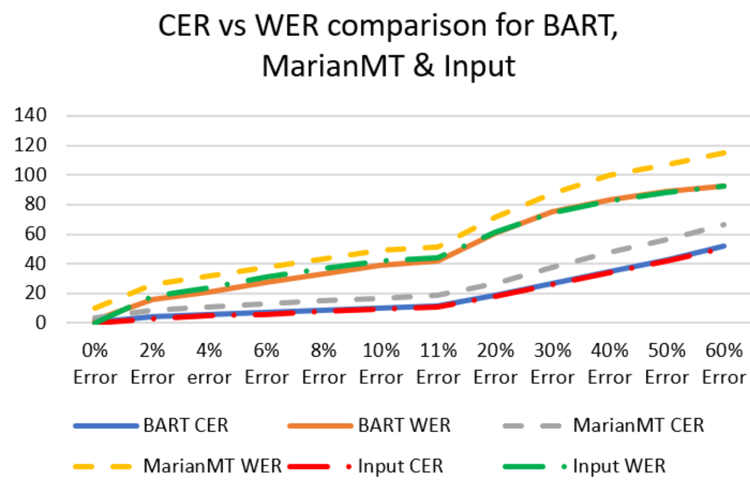


Fig. 6.6 CER vs WER comparison of the Models with Input

Although BART's output did not perform much better than the Input data, it followed a similar trend. But in the case of MarianMT the model output is seen to perform worse than BART output data. Finally, it can also be noted that the gap between the CER scores and WER scores of both the models increases with the increase in the error percentage. Hence, the gap between the performance of the models increases.

6.4 RQ4 - At what level the models trend for prediction output changes?

It can be observed that BART starts off with a lower WER score and continues to follow the trend until 26% erroneous synthetic dataset, after which BART starts generating outputs resulting in higher WER, as shown in Fig. 6.7.

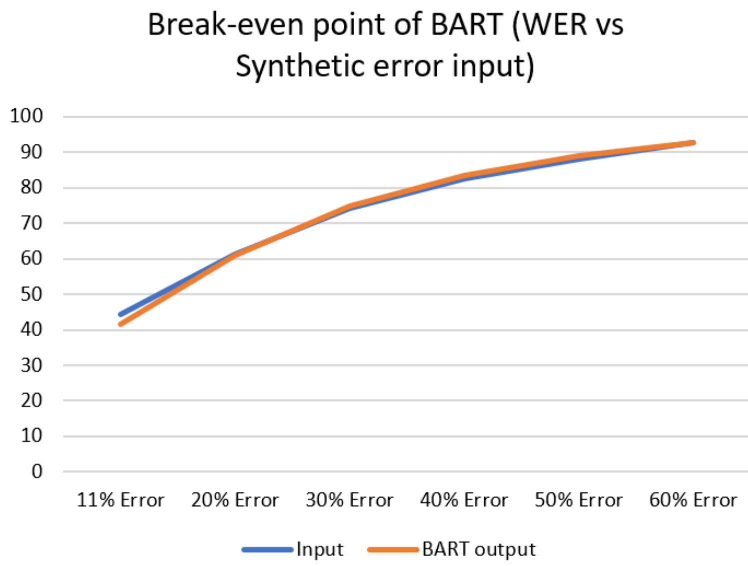


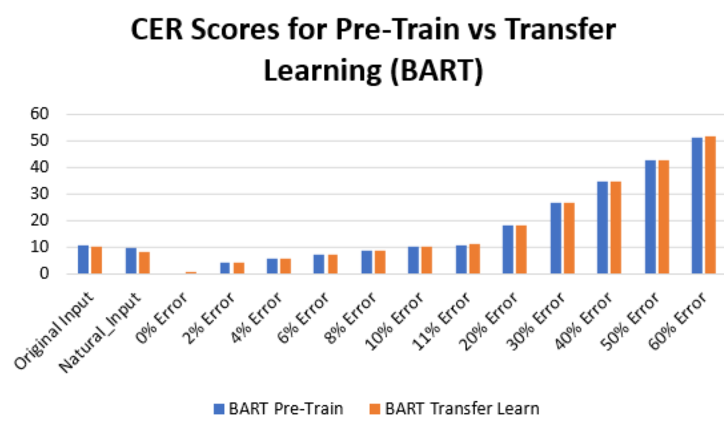
Fig. 6.7 WER comparison of BART and Input

This trend suggests that BART is capable of correcting errors efficiently until 26% erroneous input, after which it starts producing outputs with a higher WER than the input. Hence, 26% erroneous inputs can be taken as the break-even point of BART.

