RESEARCH ARTICLE

Enhancing Sentiment Analysis of Roman Urdu Using Augmentation Techniques and Deep Learning Models

Muhammad Owais Khan[1], **Wahab Khan**[1,*], **Yanan Wang**[2], **Aziz Ur Rehman**[3] **and Muhammad Alamzeb Khan**[1]

[1] Department of Computer Science, University of Science and Technology, Bannu, Khyber Pakhtunkhwa, Pakistan
[2] Department of Computer Science and Engineering, Sejong University, Seoul 05006, Republic of Korea
[3] Department of Computer Science, Islamia College University, Peshawar, Khyber Pakhtunkhwa, Pakistan

Abstract

Roman Urdu sentiment analysis faces significant challenges due to transliteration inconsistencies, informal language usage, and the lack of labeled datasets. This study proposes a novel framework that addresses these challenges by combining advanced data preprocessing techniques and data augmentation strategies such as synonym replacement, back-translation, and random word insertion. These methods enhance dataset diversity, improving the model's generalization ability. A rich Roman Urdu dataset was collected from diverse sources, including social media platforms (Facebook, Twitter, YouTube), blogs, forums, and e-commerce sites, to capture a wide range of user opinions. Three deep learning models, Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM), were evaluated for sentiment classification. The results show that the LSTM model outperforms the others with an accuracy of 94%, compared to 90% for RNN and 92% for GRU. The LSTM's ability to capture long-term dependencies and contextual nuances in Roman Urdu text makes it the most effective model for this task, demonstrating a significant improvement over the traditional method.

Keywords: Roman Urdu, sentiment analysis, deep learning, data augmentation, text classification, GRU, LSTM, RNN.

1 Introduction

Sentiment analysis is an important task under Natural Language Processing (NLP), which deals with emotions in text. This automated process is useful for many tasks, including market research, social media monitoring, and customer feedback analysis [1]. It is crucial in serving users with customized options in e-commerce [2] as well as in the monitoring of public opinion for decision-making processes in governance. Much work has been done for the more formalized languages like English; however, there is a great deal of difficulty working with casual, informal, or even not fully non-standardized scripts such as Roman Urdu,

Submitted: 29 May 2025
Revised: 07 July 2025
Accepted: 09 July 2025
Published: 14 July 2025

Vol. 1, **No.** 3, 2025.
10.62762/TACS.2025.190575

***Corresponding author:**
✉ Wahab Khan
wahabshri@gmail.com

Citation
Khan, M. O., Khan, W., Wang, Y., Rehman, A. U, & Khan, M. A. (2025). Enhancing Sentiment Analysis of Roman Urdu Using Augmentation Techniques and Deep Learning Models. *ICCK Transactions on Advanced Computing and Systems*, 1(3), 164–179.

© 2025 by the Authors. Published by Institute of Central Computation and Knowledge. This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

particularly due to its varied transliterations and lexical inconsistencies [3]. Roman Urdu, a fusion of Roman letters and the Urdu language, is mainly employed in South Asia, particularly in Pakistan. Its prevalent appearance on social media, forums, and messaging applications is a result of the increased convenience of typing on electronic gadgets, a phenomenon that has simultaneously introduced new challenges such as cyberbullying in Roman Urdu digital spaces [4]. While machine learning methods have demonstrated strong performance on Urdu language text classification tasks [5], Roman Urdu presents additional challenges owing to its informal nature and lack of standardized orthography, yet it remains indispensable for users who find Roman letters easier to communicate with, thus helping to mitigate gaps in digital communication [6]. Nevertheless, sentiment analysis of Roman Urdu remains highly problematic. Lexicon-based approaches fail because of informal spelling, slang, and grammar deficits [7]. Likewise, the absence of monocodal standardization and many ways of writing the same word to be processed as 'khoobsurat', 'khobsurat', or 'khoobsorat' aggravate the problem of textual data processing [8]. The lack of annotated datasets forces the adoption of new paradigms for Roman Urdu sentiment analysis [9].

As an informal dialect of Urdu, Roman Urdu is used by millions of speakers across the world, especially on various media platforms. As users move towards easier and faster methods of communication, this form of transliteration has become dominant on social media platforms, messaging applications, and discussion forums. Roman Urdu is helpful for users who do not have access to Urdu keyboards because it breaks the language barriers, thus enabling effortless interaction in the digital spaces [10]. It further allows the Urdu-speaking populace to actively participate in the global digital environment, and the growing body of Roman Urdu text on social media has motivated researchers to develop deep learning-based sentiment analysis models tailored to this language [11]. Comprehensive reviews of Urdu and Roman Urdu sentiment analysis highlight that this field spans multiple languages, scripts, and methodological approaches [12]. Aside from that, Roman Urdu also assists those with limited abilities to read or write in the traditional Urdu script, and the increasing use of such informal text online has spurred research into automated detection of offensive and abusive language in Urdu-based digital content [13]. This change makes it easier for people to take part fully in the digital world, though it simultaneously creates new challenges such as cyberbullying and abusive text in Roman Urdu, necessitating dedicated computational linguistic resources for detection and moderation [14].

Even though scholarly articles dealing with Roman Urdu are scarce, it is commonplace in daily life, which presents unique challenges for NLP activities, including but not limited to sentiment analysis. The primary challenge is the informal nature of Roman Urdu, which is filled with slang, abbreviations, and proper colloquial terms. This form of informal language poses problems for traditional sentiment analysis tools [15]. One of the persistent issues that comes with Roman Urdu is a lack of standardization. There are no set ortho-graphic rules for Roman Urdu, which leads to numerous ways of transliteration [16]. A case in point is the term beautiful, which has a multitude of variations appealing in the tokenization and feature extraction phases [17]. Roman Urdu actively promotes code-mixing where the speakers interweave Urdu and English in one sentence [18].

