Roman Urdu to Urdu Machine Transliteration Using T5 Transformer

Muhammad Adeel[1], Usama Ahmed[1], Hassan Masood[2], Mudassir Saeed[2], and Usama Amjad[1]
[1]Department of Artificial Intelligence, School of Systems and Technology, University of Management and Technology, Lahore 54770, Pakistan
[2]Department of Computer Science, National University of Computer and Emerging Sciences, 852-B, Faisal Town, Lahore, Pakistan

ABSTRACT Transliteration is the process of simply analyzing the words in the resource language to the words in the goal language, without any change in meaning. This method transforms the syntax of a text in resource speech into characters of the target language, known as machine transliteration. Recent studies indicate that no dedicated transliteration machine currently exists that covers the issue of RU-U Machine Translation. Previous researchers have attempted to solve this problem using the deep learning techniques, particularly RNN model. Recurrent Neural Network (RNN) transformers are built to manage sequence input information, like natural language, for tasks like translation and text summarization. This model works better on short sentences than long sentences. In the proposed methodology, T5 transformers are encoder-decoder models that translate NLP issues into text-text format. T5 is a transfer learning and the transformers used in this paper are trained on 101 languages including resources language and after training on our parallel data set which consists of 1,107,156 sentences, the study achieved a remarkable result of 91.56 Blue Score.

INDEX TERMS deep learning, encoder decoder, low resource language, machine transliteration, T5 transformer, Roman Urdu, transfer learning

INTRODUCTION

Natural Language Processing (NLP) is a field of computer science that focuses on improving a computer's ability to understand human languages and to communicate with a computer system using natural language. Natural language software can comprehend user conversations. Natural language processing is an example of an artificial intelligence technology. Urdu has gained increasing attention in NLP research and is helping shape the burgeoning science of NLP. Urdu is an Indo-European language with an Indo-Aryan racial origin. It is widely spoken throughout Asia's continent, serves as the mother tongue of Pakistan, is also spoken in India, and functions as a second language in other nations. Because the Urdu script is developed from Arabic and Persian scripts, it is oriented from left to right, and the forms of the words are quite similar to those of Arabic and Persian. Historically, the design of language-processing systems has primarily focused on European languages, leading to their advanced development for human–computer interface design. However, unlike other languages, Urdu still receives limited research funding. This study focuses on Transliteration and the contextual interpretation of uncertain/ambiguous words. Natural Language Processing is linked to natural languages as well as machine translation. It explores how computers may assist in the interpretation of common statements or statements to create useful results. NLP analysts intend to gather information on how people understand and interpret language so that appropriate approaches and techniques may be developed for computers to manipulate and manage such languages to do needed tasks [1].

Translation is the process of transferring the text of one language into another while maintaining its original meaning. It may be used in both written and vocal communications. The basic goal of translation is to keep the source and destination languages' connotations similar. Translation's impact on our daily lives is mostly structural. Translation facilitates global communication and allows nations to form ties that lead to scientific, political, and cultural breakthroughs [2]. NLP researchers are drawn to languages like Urdu and Arabic, which have right-to-left script-writing systems. Pakistan's national language is Urdu. In Pakistan, there are around 11 million Urdu speakers, with over 300 million worldwide. Finally, a neat and accurate

Roman Urdu-Urdu parallel corpora of nearly 1107156 sentences was generated. The overall Urdu vocabulary in our corpus is 34,519 words, with a total Roman-Urdu dictionary of 21,019 words. It's worth noting that the Urdu vocabulary has extra terms than the Roman-Urdu vocabulary. Numerous phrases in Urdu have a gap between them yet are only considered one word.

The Roman Urdu text is unprocessed and must be processed. One of the data mining strategies is to reshape raw data using data preparation technologies. Translational models can effectively learn from preprocessed data. Transfer learning has developed as a strong approach in natural language processing, in which before being refined on a lower-level challenge, a model is first pre-trained on a data-rich task (NLP). T5 is an encoder and decoder paradigm that translates entire natural language processing difficulties into text-to-text format. This implies that an input and a goal sequence are indeed required for training. The model is given the input data through input. The target sequence is prefixed with a begin sequence item given to the decoder through the decoder response.

- There was no Machine Transliteration and no such work for Roman Urdu to Urdu Transliteration before this work.
- Before this, the mT5 model has not previously been applied to any transliteration work.
- For Roman Urdu to Urdu Transliteration, in our Project, we employed the mT5 (Multilingual pre-trained Text-to-Text Transfer Transformer) Model.
- We achieved a 91.56 Blue Score after using this mT5 Model and conducting all the experiments.

The basic goal of any transliteration task is to create grammatically acceptable and understandable output sentences. Models must handle the context to provide appropriate transliteration for little to moderate-sized sentences. Models must be able to hold phrases of various lengths.

LITERATURE REVIEW

Three pre-processed modules, a code-based re-placement, and a Unicode-based character map make up the RUTUT translator. The author of this work proposes the RUTUT, which includes pre-processing procedures, official character replacement, and Unicode-based drawing techniques. The author utilized the basic procedure, which involves feeding the RUTUT translator 2000 Roman Urdu words. Since 1917, a RU word has been translated perfectly into Urdu, demonstrating that RUTUT translator perfectly translates RU terms into Urdu language 95.8% of the time [1]. The encoders and decoders are usually two recurrent neural networks, with the decoder adopting a learning algorithm to focus on relevant bits of the source language. The first ever 1.1 million sentence RUTU parallel corpus, three state-of-the-art encoder and decoder models, and a complete empirical examination of these three models on the Roman-Urdu to Urdu parallel corpus Overall, the attention-based model provides cutting-edge performance, with a 70 BLEU score [2].