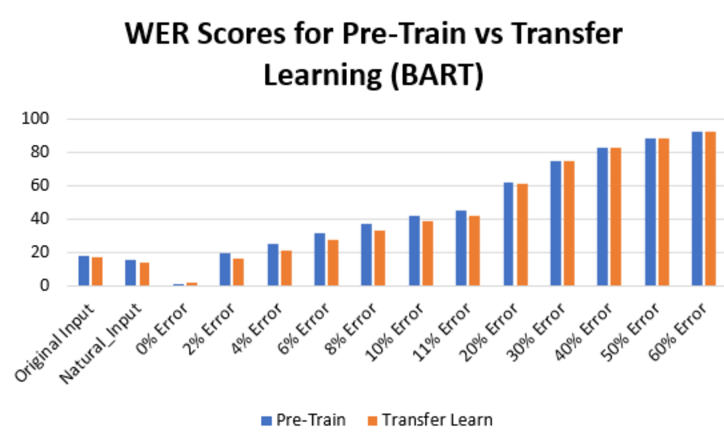
6.5 RQ5 - Do Transfer Learning models provide better prediction to the input sentences than pre-trained models?

A comparative understanding between BART pre-train and transfer learnt model with respect to CER score provides an interesting outcome, as shown in Fig. 6.8 (a). In case of sentences where there is presence of natural errors we observe that, the Transfer Learn BART model performs better than Pre-train BART only by a small margin. With respect to sentences containing synthetic errors, we observe that up to 40% error rate, transfer learn model learning performed better than pre-train model learning. As more and more errors are introduced, we observe that both models' learnings produced a similar CER score.

The natural errors present in sentences have a smaller WER score compared to the Pre-Train model, thus performing better for natural errors with respect to the BART model, as shown in Fig. 6.8 (b). Upon the introduction of synthetic errors, the Transfer Learn model performed better than Pre-train models up to 30% error rate. As more and more errors were introduced, the Transfer learn model learning and the Pre-trained model learning produced similar WER scores.



(a)

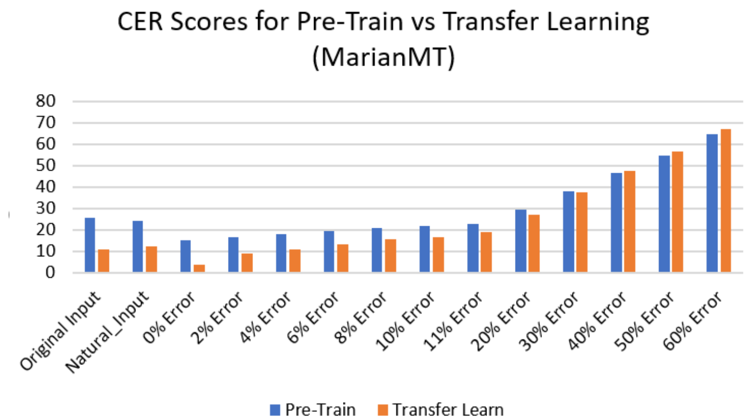


(b)

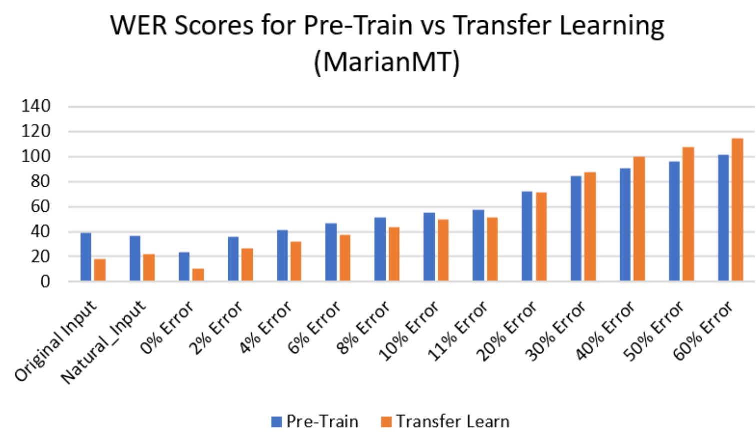
Fig. 6.8 (a) CER score trend for BART (b) WER score trend for BART

With respect to sentences containing natural errors, it is observed that the Transfer-learnt MarianMT model learning significantly outperformed the pre-train model, as shown in Fig. 6.9 (a). A similar outcome was arrived for synthetic errors as well. The transfer learnt model learning reported better predictions than the pre-train model up until an error rate of 30%. When more errors were introduced, both models produced similar WER scores with a very small margin.