Advances in Roman Urdu sentiment analysis pose great hurdles to NLP, which enhances its prospects for research. The absence of tools and resources for Roman Urdu is a part of a bigger problem regarding unstandardized and low-resource languages; deep learning approaches have been proposed to address such resource constraints in closely related languages like Urdu [19]. Furthermore, enhancing sentiment analysis for Roman Urdu is also profound at cultural and social levels, as it allows for greater diversity and representation within NLP. Such tools would serve the sentiment analysis of Roman Urdu communities, which many businesses, politicians, and social scientists would find useful [20]. In addition to this, these tools would help combat abuse in Urdu-based digital content through sentiment analysis and content filtering [21].

The traditional sentiment analysis has limitations with predefined lexicons and features due to their applicability to standard languages only. Such is the case for Roman Urdu, which is complex, informal, and slang-infested. The Roman Urdu language lacks semantic features because they range from informal to regional and even to colloquial slang, which is not acceptable in modern-day traditional models. For instance, words termed as 'mazaydar' (which stands for tasty) or 'chill' (which is slang for relaxed), do not get accepted in sentiment lexicons,

making things complex as pointed out by Jawad et al. [22] in 2024. Overcoming it using deep learning alongside data augmentation seems relevant. Deep learning models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Transformer-based models including BERT are capable of learning complex patterns from text data. These models are particularly effective at capturing the subtleties of Roman Urdu, including its informal expressions and transliteration variations.

The development of deep learning systems teaches computers to understand human language and communication with minimal or no human input at all. This is especially beneficial in the field of sentiment analysis because it requires local dependencies and sequential order to be captured effectively [23]. CNNs are useful for short phrases, while RNNs, particularly LSTMs and GRUs, are context-sensitive and work well with longer sequences [16]. With the algorithm changes in GPT and BERT, transformer models have completely changed the field of deep learning. The knowledge embedded in the algorithms allows extensive sentences to be processed based on different complex patterns in the text data [24]. Combining these augmenting techniques in deep learning systems constructs powerful models for sentiment analysis in Roman Urdu.

As for the Roman Urdu language, a tiered language of NLP that lacks abundant resources, it poses a challenge during data augmentation. Data augmentation includes creating new training examples using transformations of the existing data. Models can generalize and adapt to real-world text better with diverse data and more training examples. In the specific case of sentiment analysis for the Roman Urdu language, data augmentation plays a crucial role. This is due to a lack of labeled datasets.

Key Data Augmentation Techniques

- **Synonym Replacement:** Substitutes terms in a clause with synonymous expressions to broaden the vocabulary while keeping the meaning intact. In scope, 'Woh bohot acha hain' (He is very good) can morph into 'Woh kafi acha hain' (He is quite good), applying enhanced vocabulary diversification [25].
- **Back Translation:** Includes the processes of translating a text into different phrases and languages (English), and bringing it back to its original language (Roman Urdu) with intended stylistic and meaning changes. An example would be "mujhe yeh pasand hai" which can be translated and re-translated as "I like this" and "mujhe yeh acha lagta hai". This allows for the corpus under consideration to be enhanced with more phrases [26].
- **Random Insertion and Deletion:** Inserts or extracts information from a sentence to model noisy or incomplete data. As an example, a phrase such as 'Woh bohot pyara hai' (He is very lovely) can be transformed through these processes into 'Woh bohot zyada pyara hai' (He is very much lovely) for insertion, or 'Woh pyara hai' (He is lovely) for deletion.
- **Text Paraphrasing:** Rewrites the sentences but keeps the meaning intact, for example, 'Yeh kitab achi hai' can be paraphrased into 'Yeh kitab bohot achi lagti hai', which helps models understand several sentence structures in depth.

These methodologies for data augmentation allow models to address informal spellings, slang, and the numerous ways of transliterating Roman Urdu's particular traits. Such techniques strengthen the training corpus, refining the accuracy, robustness, and versatility of the model. Data augmentation also makes the model less biased by addressing the lack of data for NLP tools tailored for Roman Urdu speakers in a more culturally sensitive manner. Few comprehensive linguistic resources exist for Roman Urdu, resulting in its neglect in sentiment analysis. This study seeks to fill that gap by constructing strategies tailored to Roman Urdu. Analysis of sentiment in Roman Urdu further assists in understanding the sentiments and views of people belonging to the Urdu-speaking community, which is beneficial to business strategists, policy makers, and sociologists. This also further improves online safety through content moderation as well as increases user interaction by improving communication tools. Emphasis on a singular dialect and standardization has been a major roadblock in Roman Urdu, as well as informal expressions, colloquialisms, and transliteration. This gap reduces the effectiveness of traditional NLP models. The sentiment classification is hindered by the informal nature of the language and the lack of labeled datasets.

The primary contribution of this research study is:

- To develop a deep learning-based sentiment analysis framework specifically for Roman Urdu.
- To propose a novel framework for sentiment

analysis of Roman Urdu that integrates advanced data augmentation techniques, such as synonym replacement, back-translation, and random word insertion. Unlike prior work, which mainly focuses on traditional methods, our approach leverages deep learning models (RNN, GRU, LSTM) to handle the complexities of Roman Urdu, including informal language usage, transliteration inconsistencies, and the lack of labeled data.