The presence of a dataset is just a prerequisite for undertaking research in a specific language. To that purpose, this study presents the Roman-Urdu-Parallel corpus, which contains 6.37 million pairs of sentences and the first huge RU parallel dataset is freely available. It's a massive corpus culled from a variety of quality-assured sources, annotated with crowd-sourcing methodologies. It shows the 92.7 million R terms and 92.8 million Urdu terms. MEHREEN ALAM makes three contributions in their paper: first, they create a large-scale data set, and then they conduct extensive qualitative and quantitative analyses on it. With a BLEU score of 84.67, we set the latest benchmark in machine transliteration [3]. Some practices are done to complete this void of linear datasets in low-resource languages [4] ten parallel datasets of English to Arabic pairing and Basque and Bengali and Bulgarian and Dutch and Hungarian and Polish and Russian and Turkish and also Ukrainian. More experiments include English-Hindi [5]. Current research sheds light on transliteration. This means translating a word in the source language (e.g. Roman Urdu) into an equivalent word in the target language (e.g. Urdu). Convert words from one secretary system to phonetic equivalent terms in another. To find out the performance of the proposed system in a

low-resource setting, the authors used many language pronunciation dictionaries extracted from so many news websites.

The author presented low-resource ma- chine transliteration system settings that combine several neural network-based techniques (encoder and decoder, focus on mechanism, input sequence source with pre-trained aligned representation, and target embedding) [6]. Neural networks have excelled in a variety of applications, ranging from computer vision to speech recognition. Traditional phrase-based statistical machine translation systems have been supplanted by machine translation techniques NLP. Even though Urdu is a morphologically rich language with a population of 0.1 billion people, no work has been done to create a publicly available RUTU linear Dataset to our knowledge. Our Roman-Urdu-to-Urdu dataset was gathered and developed. We gathered 5.4 million Urdu sentences and 0.1 million RU sentences by crawling and scraping from the internet. Utilizing the website, we transliterated RU sentences to Urdu sentences and vice versa using only a subset of the data collected. The total number of lines in the RU to Urdu parallel corpus that we were able to generate was 0.113 million. Our approach relies on the encoders and decoders offered by sequence to sequence. The input is a sequence, in this case, a RU sentence, and the output is another sequence, this time in Urdu. Each unit is an LSTM cell, which works well on longer sequences and is resistant to vanishing gradients [7].

The primary purpose of this paper is to provide an overview of the numerous linguistic resources available for Urdu language processing, to highlight distinct tasks in Urdu language processing, and to present some cutting-edge approaches. Finally, this paper seeks to cover all aspects of the recent surge in interest in Urdu language processing research, as well as its achievements. The first topic is the available datasets for the Urdu language. The peculiarities of the Urdu language, resource sharing between Hindi and Urdu, spelling, and morphology are all discussed. Pre-processing duties include stop-word removal, diacritics removal, normalization, and stemming, to name a few [8]. The different stages offered in the proposed system for translating standard English text into Urdu. Preprocessing in the source and target languages, word embedding, encoding, decoding, and then generating the target text are the steps involved in converting standard English text to Urdu. The most crucial task in the development of any neural machine translation system is corpus preprocessing [9]. The goal of this project is to enhance Roman-Urdu to Urdu script context-based transliteration. We offer an algorithm in this research that successfully solves transliteration difficulties. The system works by converting encoded Roman words into standard Urdu script words and matching them to the dictionary. If a match is detected, the word will be shown in the text editor. If there are several matches in the lexicon, the highest frequency terms are presented. In comparison to previous models and algorithms that operate for the transliteration of Raman Urdu to Urdu in context, the results of this method revealed its effectiveness and relevance [10].

The current study aims to investigate the language barriers that machine translators may face when translating Arabic translations of English proverbs. It also tries to prove the significance of human interaction in the process. addressing accuracy issues. To achieve these goals, we randomly selected a set of English proverbs, Researchers did qualitative analysis after translating the text into Arabic using Google Translate. On the one hand, the findings indicate that Google Translate is experiencing some difficulties and language barriers when it comes to translating the similar meaning of an English proverb into Arabic [11]. Transfer rules translate source-language text into target-language text using organizational and lexicon operations in transfer-based machine translation. These transferring rules can be created in a variety of methods, including such as hand- coding and analyzing parsed aligned multilanguage datasets. Handle lexical and structured base disagreements with a transfer-based approach [12].

A recent study of students at a university in Pakistan collected a dataset of textual information in RU. The authors used a mobile phone usage dataset to accomplish this. There are 116 users and 346, 455 text messages in the database. In Pakistan, Roman Urdu text is the

most extensively used method of sending text messages. Our user research provided some interesting findings, such as the ability to quantitatively illustrate that many words were written with multiple spellings [13]. We describe a novel method for incorporating transliteration into machine translation from Hindi-Urdu. We suggest two innovative probabilistic models for the problem, based on conditional and joint probability formulations. When translating a particular Hindi word given the context, our methods represent both transliteration and translation, whereas, in prior work, transliteration was only employed for translating out-of-vocabulary words. We utilize transliteration to distinguish between Hindi homonyms that can be translated, transliterated or transliterated differently depending on the context. In comparison to 14.30 for a baseline phrase-based system and 16.25 for a system that transliterates OOV terms in the baseline system, we get final BLEU scores of 19.35 and 19.00 [14].

To translate some text from one language to another language, in-depth knowledge of both the source and target language is required. Without such knowledge, the process of translation becomes cumbersome, and the result is not reliable. The major difference between English and Urdu language is due to the difference in their sentence structure. English has a "Subject + Verb + Object" sentence structure while on the other hand the Urdu language has a "Subject + Object + Verb" sentence structure. This difference can be classified into two categories: unilateral or bilateral. This classification depends upon the direction of the translation which could be either from English (target language) to Urdu (source language), Urdu (source language) to English (target language), or in both directions. Three fundamental approaches are used in machine translation. The purpose of this study is to devise and evaluate a unique strategy for resolving the problem of translation from Roman Urdu to English. The method utilized to build this realistic model is separated into three steps, each of which works to accomplish its goal [15]. Tokenization is executed using a self-maintained dataset and its associated tag set. The syntactical framework is covered by writing the Urdu POS tagger based on grammatical principles. To translate Roman Urdu into English, we created the grammatical structures of several phrases. Transect performed better and provided more accurate results than Google Translator [16].