A closer look into the above graph suggests that, with respect to sentences containing natural errors, Transfer Learnt model learning performed significantly better than pre-train model learning, as shown in Fig. 6.9 (b). The similar conclusion can also be drawn with respect to synthetic errors up to 20% error rate. Beyond that we observe that more introduction of errors has to lead to pre-train model learning having a similar or slightly better prediction than transfer-learnt.



(a)



(b)

Fig. 6.9 (a) CER score trend for MarianMT (b) WER score trend for MarianMT

CHAPTER – 7
CONCLUSION

In conclusion, the experiments began by developing two models, BART and MarianMT, and extensively training them on an English language dataset. Two different datasets, natural and synthetic, were then constructed and used to test the models' performance. It was determined that BART performed similarly on both datasets, indicating that it could be tested on synthetic data to study its break-even point. The subsequent experiments analyzed the models' performance at various error rates and their ability to correct errors using the CER and WER metrics. Another experiment was conducted to compare the performance of transfer learning models versus pre-trained models in terms of CER and WER. The results indicated that transfer learning models outperformed pre-trained models in most cases. Finally, the experiments concluded by testing BART's break-even point, which was found to be in the vicinity of 26% erroneous input data.

7.1 Inferences of the Research Questions

After analyzing research question, the following are the inferences drawn from each research question:

- It has been observed that BART performs similarly for both natural and synthetic errors, whereas MarianMT has higher error rate for synthetic error.
- When tested on synthetic datasets, BART and MarianMT demonstrated notable differences in performance. Specifically, BART showed greater promise than MarianMT in terms of CER and WER, consistently producing better results across all error categories.
- BART's performance follows a similar trend to that of the input data. MarianMT performed relatively poorly on both metrics. The gap between

BART's and MarianMT's performance increased, with MarianMT's performance becoming worse in comparison to BART.

- The generated chart indicates that BART has achieved a successful break-even point, which appears to be in the vicinity of 26% erroneous input data.
- On multiple occasions, it was noticed that the transfer learning model performed better than the pre-trained models. However, it should be acknowledged that the pre-trained models had an advantage over the transfer learning models in some of the error categories.

After a thorough study of BART and MarianMT's ability of correcting various errors, it can be concluded that BART has a breakeven point at 26% erroneous data. MarianMT requires more study and analysis to find it's breakeven point.

7.2 Future Scope of the Work

The Model's performance on various error categories needs to be studied better in detail to understand the reason behind the poor performances across some categories.

The model was tested on natural (typed data) but it has a scope to be extended to other types of erroneous data such as OCR and speech-recognized data to study the performance under various errors even better.

The transfer learnt model was observed to outperform the pre-trained model and this occurred without hyperparameter optimization. The models hyper parameters could be optimized to generate even better results.

The poor performance of MarianMT could be studied more in detail to achieve the potential breaking point of MarianMT.

REFERENCES

[1] Barber K (2007) The Anthropology of Texts, Persons and Publics (New Departures in Anthropology). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511619656.

[2] Smith R (2007) An Overview of the Tesseract OCR Engine - 1995. Proc. Ninth Int. Conference on Document Analysis and Recognition (ICDAR), IEEE Computer Society, pp. 629-633.

[3] Nassif A B, Shahin I, Attili I, Azzeh M, Shaalan K (2019) Speech Recognition Using Deep Neural Networks: A Systematic Review. IEEE Access, vol. 7, pp. 19143-19165, doi: 10.1109/ACCESS.2019.2896880.

[4] Singrodia V, Mitra A, Paul S (2019) A Review on Web Scraping and its Applications. Int. Conf. on Computer Communication and Informatics (ICCCI), Coimbatore, India, pp. 1-6, doi: 10.1109/ICCCI.2019.8821809.

[5] Sewwandi U, Ranathunga L, Wijethunge S (2021) A Rule Based Approach for Detection and Correction of Grammar Errors in Written Active Voice Sinhala Sentences. 21st International Conference on Advances in ICT for Emerging Regions (ICter), Colombo, Sri Lanka, pp. 159-164, doi: 10.1109/ICter53630.2021.9774800.

[6] Sutskever I, Vinyals O, Le Q V (2014) Sequence to Sequence Learning with Neural Networks. In: New York, US, Cornell University. https://arxiv.org/abs/1409.3215v3.

[7] Lewis M, Li, Y, Goyal N, Ghazvininejad M, Mohamed A, Levy O, Stoyanov V, Zettlemoyer L (2019) BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. New York, US, Cornell University. https://arxiv.org/abs/1910.13461v1.