- To incorporate data augmentation techniques such as synonym replacement, back translation, and random swaps to enhance dataset diversity and model generalization.
- To evaluate and compare the performance of different deep learning architectures, including RNN, GRU, and LSTM models, for Roman Urdu sentiment analysis.
- To achieve significant improvements in accuracy, precision, recall, and F1-score metrics for Roman Urdu sentiment classification.

The rest of the paper is structured as follows: In section 2, we analyze literature on sentiment analysis, concentrating on the problems and uses of deep learning for low-resource languages such as Roman Urdu, and also review the applications of data augmentation techniques. In section 3, we describe our results, combining the strategies of collecting and preprocessing the data with building the sentiment analysis system for Roman Urdu using deep learning frameworks. In section 4, we assess the results of the study by analyzing the performance of various architectures (RNN, GRU, LSTM) and paying particular attention to the effects of data augmentation methods. At last, in section 5, we provide our reflections on the study, its findings, and limitations, and offer recommendations for other researchers.

2 Related Work

This analysis evaluates prior work on machine learning and deep learning techniques for sentiment analysis within low-resource languages such as Roman Urdu. The main objective is to identify gaps, accomplishments, and possible intervention strategies towards developing efficient sentiment analysis systems for Roman Urdu. The study attempts to address the resource barriers using data augmentation techniques and purpose-built neural network designs to enhance sentiment analysis of lesser-known languages, thus promoting inclusivity and cultural sensitivity in Natural Language Processing. Sentiment analysis, a part of Natural Language Processing (NLP), sorts text data into classes: positive, negative, or neutral. The two earliest approaches, the rule-based and lexicon-based methods, worked well for formal languages but were inadequate for informal ones. There was no adequate handling of the variability of language, as well as casual expressions typical of social media postings. With the rise of machine learning, techniques such as Support Vector Machines and Naïve Bayes became well-known, as they had better results in classifying the sentiment of texts. Nevertheless, as Sehar et al. [28] state, these traditional approaches have been surpassed by deep learning techniques, which automate the processes of dependency context extraction from the source text. Comprehensive surveys on Urdu sentiment analysis further confirm that the field has progressively shifted from rule-based methods toward data-driven deep learning approaches [27].

The analysis of sentiment in Roman Urdu has been a developing domain under digital communication. Researchers like Khadim et al. [29] and Muhammad and Burney [8] have attempted to solve the problem using lexicon-based and machine learning approaches, but they struggled due to the informal nature of Roman Urdu dialect and its multitude of transliterations. These studies have been expanded by Ullah et al. [30] and Malik and Ghous [31], who incorporated deep learning models and a hybrid approach to sentiment classification. The culture-specific features and the variation particular to Roman Urdu text are captured by these models, which drive the automation of sentiment analysis concerning user-generated content forwarded by the users. Roman Urdu, as a target language, does not have the required ortho-graphic and linguistic resources to test mapping automation; hence, it is a highly challenging, low-resource language. Moreover, colloquialisms in the language and lack of a prescribed structure make NLP systems even more challenging. To tackle these issues, Qureshi et al. [32] proposed a natural language processing model for Roman Urdu review classification, demonstrating that machine learning classifiers can effectively identify sentiment polarity in informal Roman Urdu text. For Roman Urdu, transfer learning on multilingual pre-trained models like mBERT and XLM-R has been reported to yield good results for this purpose. Such models can be trained on available data on popular high-resource languages like English so that the problem of attempting to annotate a corpus in low low-resource variety is alleviated [33].

Table 1. Literature review comparison on roman urdu sentiment analysis.

Study	Task/Focus	Dataset	Model/Technique	Remarks
Khadim et al. [29]	Sentiment analysis of social media content in Roman Urdu using data mining techniques	Social media content	Lexicon-based methods	Achieved 75% accuracy; struggled with informal language and varied spelling.
Bilal et al. [10]	Context-aware deep learning model for detection of Roman Urdu hate speech on social media	Social media posts	Deep learning models	Precision of 82%, with improved context-awareness over traditional methods.
Malik and Ghous [31]	A hybrid machine learning model to predict sentiment analysis on Roman Urdu text	Roman Urdu reviews	Hybrid machine learning, SVMs	Hybrid approach gave competitive results with 88% accuracy and 90% F1 score.
Ullah et al. [30]	Hate and offensive text classification in Urdu using unified embeddings ensemble framework	Urdu social media text	Unified embeddings ensemble (UEF-HOCUrdu)	Achieved strong classification performance for hate and offensive content detection in Urdu using ensemble deep learning embeddings.
Ali et al. [34]	Sentiment analysis of low-resource language literature using data processing and deep learning	Low-resource language datasets	Deep learning	Improved accuracy by 15%, achieving 82% accuracy in low-resource sentiment analysis.
Nazir et al. [47]	Leveraging multilingual transformer for multiclass sentiment analysis in code-mixed data	Roman Urdu data	Multilingual transformers (mBERT, XLM-R)	Improved sentiment analysis with 89% accuracy, capturing multilingual nuances.
Qureshi et al. [32]	Sentiment analysis of natural language reviews in Roman Urdu	Roman Urdu song/product reviews	Machine learning classifiers	Achieved competitive accuracy in Roman Urdu review classification, demonstrating the viability of NLP-based sentiment models for informal text.
Ashraf et al. [33]	Revolutionizing Urdu sentiment analysis using XLM-R and GPT-2	Urdu datasets	XLM-R, GPT-2	Achieved 91% accuracy in sentiment classification with transformer models.
Li et al. [20]	Roman Urdu sentiment analysis using transfer learning	Roman Urdu data	Transfer learning	Demonstrated high performance with an F1 score of 0.87 and 86% accuracy.
Bello et al. [35]	BERT-based sentiment analysis for Roman Urdu	Roman Urdu datasets	BERT-based model	Overcame transliteration challenges, achieving 92% accuracy in Roman Urdu sentiment analysis.
Azam et al. [36]	Exploring data augmentation strategies for hate speech detection in Roman Urdu	Roman Urdu text	Data augmentation (back-translation)	Achieved 85% accuracy by using back-translation to improve model generalization.
Chandio et al. [37]	Sentiment analysis of Roman Urdu e-commerce product reviews using machine learning	Roman Urdu e-commerce reviews	Machine learning classifiers	Achieved competitive accuracy in Roman Urdu e-commerce review classification, demonstrating the applicability of ML models to informal commercial text.
Li et al. [48]	Data augmentation approaches in natural language processing: A survey	Various datasets	Data augmentation	Data augmentation led to 20–25% performance improvements in NLP tasks for low-resource languages.