In this paper, we suggest an alternative to classic statistical MT that uses recurrent neural networks (RNNs) (SMT). We compare the performance of the SMT phrase table to that of the suggested RNN to increase the MT output quality [16]. In their study, Blossom et, they developed a comparable paradigm based on the encoder and decoder concepts. For the encoder, they employed a convolutional n-gram model (CGM), and for the decoder, they used a mix of inverse CGM and an RNN. The model's performance was assessed by rescoring the n-best selection of phrases from the SMT phrase table [17]. This research investigates the use of triangulation and transliteration to improve Urdu to Hindi-English machine translation. They begin by introducing triangulated and transliterated phrase tables from Urdu-English and Hindi-English phrase translation models to create an Urdu-to-Hindi SMT system. Our phrase translation technique outperforms the baseline Urdu-to-Hindi system by 3.35 BLEU points. This method helps to enhance the Hindi-to-English translation algorithm [18]. Based on the research in this paper, they developed an interactive machine translation system that provides support for idioms, homographs, gender, and words with plural and singular meanings together, the corpus' ability to expand and answer inquiries, as well as its ability to develop to a greater spectrum of coverage, are both positives translated text is sorted. Ordering is a difficult problem for computers to solve, but it is much easier for humans. The interactive system makes the user's life easier., thus meeting the basic goal of research in this direction, which is to facilitate the user and improve the task efficiency during the translation process.

Our MT system is especially well-suited to the situations for which it has been trained. This shows that we can create MT systems adapted to domain-specific demands using the concepts outlined here. Furthermore, given the size of our corpus, it appears to be suitable for embedded devices with limited memory. This MT system uses phrase-based example sets, which offers it broader coverage [19]. Jianmo Ni et al, give the first investigation of effectively integrating de-

derived from text-to-text converters (T5). T5 is proven to generate constant additional gains when scaled up from millions to billions of parameters. Furthermore, when employing sentence embeddings, our encoder-decoder approach reaches a new state-of-the-art on STS [20]. The creation of English to Tamil, English to Hindi, English to Malayalam, and English to Punjabi language pairings is the emphasis of this machine translation common work in Indian languages and produces the best BLUE score for each translation [21] Work in this domain to improve the value of QA systems, allowing consumers to better understand privacy regulations before consenting to them. To create questions, this article leverages current deep learning models such as T5, The T5- small model with labels improves its METEOR and ROUGE-L scores by 2.46 percent and 3.67 percent, respectively [22]. Dabre, R offers a survey on many language neural machine translation (MNMT), a hot topic in recent years. As a result of translation knowledge transfer, MNMT has proved effective in enhancing translation quality (transfer learning) [23] Zeeshan et al, was trained a corpus using two NMT Models (LSTM and transformer Model), and the results were compared to the desired translation using the many languages evaluation understudy (BLEU) score.

On NMT, the LSTM Model enhances the BLEU score by 0.067 to 0.41, however, the Transformer model enhances the BLEU score by 0.077 to 0.52, which is better than the LSTM Model score [24]. We looked into using a T5 model to help with four code-related activities: automatic problem correction, assert statement generation in test procedures, code summarizing, and code mutant insertion [25]. The Chatbot Interaction with Artificial Intelligence using the T5, and when training data is supplemented with the T5 model, we find that all models improve, with an average improvement in classification accuracy of 4.01 percent. The RoBERT model, which was trained using T5-enhanced data and attained 98.96 percent classification efficiency, was the best [26].

Raffel et al, introduced a uniform framework that translates all text-based language issues into a T5 format to investigate the landscape of transfer learning approaches for NLP [27]. Fine-tuned Generative Pre-Trained Transformer 2 (GPT-2) model, Text-To-Text Transfer Transformer (T-5) model, and Bidirectional Encoder Conceptions via Transformers (BERT) model are the three pre-trained transformer models compared. They also found that by lowering the incidence of repetition, the transliteration-based Generative Pre-trained Transformer 2 (GPT-2) model obtains superior summarization performance [28].

Analyzing the empirical capabilities of current state-of-the-art sequence-based neural architectures in assessing tiny computer programs is extremely important. T5 Transformer can compute the output for both valid and Python code blocks with greater than 65 percent efficiency, according to tests [29]. The author introduces mT5 and mC4: massively multilingual variations of the T5 model and the C4 dataset in this paper. They demonstrated that the T5 recipe is easily adaptable to a multilingual setting and obtained excellent results on a variety of criteria. They also developed a simple strategy for avoiding illegal predictions that can occur during zero-shot evaluation of multilingual pre-trained generative models. They make available all of the code and pre-trained datasets used in this publication to promote future multilingual research [30]. In the existing text-to-text passage re-ranking model, the Author introduces the concept of multi-view learning. The suggested text-to-text multi-view architecture uses an instance mixing strategy to combine the text-generation and text-ranking objectives. The text generation view is useful in increasing re-ranking efficacy, according to our empirical study.