[8] Junczys-Dowmunt M, Grundkiewicz R, Dwojak T, Hoang H, Heafield K, Neckermann T, Seide F, Germann U, Aji A F, Bogoychev N, Martins A F T, Birch A (2018) Marian: Fast Neural Machine Translation in C++. Proceedings of ACL 2018, System Demonstrations, pp. 116–121, Melbourne, Australia.

[9] Liang Z, Youssef A (2020) Performance Benchmarking of Automated Sentence Denoising using Deep Learning. IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 2020, pp. 2779-2784, doi: 10.1109/BigData50022.2020.9377985.

[10] Saluja R, Adiga D, Chaudhuri P, Ramakrishnan G, Carman M (2017) Error Detection and Corrections in Indic OCR Using LSTMs. 14th IAPR International

Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 2017, pp. 17-22, doi: 10.1109/ICDAR.2017.13.

[11] Tan M, Chen D, Li Z, Wang P (2020) Spelling Error Correction with BERT based on Character-Phonetic. IEEE 6th International Conference on Computer and Communications (ICCC), China, 2020, pp. 1146-1150, doi: 10.1109/ICCC51575.2020.9345276.

[12] Alikaniotis D, Raheja V (2019) The unreasonable effectiveness of transformer language models in grammatical error correction. ACL 2019 - Innovative Use of NLP for Building Educational Applications, BEA 2019 - Proceedings of the 14th Workshop.

[13] Bryant C, Briscoe T (2018) Language Model Based Grammatical Error Correction without Annotated Training Data. Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, pp. 247–253, New Orleans, Louisiana. Association for Computational Linguistics.

[14] Zhao W, Wang L, Shen K, Jia R, Liu J (2019) Improving Grammatical Error Correction via Pre-Training a Copy-Augmented Architecture with Unlabeled Data. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 156–165, Minneapolis, Minnesota. Association for Computational Linguistics.

[15] Huang L, Li J, Jiang W, Zhang Z, Chen M, Wang S, Xiao J (2021) PHMOSpell Phonological and morphological knowledge guided Chinese spelling check. ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int. Joint Conf. on Natural Language Processing, Proceedings of the Conference.

[16] Zhang R, Pang C, Zhang C, Wang S, He Z, Sun Y, Wu H, Wang H (2021) Correcting Chinese Spelling Errors with Phonetic Pre-training. Findings of the Association for Computational Linguistics: ACL-IJCNLP, pp. 2250–2261, Online. Association for Computational Linguistics.

[17] Xu H D, Li Z, Zhou Q, Li C, Wang Z, Cao Y, Huang H, Mao X L (2021) Read, Listen, and See: Leveraging Multimodal Information Helps Chinese Spell Checking. Findings of the Association for Computational Linguistics: ACL-IJCNLP, pp. 716–728, Online. Association for Computational Linguistics.

[18] Rohit R, Gandheesh S A, Suriya K S (2023) "Report: System for Enhancing Accuracy of Noisy Text using Deep Learning Architecture". Amrita School of Computing, Amrita Vishwa Vidyapeetham, Bengaluru, India.

[19] Ahmed H, Hoque M, Alam M D M (2018) Nature of errors and mistakes in the English writings of graduating students in Bangladesh: A case study. IIUC Studies. 15. 11-22. 10.3329/iiucs.v15i0.49341.

[20] Mounika Y, Tarakaram Y, Prasanna Y L, Gupta D, Pati P B (2022) Automatic Correction of Speech Recognized Mathematical Equations using Encoder-Decoder Attention Mode. India Council International Conference (INDICON, IEEE Computer Society.

[21] C4 Data: www.tensorflow.org/datasets/catalog/c4. (2019)

[22] Sutskever I, Vinyals O, Le Q V (2014) Sequence to sequence learning with neural networks. Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'14). MIT Press, Cambridge, MA, USA, pp. 3104–3112.

[23] Cho K, Merriënboer B V, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734, Doha, Qatar. Association for Computational Linguistics.

[24] Shi T, Keneshloo Y, Ramakrishnan N, Reddy C K (2021) Neural Abstractive Text Summarization with Sequence-to-Sequence Models. ACM/IMS Trans. Data Sci. 2, 1, Article 1 (February 2021), 37 pp. https://doi.org/10.1145/3419106.

[25] Malla A, Jafar M A, Ghneim N (2022) Image captioning model using attention and object features to mimic human image understanding. J Big Data 9, 20. https://doi.org/10.1186/s40537-022-00571-w.