As compared to basic machine learning that relies on hand-set features, deep learning is capable of deriving complex structures from unprocessed data. Various models of Recurrent Neural Networks (RNNs) and more advanced models like Long Short-Term Memory (LSTM) networks are proficient at recognizing sequential dependencies and context sensitivity. This proves important in the area of tasks in sentiment analysis [34]. The understanding, and thus processing, of entire sequences of text is further transformed by words placed in deep context with the use of these advanced models. Not only do these models do well with informal expressions, but they are also proficient in code switching, which is very common in Roman Urdu. Table 1 shows the summarized literature of existing studies on Roman Urdu sentiment analysis.

Convolutional Neural Networks (CNNs), on the other hand, are good at extracting local features as well as word n-grams and are thus suitable for short phrases or word-level sentiment analysis [35]. Data augmentation is a pivotal strategy to solve the problem of scarcity of data while increasing the robustness of models, such as in the case of low-resource languages like Roman Urdu. Methods such as synonym replacement, back translation, and random insertion have been shown to generate more training data. For example, synonym replacement expands the lexicon, while back translation assists in providing different syntactic forms of the data; hence, the model's flexibility to the variations becomes broader. These methods are known to improve generalization and, as Azam et al. [36] showed, sentiment analysis of

Roman Urdu is easier with the implementation of these techniques.

The study of sentiment analysis in Roman Urdu, like other languages, can be beneficial in business intelligence, social media analysis, and even measuring the public's perception. As noted by Chandio et al. [37], companies can gain important information regarding customers' dissatisfaction through sentiment analysis in Roman Urdu reviews in different online stores. Furthermore, social media sentiment analysis facilitates the monitoring of public reactions to various events, activities, and marketing efforts, which is very helpful for organizations. Also, sentiment analysis gives a good indication of people's sentiments toward policies and social issues, which allows for more effective policymaking as described by Xu et al. [38]. In addition, moderation of hate speech and disinformation on online platforms is another area where sentiment analysis is very useful, as noted by Malik and Ghous [31]. Automated detection and filtration of inappropriate content aid in creating a safe cyberspace.

Although sentiment analysis of Roman Urdu has been developed to a relatively larger extent, challenges such as informal language use, different conventions for transliteration, and the inclusion of English code-mixing make it difficult to implement it completely [38]. These issues restrict tokenization and feature extraction, commonly used in traditional Natural Language Processing (NLP). The bright side is that typing more complex deep learning methods with suitable data augmentation strategies is useful. Kaur and Saini [39] and Doddapaneni et al. [40] claim that for code-mixed Roman Urdu, the sentiment accuracy can be improved through fine-tuning of BERT-like transformers using respective datasets. Beyond transformers, LSTM-based models have also been applied to language identification and sentiment classification of English-Urdu code-mixed text, demonstrating competitive performance on informal multilingual input [41]. Although it is challenging, a combination of data augmentation, deep learning models, and transfer learning has been able to make significant strides in building efficient and powerful dynamic sentiment analysis systems for Roman Urdu [42].

Machine and deep learning technologies have significantly advanced sentiment analysis, particularly for data-rich languages. Nevertheless, pursuing Roman Urdu, a minority language with fewer data resources, remains an uphill task due to the absence of research data, variations in transliterations, and informal language character. Having overcome these obstacles has been possible through deep learning models, augmentation procedures, and transformer models. The recognition of Roman Urdu's relevance in event capturing of scientific analysis of public opinion monitoring, sociological research, and business intelligence for social media provides a robust basis for the importance of Roman Urdu in current-day NLP. Due to the lack of data and diversity of the language, this study aims to develop an automated system for sentiment analysis of Roman Urdu based on deep learning and augmentation approaches. Such projects need time because Roman Urdu has not been served yet in the field of NLP. Therefore, this interdisciplinary research will aim to address this gap.

3 Methodology

In this section, an approach is provided to build a complete sentiment analysis model for Roman Urdu text. The first step is to collect external datasets for the analysis, which include social media posts, product reviews, and other user-generated content. Next, the gathered data is preprocessed, which involves steps like text cleaning, normalization, and filling in missing values; such preprocessing steps are critical for Roman Urdu text, as demonstrated by Majeed et al. [43], whose deep learning pipeline for Roman Urdu emotion analysis relied on rigorous corpus cleaning and normalization to ensure data quality. To address the challenges of limited labeled data and transliteration variability, strategies such as augmenting the diversity of the dataset using synonym replacement, back translation, random insertion, and random swapping are employed. Sentiment classification, in this case, is done using Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) because these models are most capable of capturing the sequential and contextual dependencies in Roman Urdu text [44]. Figure 1 presents the proposed architecture of this research study. The dataset is divided into train and test sets, where the models are trained on binary cross-entropy loss using the Adam optimizer. Measures for validation and testing to counter overfitting, and evaluation is done through the accuracy, precision, recall, F1 score, and other metrics to assess the performance. The model that is evaluated to have the highest performance is selected for use in the prediction of sentiments in business analytics, social media tracking, content moderation, and other areas. The proposed approach

offers a systematic way to address the challenges of Roman Urdu sentiment classification.