Furthermore, the findings imply that the mixing rate for sampling cases from diverse perspectives is the most relevant aspect. They also increase the re-ranking depth of the multi-view model to test its re-ranking robustness. Even though the links between distinct views remain ambiguous, see multi-view learning as a flexible framework for achieving a more universal representation with easy additions. Another research highlights recent advancements in English-to-Urdu machine translation, focusing on the use of Long Short-Term Memory (LSTM) neural networks. LSTM-based models are noted for handling Urdu's complex linguistic features, such as morphological richness and differing word order

from English. Preprocessing techniques, including tokenization, grammar analysis, and word embeddings, have been employed to improve translation accuracy. The model demonstrates strong performance, with BLEU scores of 50.86 (training) and 47.06 (test), and human validation shows high translation quality. This supports the effectiveness of LSTM-based neural machine translation for structurally different languages like English and Urdu [31]. The research aims to improve Roman Urdu to Urdu transliteration using machine learning models like RNN+LSTM, Seq2Seq, and Transformer. A dataset of 6.5 million Roman Urdu sentences was used for training. The Transformer model outperformed others, achieving a BLEU score of 75 due to its ability to handle long sentences and rare words. An Android app was also developed for transliteration. The study concludes that the Transformer model is the most effective, with plans for a web application and expanded datasets [32]. The research tackles Roman Urdu spelling variations using machine learning. A dataset of 5,244 Roman Urdu words with up to five spelling variations was collected from social platforms. Six ML classifiers—SVM, LR, DT, NB, KNN, and RF—were tested, with the SVM model achieving the highest accuracy of 99.96%. The study concludes that the SVM classifier is most effective for handling spelling variations and that the dataset will aid future natural language processing research [33]. The research on Romanian to Urdu transliteration provides an understanding of the methodologies of the use of Machine Learning models including RNN+LSTM, Seq2Seq, and Transformer. The Seq2Seq models at first hit half of 48 % on the BLEU scale but failed on long sentences and words that are rarely used. However, the Transformer model with the help of attention mechanisms gave about 80% of the BLEU score and was progenitive in handling the complexity as well as compounding words [34]. Investigate the transliteration of Romanized Assamese social media text, highlighting the challenges due to the lack of a standardized romanization system. They develop three models—PBSMT, BiLSTM seq2seq with attention, and a transformer model for character-level transliteration. Among the models, the BiLSTM seq2seq with attention outperforms the others in accuracy [35]. Ranathunga et al. provide a comprehensive survey of neural machine translation (NMT) techniques for low-resource languages (LRLs), addressing the challenges posed by the lack of large parallel corpora. The authors review advancements in NMT for LRLs, offering a quantitative analysis of the most widely used methods. They also present guidelines for selecting the optimal NMT approach based on specific low-resource data settings [36].

Table 1. Recent Research

Translation	Technique	Results %
Roman Urdu to Urdu	Rule-based character substitution And Unicode based character mapping techniques	95.8%
Roman-Urdu To Urdu Transliteration	Deep neural network-based encoder-decoder	70 BLEU score
Roman-Urdu and Urdu Parallel Corpus	Roman-Urdu- Parl, with 6.37 million sentence-pairs	BLEU score of 84.67
Translated Fictional Texts	Digitizing, transcribing And aligning translations of this text	10 corpus result out of 20 collected translations
English to Urdu Translation	Neural network- based deep learning technique	45.83 BLEU score

Roman-Urdu to Urdu Script	Algorithm convert the encoding roman words into the Urdu words	91.2%
Low-Resource Machine Transliteration	Neural networks—encoder decoder	60 BLEU score
STM Neural Machine Translation	LSTM encoder-decoder	Training: 50.86, Test: 47.06
Seq2Seq Sequence-to-sequence model	Transformer Model Attention-based deep learning model	75 BLEU
Spelling Variation of Roman Urdu	ML models	accuracy of 99.96

PROPOSED METHODOLOGY

As Roman Urdu is not a standardized language, it lacks basic grammar and written vocabulary norms. The suggested strategy in this work is to establish a transliteration model for Roman Urdu, which is a unique technique that provides a good standard for Roman Urdu. This section comprises the proposed techniques and strategies for achieving the research objectives.

3.1 Main Frame

Data pre-processing, rules-based made-up character substitution, and a Unicode-based fictional character map are the three components of the proposed technique, as shown. After starting the translator, the user can utilize a simple interface with one input frame an output frame, and a conversion key. When a consumer types a Roman Urdu text into the key box, the 1st component preprocessing strips out the superfluous information. The preprocessed Roman Urdu text then moves on to the next element. The Roman-Urdu content has been completely converted into Urdu text, which can subsequently be used by the user. The user may easily comprehend the precise definition of RU text and converse more expressively with other Roman Urdu users by utilizing the provided translator.

3.2 Data Preprocessing

The Roman Urdu text is inherently unprocessed and therefore requires preprocessing. One of the data mining strategies is to re-shape raw data using data preparation technologies. Translation models can effectively understand from preprocessed input data. Real-world Roman-Urdu data is likely to have numerous inaccuracies since it is partial, inconsistent, or absent behaviors or patterns. Preprocessing data is a well-known method for fixing such issues. In the real world, data that lacks counts of elements contains errors and aberrations, or just summarized data, is considered incomplete. During preprocessing, terms, sentences, or even complete sentences can be used as tokens. Tokenization is a concept that refers to the process of breaking down documents or phrases into individual terms to filter out non-essential keywords and punctuation. Second, in the situation of complex systems, large and lower-issue letters are treated as distinct words, therefore converting capital letters to lowercase reduces the number of unique terms in documents. This improves the feature extraction process' efficiency. 3rd, preprocessing is the method of converting data into something that a computer can realize.

3.3 Model Architecture

Transfer learning has grown in popularity as a powerful strategy in natural language processing, in which a computer is first pre-trained on a data-intensive task before being perfectly alright for a lesser goal (NLP). Because transfer learning success with a wide range of techniques, methodologies, and practices have emerged, we introduce a single framework that translates problems in every language into a text-text format in this research, which explores the

landscape of transfer learning approaches for NLP. On hundreds of language understanding tasks, our systematic analysis examines pre-training objectives, architectures, unlabeled datasets, move methodologies, and more parameters.