[26] Ali A, Amin M (2019) Conversational AI Chatbot Based on Encoder-Decoder Architectures with Attention Mechanism. Artificial Intelligence Festival 2.0, NED University of Engineering and Technology, Karachi, Pakistan 10.13140/RG.2.2.12710.27204.

[27] Dehghani M, Rothe S, Alfonseca E, Fleury P (2017) Learning to Attend, Copy, and Generate for Session-Based Query Suggestion. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM '17). Association for Computing Machinery, New York, NY, USA, 1747–1756. https://doi.org/10.1145/3132847.3133010.

[28] Subramaniam L V, Roy S, Faruquie T A, Negi S (2009) A survey of types of text noise and techniques to handle noisy text. ACM International Conference Proceeding Series. 115-122. 10.1145/1568296.1568315.

[29] Transformers: simpletransformers.ai/docs/usage/. (2021)

[30] Kumar A, Pati P B (2022) Offline HWR Accuracy Enhancement with Image Enhancement and Deep Learning Techniques. Int. Conf. on Machine Learning & Data Engineering. doi:10.1016/j.procs.2022.12.399.

[31] Zhang W E, Sheng Q Z, Alhazmi A, Li C (2020) Adversarial Attacks on Deep-learning Models in Natural Language Processing: A Survey. ACM Trans. Intell. Syst. Technol. 11, 3, Article 24, 41. https://doi.org/10.1145/3374217.

[32] Pruthi D, Dhingra B, Lipton Z C (2019) Combating Adversarial Misspellings with Robust Word Recognition. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5582–5591, Florence, Italy. Association for Computational Linguistics.

[33] Zhang Z, Han X, Liu Z, Jiang X, Sun M, Liu Q (2019) ERNIE: Enhanced Language Representation with Informative Entities. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1441–1451, Florence, Italy. Association for Computational Linguistics.

[34] Kantor Y, Katz Y, Choshen L, Cohen-Karlik E, Liberman N, Toledo A, Menczel A, Slonim N (2019) Learning to combine Grammatical Error Corrections. Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pp. 139–148, Florence, Italy. Association for Computational Linguistics.

[35] Jayanthi S M, Pruthi D, Neubig G (2020) Neuspell: A neural spelling correction toolkit. Proc. Conf. on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP) - 2020, pp. 158–164.

[36] Hangaragi S, Pati P B, Neelima N (2022) Accuracy Comparison of Neural Models for Spelling Correction in Handwriting OCR Data. 4th International Conference on Communication, Computing and Electronics Systems (ICCCES), doi:10.1007/978-981-19-7753-4_18.

[37] Urs-Viktor Marti et al (2019) IAM Dataset: fki.tic.heia-fr.ch/databases/iam-handwriting-databasehttps://fki.tic.heia-fr.ch/databases/iam-handwriting-database.

[38] Fu K, Huang J, Duan Y (2018) Youdaos Winning Solution to the NLPCC Task Challenge: A Neural Machine Translation Approach to Chinese Grammatical Error Correction. Natural Language Processing and Chinese Computing. NLPCC. Lecture Notes in Computer Science, vol 11108. Springer, Cham. https://doi.org/10.1007/978-3-319-99495-6_29.

[39] Ali A, Renals S (2018) Word Error Rate Estimation for Speech Recognition: e-WER. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 20–24, Melbourne, Australia.

[40] Sreevidhya V, Narayanan J (2021) Short descriptive answer evaluation using word-embedding techniques. 12th Int. Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-4.

[41] Rohit R, Gandheesh S A, Suriya K S, Pati P B (2023) System for Enhancing Accuracy of Noisy Text using Deep Network Language Models. The 2023 International Conference for Convergence in Technology (I2CT), Bombay, India.

PUBLICATIONS

Conference Paper

Rohit R, Gandheesh S A, Suriya K S, Pati P B (2023) “System for Enhancing Accuracy of Noisy Text using Deep Network Language Models”. The 2023 International Conference for Convergence in Technology (I2CT), Bombay, India.

Journal Paper

“Error Category-Shift Behavior Analysis for Text Enhancement with BART and MarianMT”. Submitted to Computer Speech and Language. (Status: Under Review)

Patent Work

“System and Method for Enhancing Accuracy of Noisy Text using Deep Network Language Models”, Deeptech Ref. No.: AMRT-020300INS, April 2023. (Status: Filing)

Journal Paper 2

“A Comparative Study of Language Model Training on Synthetic errors and Natural errors”. (Status: To be communicated)