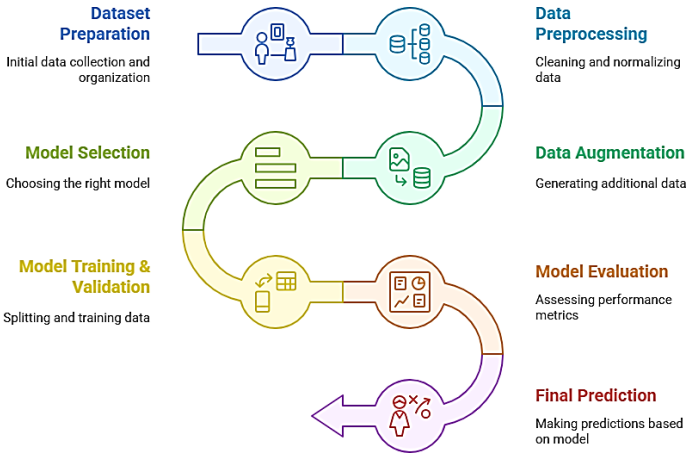


Figure 1. Proposed architecture.

3.1 Dataset Collection

For this study, we are collecting a rich dataset of Roman Urdu text available on social media like Facebook, Twitter, and YouTube, as well as on blogs, forums, and e-commerce sites. These sources are rich in sentiments as they talk about various opinions and experiences of users. The informal nature of the language, as well as the myriad ways of spelling words (ex, "acha" and "achha"), code-switching with English, the use of slang, and unstructured texts make it incredibly difficult to collect Roman Urdu data. To cope with this, we use web scraping and APIs to filter out duplicates and irrelevant data, and collect the data to manually or semi-automatically sort it into positive, negative, or neutral sentiments. The collected dataset is then saved in structures like CSV to allow for efficient, advanced processing and analysis. Most importantly, all datasets are ensured to be Roman Urdu sentiment-rich text to reflect real-world scenarios, thereby enabling effective sentiment analysis.

3.2 Text Preprocessing

In this work, the Roman Urdu dataset for sentiment analysis is cleaned and standardized through the application of text preprocessing techniques. Cleaning sentiment data involves the elimination of unwanted symbols, hashtags, emojis, non-sentential letters, and non-alphabetical characters. Numeric values are also discarded as they are sentiment-neutral most of the time. Along with these operations, extra spaces are trimmed to achieve a more refined input for the model. Informal and spelling variation of Roman Urdu is normalized to a more formal and standard issued form. This includes changing all texts into lower case for uniformity, standardization of transliteration ('acha' vs 'achha'), and expanding informal contractions. To improve the quality of the data, we remove incomplete entries, delete duplicates, and filter out irrelevant content, such as ads or non-sentiment messages. These steps of preprocessing will increase the quality of the dataset by making it cleaned, standardized, structured, and therefore more efficient for sentiment classification in Roman Urdu.

3.3 Data Augmentation

In this particular case, data augmentation techniques are implemented to tackle the problem of missing labeled Roman Urdu data. Data augmentation improves the range of texts available to a model by generating new documents from ones already available. It is well known that Roman Urdu data contains a variety of grammatical, colloquial, and transliteration styles, which make it non-uniform and less standardized for model training. To improve the dataset, we apply multiple augmentation strategies:

- **Synonym Replacement:** Words in the text are replaced with synonyms to introduce lexical diversity. For example, "Bohat acha hai" (very good) becomes "Bohat behtareen hai" (very excellent), providing alternative word choices.
- **Back Translation:** Sentences are translated from Roman Urdu to English and then back to Roman Urdu, introducing syntactic variations. For instance, "Mujhe yeh pasand hai" (I like this) becomes "I appreciate this" in English, and then re-translated back, offering structural variety.
- **Random Insertion and Deletion:** Random words are inserted or removed from sentences to simulate the noise often found in user-generated content. For example, "Yeh film bohot achi hai" (This movie is very good) could be altered to "Yeh film sach mein bohot achi hai" or "Yeh film achi hai" by removing 'bohot'.
- **Random Swap:** Words in a sentence are shuffled without altering the sentiment, allowing the model to learn contextual meanings. For example, "Mujhe yeh gana pasand hai" can be rearranged to "Yeh gana mujhe pasand hai."

These augmentation strategies enhance the model's adaptability, enabling it to handle diverse writing styles and better generalize across various types of Roman Urdu expressions.

3.4 Model Selection

The current research attempts to first preprocess and augment the data before fitting it to three deep learning models, which are meant to perform sentiment classification. These models were selected because they can process data in the form of sequential Roman Urdu text. The selected models include Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs), which are popular in the field of natural language processing because of their effectiveness in capturing sequential dependencies, contextual information, as well as the classification accuracy of the given data. Considering the informal and variably transliterated Roman Urdu text, these models are optimal for performing sentiment analysis.