Over a hundred languages have been pre-trained into the mT5 model. Let's look at how we might use this to train a bilingual translation model for a language with few resources, such as Roman Urdu and Urdu. The multilingual Trans- former model mT5 was pre-trained on the mC4 dataset, which comprises text in 101 languages. The mT5 model's architecture (based on T5) is meant to accommodate any NLP task by recasting it as a sequence-to-sequence task. To put it another way, text enters and text exits. In a classification, for example, the text sequences to be classified can be the model's input, and the model's output will be the sequence's class label. This becomes even more straightforward in terms of translation. The input text seems to be in one language, as well as the output text is in a different language. Let's explore how we may fine-tune a mT5 model for machine translation, taking into account the multilingual capabilities of mT5 and the applicability of the sequential format for language translation. We'll be developing a translation model to convert between Roman Urdu and Urdu in this article. Because of the scarcity of resources, training excellent translation models for low-resource languages like Urdu is fairly difficult. Hopefully, the mT5 model will be able to compensate for the lack of training data in the form of straight Roman-Urdu to-Urdu (and vice versa) sequences thanks to the multilingual pre-training on a large dataset. To train the mT5 model, we'll use the Simple Transformers library (based on the Hugging face Transformers library).

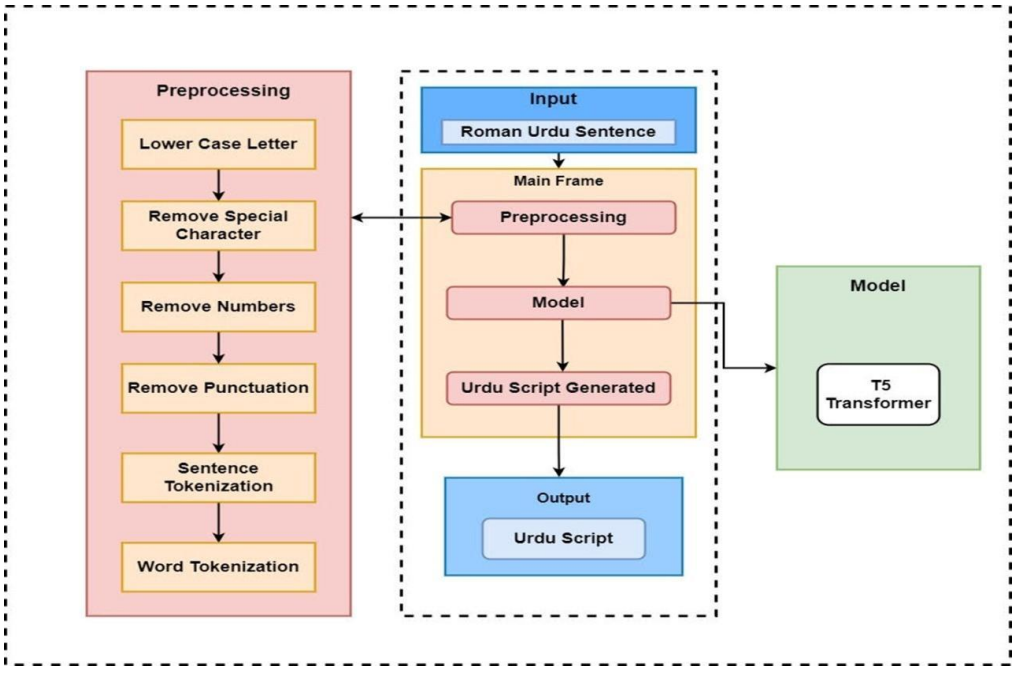

Figure 1. Proposed Architecture

Weights & Biases, which is natively supported in Simple Transformers for trial tracking and hyperparameter optimization, is used to create graphs and charts.

- T5 is a simple encoder-decoder model that has been pre-trained on a variety of supervised and unsupervised jobs, with each task transformed to text-to-text. T5 performs very well with a range of tasks right out of the box by prepending a distinct prefix to each task's input.
- T5 employs relative linear embeddings. Both the left and right sides of the encoder input can be padded.

T5 is an encoder and decoder paradigm that translates entire natural language processing difficulties into text-text format. The goal series is shifted to the right and sent into the decoder through to the decoder line, along with a start-sequence token. The EOS token then adds the target sequence, which is linked to the tags in the teacher-forcing method. The PAD token will be used to start the sequence. T5 may be fine-tuned in both supervised and unsupervised settings.

3.4 Data Set

Finally, a neat and associated Roman Urdu to A comparable corpus of 1,107,156 lines in Urdu was generated. As indicated in Table 1, our dataset comprises a whole Urdu dictionary there are 34,523 essential words and a total of 21,021 terms in the Roman-Urdu lexicon. It's worth noting that the Urdu dictionary contains more words than the Roman Urdu vocabulary. Because many words in Urdu have a gap among them but are still counted as one word. In Roman-Urdu, any such compound term is normally expressed as a single concept Com- pounding is a nice example like, and have their parallel "Islamabad," "bewakuf," and "ilmoadab," respectively, are Roman-Urdu transliterations of a single word.

Table 2. Details of the Parallel Corpus, Roman

Roman Urdu to Urdu Corpus	1,107,156
Total Roman Urdu Words	21,021
Total Urdu Words	34,523

The dataset was casually partitioned interested in three sets: train, progress, and test, with ratios of 70%, 15%, and 15%, respectively. We translated our parallel corpus into its indexed version for use in our sequence-to-sequence models. Every distinct word was assigned a numerical value. We used an indexed form of Roman-Urdu words as a response and received an indexed form of Urdu words transformed end to the initial Urdu script as a result.

Table 3. Steps to Transformation

Step 1	Our Input	Sara aur Zara dost hain
Step 2	Indexed Roman- Urdu	21 52 1 664 3200
Step 3	Indexed Urdu	451 562 2343 44
Step 4	Converted Output	سارہ اور زارا دوست ہیں۔

To get beyond the one-to-one correspondence constraint based on alignment, we are additionally considering irregular-size sentences in the Roman Urdu and Urdu parallel corpora. Text messaging or tweets in Urdu or Roman-Urdu shorthand, on the other hand, are not taken into account. Because the sequence-to-sequence model can learn dependencies on its own, we didn't apply any word insertion methods to translate each word to its vector form.

RESULTS

We tested the above-mentioned models using parallel corpora of Roman-Urdu to Urdu. The corpus consists of approximately 1.1 million phrases that were generated using a combination of automated and human processes. Our Roman Urdu vocabulary totals 21K words, whereas our Urdu vocabulary totals 35K. We test on the mT5 model, and so this model is pre-trained with just over a hundred distinct languages. Let's look at how we might use this to train a bilingual translation model for a language with few resources, such as Roman Urdu and Urdu. mT5 is a multilingual Transformer model that has been pre-trained on a dataset (mC4) that contains text in 101 languages. The mT5 model's architecture (based on T5) is meant to accommodate any NLP task by rephrasing the task as a sequence-to-sequence task.

4.1 Quantitative Analysis

Although the exact loss values don't tell us much, the fact that they will be falling means the model is learning, as illustrated in Figures 2 and 3. In

reality, the evaluation loss appears to be decreasing, indicating that the model has not yet converged. It's possible that training for a further epoch or two will improve the model's performance.

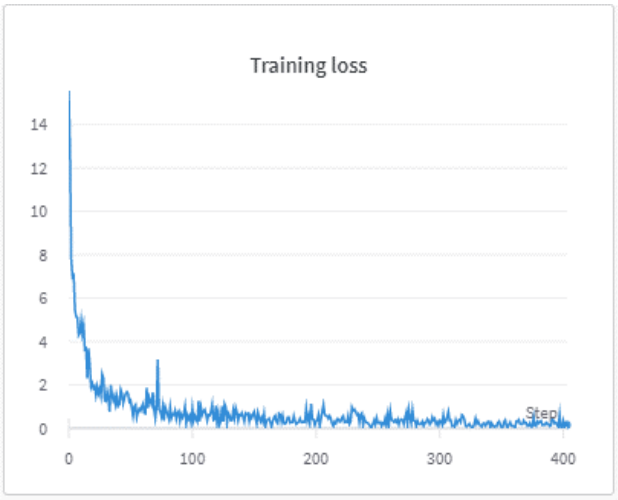

Figure 2. Training Loss

BLEU (BiLingual Evaluation Understudy) is a measure for assessing machine-translated text automatically. The BLEU score is a value between 0 and 1 which indicates how closely the machine-translated text resembles a collection of high-quality reference translations. The BLEU statistic is used to compare the output of SMT to that of human reference translations. It's vital to remember that SMT and human translations might differ greatly in terms of word usage, word order, and phrase length. To address these issues, BLEU tries to match variable-length phrases between SMT output and reference translations. The translation score is calculated using weighted match averages.

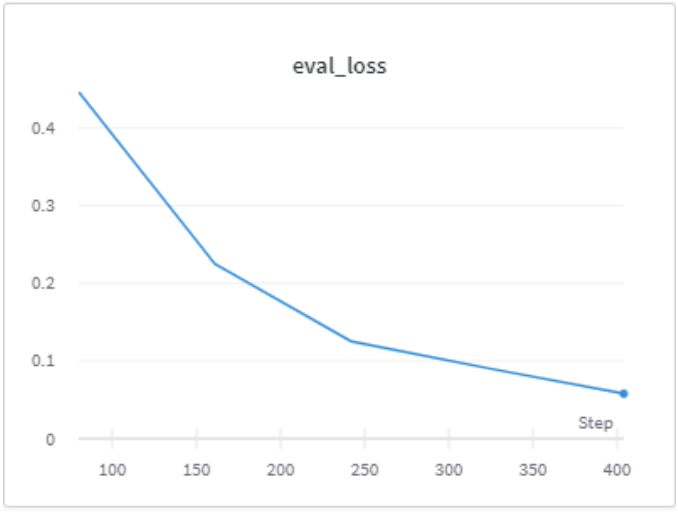

Figure 3. Evaluation Loss

The BLEU metric comes in a variety of forms. The fundamental metric, on the other hand, necessitates the computation of a shortness penalty P!

$$P_B = \begin{cases} 1, & c > r \\ e^{(1-r)}, & c <= r \end{cases} \quad (1)$$

where r is the length of the reference corpus and c is the length of the candidate (reference) translation. The Basic BLEU metric is then determined as shown

$$BLEU = P_B exp() \sum_{N=0}^{N} w_n log P_n \qquad (2)$$

Where Wn are positive weights summing to one, and the n-gram precision Pn is calculated using n-grams with a maximum length of N. The BLEU score, specifically the BLEU system used by the annual Conference on Machine Translation, is the standard statistic for evaluating and comparing machine translation models (WMT). This score may be calculated using the Sacre BLEU library, and we achieved a good result of 91.56, as shown in Table 4.

Table 4. Details of the Bleu Score after Testing

Translation	Bleu Score
Roman-Urdu To Urdu	91.56

4.2 Qualitative Analysis

Although our models achieved a commendable BLEU score, their qualitative performance is even more impressive. Table 5 and Figure 4 illustrate these results, with the model demonstrating robust capabilities in transliteration tasks. As shown in Table 5, when provided with Roman Urdu input, the model successfully and accurately converts it into Urdu. This highlights the model's ability to handle the complexities of Roman Urdu and its transliteration to standard Urdu, as seen in various examples. For instance, the model correctly transliterates simple words such as 'wakeel' to 'وکیل' and more complex sentences with nuanced meanings, as evidenced by Sentence 5, which involves multiple clauses and detailed terminology.