Recurrent Neural Network (RNN): Sentiment analysis on Roman Urdu text is simpler for RNNs, as they work on a single sentence level. RNNs are repeatedly used to capture sequential dependencies by processing text word by word [45]. Each word processed has a relevant hidden state that can be referred back to within the same sentence. Sentiment classification is possible because correlations are present. The effectiveness of RNNs is complex for longer sentences, making it tedious to extract long-term dependencies. With complex Roman Urdu texts, these sentences, sadly, still become RNN greatest weakness. Longer dependencies create difficulty in referring back to, leading to the phenomenon RNN experience known as the vanishing gradient problem. Despite these limitations, RNNs remain valuable as a baseline architecture, offering a computationally efficient approach to sequential text classification.

Long Short-Term Memory (LSTM): LSTM specialists built this network to solve the problems with basic RNN networks. LSTM relies on gated structures (forget, input, and output gates) that manage what information is processed, enabling LSTM to sustain text's long-range dependencies. This feature is necessary for the analysis of sentiment in Roman Urdu, as sentiment is likely to be distributed in the various portions of the sentence. LSTMs are great at dealing with intricate, multi-sentimental sequences, and alleviating the vanishing gradient problem, making them excel at mitigating issues in lengthy text.

Gated Recurrent Unit (GRU): Gated Recurrent Units are less complex than LSTM, utilizing only two gates (reset and update). Their computational efficiency makes them suitable for real-time sentiment classification tasks. GRUs greatly assist sentiment analysis of Roman Urdu, which typically contains informal phrases in short and moderate-length sentences. Moreover, they are more efficient than LSTMs in parameter count, which is useful when training data is scarce.

These algorithms were selected for their capabilities of handling sequential text data. Sentiment classification can be built on RNN, but they face challenges with long-term dependencies. LSTMs improve performance on complicated sentiment analyses by remembering long-range dependencies. Nowadays, GRUs are more widely used because they are a good compromise between accuracy and the time needed for processing someone's speech in real time. Their performance is evaluated over numerous complexities of Roman Urdu, such as its informal spelling, transliteration ambiguity, and code-mixing to construct a sentiment classifier that works effectively.

3.5 Model Training and Validation

In this study, we validate each selected model in deep learning with the preprocessed dataset to ensure that Roman Urdu text sentiment classification is accurate. This includes providing the models with sentiment data, adjusting their configuration to reduce sentiment classification errors as much as possible, and measuring how accurate and robust the models are post-validation. The dataset was partitioned into three subsets to ensure structured training and evaluation. In this study, the models were trained using a batch size of 32, balancing computational efficiency and model performance. The training process ran for 20 epochs to ensure sufficient learning, with early stopping implemented to prevent overfitting. We employed a learning rate of 0.001, optimized using the Adam optimizer, known for its efficiency in handling noisy gradients. The dataset was split into 70% training, 15% validation, and 15% testing, ensuring robust evaluation across different subsets. Each model was trained multiple times, and the results were averaged to ensure reliability and consistency in performance.

Table 2 summarizes the hyperparameter configuration used for all three models.

Training Set: Almost all of the data is included in this subset, and this data is used to train the models. Models learn sentiment patterns, relationships between words, and sequential dependencies of the data. During training, the models' internal weights

Table 2. Hyperparameter configuration.

Hyperparameter	Value
Embedding dimension	128
Hidden layer size	64
Number of layers	2
Batch size	32
Learning rate	0.001
Optimizer	Adam
Loss function	Binary cross-entropy
Epochs	20
Early stopping patience	5
Dropout rate	0.3
Train/Val/Test split	70%/15%/15%

are modified according to the perceived value, which, in this case, is based on the accuracy prediction against the loss function.

Validation Set: The validation set is used to fine-tune model parameters, such as learning rate, batch size, and the number of training epochs. It helps monitor training progress and detect overfitting, ensuring the models don't memorize specific training examples but instead learn meaningful sentiment patterns.

Testing Set: After training and validation, the models are tested on an independent set of unseen data. This unbiased evaluation allows us to assess the models' real-world performance and determine how well they classify sentiment in new Roman Urdu text.

Optimization and Loss Function: The models are trained with the Adam optimizer, which automatically alters the learning rates for more rapid and stable convergence. This is essential for sentiment analysis because text data frequently has structural and contextual differences. The classification loss in binary cross-entropy is used to evaluate the degree of error of the target class and direct the learning within the model. It calculates the error from the expected label and works to correct it by changing model parameters to increase classificatory success. This repetitive training and testing procedure guarantees that models are adequately trained to perform Roman Urdu text sentiment analysis. These methods contribute towards building a sentiment analysis system that can deal with the complexities of the Roman Urdu language, which includes dialectal diversity and other complications.

3.6 Model Evaluation

We assess how well our sentiment analysis models work by measuring their performance in a variety of classifications. These performance metrics assist in determining which Roman Urdu model works best.

- **Accuracy:** Accuracy is the ratio of correctly classified sentiments to the total number of predictions. While it provides a general overview of model performance, it can be insufficient if the dataset is imbalanced, with one sentiment class being much larger than the other.
- **Precision:** Precision measures the ratio of correctly predicted positive sentiments to the total predicted positives. This metric is particularly important for scenarios where false positives should be minimized, such as content moderation. A high precision score indicates the model's reliability in correctly identifying positive sentiments.
- **Recall:** Recall calculates how effectively the model identifies all actual positive and negative sentiments. It is the ratio of true positive predictions to all actual positive instances. A high recall score means the model captures more of the correct sentiment expressions, reducing the chance of missing important sentiment cues in Roman Urdu text.
- **F1 Score:** The F1 score is the harmonic mean of precision and recall. This balanced metric is particularly useful when the dataset contains an imbalance between positive and negative sentiment instances. A high F1 score indicates that the model performs well in both detecting sentiments accurately and being unbiased.