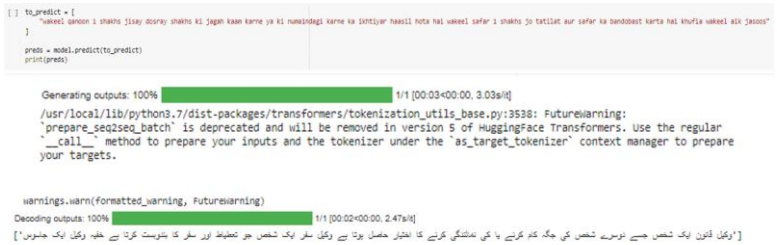

Figure 4. Output of Prediction

Table 5. Table of Output Prediction

Sentence No:	Input Sentence	Output Sentence
1	wakeel	وکیل
2	wakeel qanoon	وکیل قانون
3	woh mein wazeer taleem bhi reh chuke hain	وہ میں وزیر تعلیم بھی رہ چکے ہیں
4	Pakistan mein dehshat gardi se morad poooray malik mein majmoi dehshat grdanh karwaiyan hain	پاکستان میں دہشت گردی سے مراد پورے ملک میں مجموعی دہشت گردانہ کاروائیاں ہیں
5	wakeel qanoon 1 shakhs jisay dosray shakhs ki jagah kaam karne ya ki numaindagi karne ka ikhtiyar haasil hota hai wakeel safar 1 shakhs jo tatilat aur safar ka bandobast karta hai khufia wakeel aik jasoos	وکیل قانون ایک شخص جسے دوسرے شخص کی جگہ کام کرنے یا کی نمائندگی کرنے کا اختیار حاصل ہوتا ہے وکیل سفر ایک شخص جو تعطیلات اور سفر کا بندوبست کرتا ہے خفیہ وکیل ایک جاسوس

Conclusion

The purpose of this study is to design a translator for Roman-Urdu to Urdu Transliteration. As discussed earlier, there are already quite a few translators for the above specific languages but there is no specific translator for Roman-Urdu to Urdu for long sentences and give good accuracy. Previous translators use the RNN Technique which gives a 48.6 BLEU score and gives an output of length consisting of 10 [7] shown in Figure: 6 similarly to overcome the problem of short sentences we use the RNN updated version model which is LSTM which give 70 BLEU score [2] shown in the Figure: 6 but it takes so much time to train and might be an issue with accuracy. This study, therefore proposed to encode decode the T5 Transformer, which is based on transfer learning the transformer we are using is multilingual and pre-trained in 101 languages due to this it improves the accuracy, and time cost as we showed after the training of our model we got a remarkable score of 91.56 BLEU score shown in figure:6 and each task, involving translations, info extraction, and categorization, is characterized T5 may be used to give the model text as input and train it to create some goal text, a transformer-based framework. After the implementation of this transformer, our results are more accurate than all other compared transformers.

These findings imply that future advances in the scale and quality of pre-trained text-to-text models might lead to even more advantages for sentence encoder models. Moreover, we will perform more variations of the Roman Urdu language in our model. And improve our translation model on different variations with high results on long paragraphs. Furthermore, we will perform an in-depth study on the essence of Urdu translation so that people can use Roman Urdu to Urdu translation easily in daily life.

References

[1] M. Shahroz, M. F. Mushtaq, A. Mehmood, S. Ullah, and G. S. Choi, "RUTUT: Roman Urdu to Urdu translator based on character substitution rules and Unicode mapping," *IEEE Access*, vol. 8, pp. 189823–189841, Oct. 2020, doi: https://doi.org/10.1109/ACCESS.2020.3031393.

[2] M. Alam and S. ul Hussain, "Deep learning-based Roman-Urdu to Urdu transliteration," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 35, no. 4, 2021, Art. no. 2152001, doi: https://doi.org/10.1142/S0218001421520017.

[3] M. Alam and S. ul Hussain, "Roman-Urdu-Parl: Roman-Urdu and Urdu parallel corpus for Urdu language understanding," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 21, no. 1, Jan. 2022, Art. no. 13, doi: https://doi.org/10.1145/3464424.

[4] A. Fraisse, R. Jenn, and S. F. Fishkin, "Building multilingual parallel corpora for under-resourced languages using translated fictional texts," in *Proc. 3rd Workshop Collab. Comput. Under-Resour. Lang.*, Miyazaki, Japan, Oct. 2018.

[5] A. Kunchukuttan, P. Mehta, and P. Bhattacharyya, "The IIT Bombay English–Hindi parallel corpus," *arXiv preprint*, arXiv:1710.02855, 2017.

[6] N. T. Le *et al.*, "Low-resource machine transliteration using recurrent neural networks," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 18, no. 2, Mar. 2019, Art. no. 13, doi: https://doi.org/10.1145/3265752.

[7] M. Alam and S. Hussain, "Sequence-to-sequence networks for Roman-Urdu to Urdu transliteration," in *Proc. Int. Multi-Topic Conf. (INMIC)*, Lahore, Pakistan, Dec. 2017, pp. 1–6.

[8] A. Daud, W. Khan, and D. Che, "Urdu language processing: A survey," *Artif. Intell. Rev.*, vol. 47, pp. 279–311, Mar. 2017, doi: https://doi.org/10.1007/s10462-016-9486-x.

[9] S. A. B. Andrabi and A. Wahid, "Machine translation system using deep learning for English to Urdu (retracted)," *Comput.*

Intell. Neurosci., vol. 2022, 2022, Art. no. 7873012, doi: https://doi.org/10.1155/2022/7873012.

[10] H. M. Shakeel, R. Khan, and M. Waheed, "Context-based Roman-Urdu to Urdu script transliteration system," *arXiv preprint*, arXiv:2109.14197, 2021.

[11] M. H. Al-Khresheh and S. A. Almaaytah, "English proverbs into Arabic through machine translation," *Int. J. Appl. Linguist. English Lit.*, vol. 7, no. 5, pp. 158–166, Sep. 2018.

[12] S. Khan *et al*., "Translation divergence patterns handling in English to Urdu machine translation," *Int. J. Artif. Intell. Tools*, vol. 27, no. 5, Oct. 2018, Art. no. 1850017, doi: https://doi.org/10.1142/S0218213018500173.

[13] A. Bilal *et al*., "Roman-txt: Forms and functions of Roman Urdu texting," in *Proc. Int. Conf. Hum.-Comput. Interact. Mobile Devices Serv.*, Sep. 2017, doi: https://doi.org/10.1145/3098279.3098552.

[14] N. Durrani *et al*., "Hindi-to-Urdu machine translation through transliteration," in *Proc. 48th Annu. Meet. ACL*, Uppsala, Sweden, Jul. 2010, pp. 465–474.

[15] H. Masroor *et al*., "Transtech: Development of a novel translator for Roman Urdu to English," *Heliyon*, vol. 5, no. 5, May 2019, Art. no. 01780, doi: https://doi.org/10.1016/j.heliyon.2019.e01780.

[16] S. K. Mahata, D. Das, and S. Bandyopadhyay, "MTIL2017: Machine translation using recurrent neural networks on statistical MT," *J. Intell. Syst.*, vol. 28, no. 3, pp. 447–453, Jul. 2019, doi: https://doi.org/10.1515/jisys-2018-0016.

[17] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proc. Conf. Emp. Methods Nat. Lang. Proc.*, Oct. 2013, pp. 1700–1709.

[18] N. Durrani and P. Koehn, "Improving machine translation via triangulation and transliteration," in *Proc. Annu. Conf. Eur. Assoc. Mach. Transl.*, Jun. 2014.

[19] M. Zafar and A. Masood, "Interactive English to Urdu machine translation using example-based approach," *Int. J. Comput. Sci. Eng.*, vol. 1, no. 3, pp. 275–282, 2009.

[20] J. Ni *et al*., "Sentence-T5: Scalable sentence encoders from pre-trained text-to-text models," *arXiv preprint*, arXiv:2108.08877, 2021.

[21] M. A. Kumar *et al*., "An overview of the shared task on machine translation in Indian languages (MTIL-2017)," *J. Intell. Syst.*, vol. 28, no. 3, pp. 455–464, Jul. 2019, doi: https://doi.org/10.1515/jisys-2018-0024.

[22] D. Lamba and W. H. Hsu, "Answer-agnostic question generation in privacy policy domain," in *Proc. Int. Conf. Electron., Commun. Inf. Technol. (CECIT)*, Dec. 2021, pp. 1–6.

[23] R. Dabre, C. Chu, and A. Kunchukuttan, "A survey of multilingual neural machine translation," *ACM Comput. Surv.*, vol. 53, no. 5, pp. 1–38, Oct. 2020, doi: https://doi.org/10.1145/3406095.

[24] Z. A. Zeeshan and M. Z. Jawad, "Chinese–Urdu machine translation based on deep learning," *J. Auton. Intell.*, vol. 3, no. 2, pp. 34–44, 2020.

[25] A. Mastropaolo *et al*., "Studying the usage of text-to-text transfer transformer to support code-related tasks," in *Proc. IEEE/ACM Int. Conf. Softw. Eng. (ICSE)*, May 2021.

[26] J. J. Bird, A. Ekárt, and D. R. Faria, "Chatbot interaction with artificial intelligence using T5," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 4, pp. 3129–3144, Apr. 2023, doi: https://doi.org/10.1007/s12652-021-03439-8.

[27] C. Raffel *et al*., "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.
[28] C. R. Dhivyaa *et al*., "Transliteration-based GPT-2 model for Tamil text summarization," in *Proc. Int. Conf. Comput. Commun. Info.*, Jan. 2022.
[29] I. Ganguli *et al*., "Empirical auto-evaluation of Python code using T5 architecture," in *Proc. Int. Conf. Smart Comput. Commun. (ICSCC)*, Jul. 2021.
[30] L. Xue *et al*., "mT5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint*, arXiv:2010.11934, 2020.
[31] S. H. Kumhar *et al*., "Translation of English into Urdu using LSTM model," *Comput., Mater. Contin.*, vol. 74, no. 2, pp. 3899–3912, 2023.
[32] A. Ahmad and M. A. Ahmad, "Advancing Roman Urdu to Urdu transliteration using machine learning techniques," *Asian J. Multidiscip. Res. Rev.*, vol. 5, no. 2, pp. 108–127, Apr. 2024.
[33] M. A. Soomro *et al*., "Spelling variation of Roman Urdu using machine learning," *J. Comput. Biomed. Informatics*, vol. 7, no. 2, 2024.
[34] J.-H. Ju, J.-H. Yang, and C.-J. Wang, "Text-to-text multi-view learning for passage re-ranking," in *Proc. ACM SIGIR Int. Conf. Res. Dev. Inf. Retrieval*, Jul. 2021.
[35] H. Baruah, S. R. Singh, and P. Sarmah, "Transliteration characteristics in Romanized Assamese social media text," *ACM Trans. Asian Low-Resour. Lang. Inf. Proc.*, vol. 23, no. 2, Art. no. 33, Feb. 2024, doi: https://doi.org/10.1145/3639565.
[36] S. Ranathunga *et al*., "Neural machine translation for low-resource languages: A survey," *ACM Comput. Surv.*, vol. 55, no. 11, Art. no. 29, Nov. 2023, doi: https://doi.org/10.1145/3567592.