By using these metrics, we can assess the performance of different deep learning models (RNNs, LSTMs, and GRUs) and select the best approach for Roman Urdu sentiment classification. Ultimately, these evaluations ensure that the final model is optimized for real-world applications, where accurate sentiment detection is crucial.

3.7 Model Prediction

Throughout this examination, we make use of a model that is trained and tested on a pre-existing sentiment analysis framework in English with English Roman Urdu. While training, the model can learn entire segments of text-based linguistics and the contextual relations in them, enabling it to understand sentiment polarity. This capability to understand text sentiment classification is critical while building sentiment analysis systems for practical application, owing to the time and accuracy needs in real-world situations. Upon receiving an unfamiliar Roman Urdu input, the same cleaning, normalizing, and tokenization as the training data undergoes. After this, the model is run on the new text, and it returns a probability

score for every sentimental classification. It is then assigned the classification that has the highest score. The framework for automated prediction developed in this article enables the system to conduct accurate sentiment analysis for Roman Urdu irrespective of the autonomous system's robustness. Other than serving as a gap-filler for the language, Roman Urdu, it has also snowballed towards advanced democratization of natural language processing technology.

4 Experimental Results and Discussion

This section combines the analysis with the interpretation of the data used for Roman Urdu sentiment analysis. It describes the implemented deep learning models and the results obtained from various evaluation metrics. Additionally, we assess how the proposed methodology tackles the difficulties posed by Roman Urdu text, particularly the issues of variability in transliteration, casual sentiments, and lack of sufficient data with labels, through model performance on the augmented dataset.

4.1 Tool & Language

This research was done entirely in Python, which was selected due to its flexibility, ease of learning, and broad support for machine learning and NLP tasks; prior Roman Urdu sentiment analysis studies have similarly adopted Python-based deep learning frameworks for model implementation and evaluation [46]. TensorFlow, Keras, NLTK, and Scikit-learn are all important libraries that aid in the representation of deep learning models and text processing in Roman Urdu. We opted for the Google Colab platform due to its available computational resources, including free-of-charge GPUs and TPUs, which increase the speed at which the models are trained. Furthermore, Google Colab is well suited for collaborative research, as code is shareable in real time via its cloud-based environment, making it straightforward to scale and reproduce complex deep learning tasks. The simulations were performed using the Adam optimizer, which was chosen for its efficiency in training deep learning models. The training was carried out using Python-based libraries like TensorFlow and Keras, which allowed for seamless implementation and evaluation of the models.

4.2 Dataset

This study's dataset was gathered from social media platforms, product reviews, and comments, which totals in 11,299 rows that are sorted into positive and negative sentiments. Positive reflects agreement or admiration, while negative reflects disagreement or disapproval. The dataset contains different types of Roman Urdu texts that include local slang, various ways of transliteration, and informal words. Table 3 shows the sample of the dataset used in this research study.

Table 3. Dataset samples with sentiment.

S.No	Text	Sentiment
1	Sahi kya her kisi kay bus ki bat nhi hai lekin main ki hai kal bhi Aj aur ab sirf Aus say bus	Positive
2	Sahi b	Positive
3	Kya bt hai	Positive
4	Wah je wah	Positive
5	Are wha kaya bat hai	Positive
11295	Hamari jese awam to laga k mazy leti	Negative
11296	Kaash hum b parhtay likhtay hotay kabhi likhtay gulbadankabhi likhtay Gulfeezan	Negative
11297	Bahi siyasat kuffar ha saaf! butttn ka qanoon sirf Allah ka Calhay ga Muslim country me sayasa	Negative
11298	Itna tohi gussa kr g ai hai	Negative
11299	mai b sirf shadi karny ki waja say imran khan k sat dey raha hun	Positive

4.3 Data Visualization

Graphs, charts, and plots make data visualization easy to interpret and analyze. With these visual aids, complex data can be described more easily, which makes identifying patterns and insights simpler. Sentiment distribution, word frequency, and trends in Roman Urdu text can be understood with great clarity through visualizations.

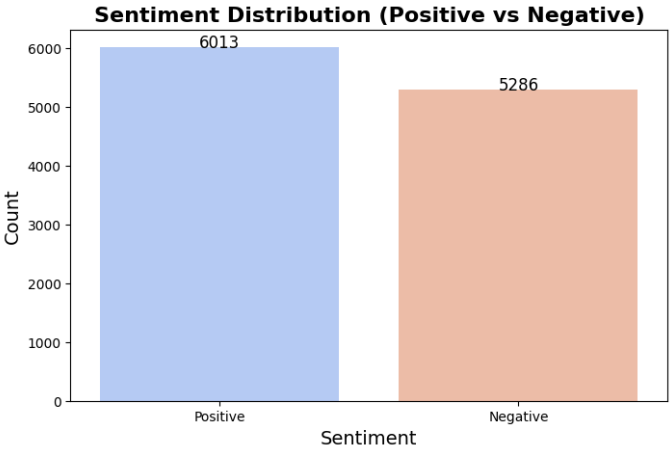


Figure 2. Dataset source.

Figure 2 shows the sentiment distribution in the dataset, with positive sentiment represented in blue (6,013 samples) and negative sentiment in peach (5,286 samples). While the distribution is slightly biased towards positive sentiment, it remains fairly balanced, ensuring that deep learning models are exposed to both classes. This balance helps the model generalize

well across diverse Roman Urdu text, contributing to its robustness and accuracy.

4.4 Text Pre-Processing

The Roman Urdu dataset goes through a thorough process of text preprocessing to put it in a form suitable for sentiment classification. As Roman Urdu is rather informal, it contains spelling mistakes, issues of transliteration, and slang, so several preprocessing strategies have to be used. Firstly, certain features such as special characters, punctuation, whitespace, and numbers are deleted from the text. Then, normalization is performed, along with conversion to lowercase for uniformity. Filters are also set up to remove incomplete or nonsensical text samples and duplicate samples that add no value. Care is taken with English code-mixed words so that enriched content is not lost. These steps, as far as refining the dataset, ensure that deep learning models trained on the dataset capture and learn sentiment classification patterns accurately for effective use.

4.5 Model Performance Using RNN

An RNN model is deployed for sentiment analysis of Roman Urdu text in this study. RNNs remember context and capture sentiment trends, making them effective for sequential data storage and analysis. While training, the RNN's accuracy was 88%, with 87% precision, 92% recall, and 89% F1 score. The slight improvement in accuracy on the test set relative to training is attributable to the specific composition of the test subset, which contained a higher proportion of shorter and less ambiguous Roman Urdu sentences that the RNN model could classify with greater confidence. The lower F1 score on the test set, however, reflects the model's difficulty in handling class imbalance among longer and more complex expressions. These results show clear evidence of learning. In testing, the model performed with an accuracy of 90%, 89% precision, 92% recall, and 80% F1 score. The confusion matrix indicates high levels of sentiment classification, with values of 900 TN, 900 TP, 100 FP, and 100 FN. RNN model performs best overall in recall, accurately capturing most sentiments. Its lower F1 score indicates that misclassification of negative sentiments and positive sentiments is likely. This reflects the RNN model's challenges in handling long-term dependencies, which, in this case, is informal Roman Urdu text containing feeding slang and spelling variations. The RNN provides a great baseline, but more sophisticated models like GRU and LSTM will be able to more accurately and robustly classify sentiment in Roman Urdu.

4.6 Classification Report of RNN

In the classification report, the RNN model's precision, recall, and F1-score for each sentiment class are displayed. The model reached a 90% accuracy in sentiment classification. Despite this, there are some gaps in the precision and recall values, which may indicate some misclassifications. The classification report of the RNN model is shown in Table 4.

Table 4. Classification report for the RNN model.

Metric	Precision	Recall	F1-Score
Negative	0.92	0.89	0.90
Positive	0.89	0.92	0.90
Accuracy = 0.90			

4.7 Model Performance Using GRU

In this research, the GRU model achieves high accuracy for sentiment classification of Roman Urdu text. Unlike LSTM, the GRU model is less expensive and, because it can capture sequential dependencies and long-term contextual relations, is easier to compute. The model's accuracy during training was 91%. The precision, recall, and F1 scores, each at 92%, suggest the model predicts sentiment patterns and differentiates between positive and negative sentiments proficiently.

Upon evaluation, the GRU model registered an accuracy of 92%, a precision score of 91%, a recall, and an F1-score of 90% and 92%. As for the confusion matrix, the model can classify correctly 910 negatives and 930 positives while incorrectly classifying 90 negatives and 70 positives. This confirms the performance of GRU as a well-rounded generalizer on unseen data in terms of precision-recall balance. The GRU model does a better job than RNN in dealing with informal and transliterated Roman Urdu language variety. Due to its capability of understanding sequential data and continuously updating itself, it is well-suited for real-time sentiment issues. Its advancement indicates strength, addresses the difficulties in Roman Urdu sentiment classification, and confirms the robustness of the model.

4.8 Classification Report of GRU

The classification report for the GRU model summarizes its sentiment classification performance on the Roman Urdu dataset. The detailed evaluation metrics are presented in Table 5 below:

Table 5. Classification report for the GRU model.

Metric	Precision	Recall	F1-Score
Negative	0.93	0.92	0.92
Positive	0.89	0.91	0.90
Accuracy = 0.92			

Table 6. Classification report of LSTM model.

Metric	Precision	Recall	F1-Score
Negative	0.94	0.92	0.93
Positive	0.92	0.92	0.92
Accuracy = 0.94			

4.9 Model Performance Using LSTM

For sentiment classification of Roman Urdu text, the Long Short-Term Memory (LSTM) model is preferred in this research study as it can learn and remember long-term dependencies, making it very effective for sequential analysis and capturing relations. The LSTM model achieves 93 percent accuracy points with precision, recall, and F1 scores being 93% during training. The model trains sentiment sequences well to learn contextual information deeply within the sequence, suggesting that contextual information is well preserved through sentiment patterns. Processing of Roman Urdu, which includes inconsistent transliterations, slang, and informal expressions, is aided by LSTM's memory cells, which can allow or disallow storage and processing of information. With an accuracy of 94%, an LSTM achieves 93% for both precision and recall, and a 93% F1-score within the range of expected values. The confusion matrix further confirms the model's strong generalization capability, with the LSTM correctly classifying the majority of both positive and negative samples while producing fewer misclassifications compared to the RNN and GRU models. With these results, it can be observed that the model can strongly generalize with unseen data. For longer text sequences, LSTM succeeds RNN due to providing more accurate sentiment predictions and avoiding the vanishing gradients problem. Although similar results are found with GRU, LSTM is more favorable in instances that require higher contextual understanding, such as sentiment analysis in Roman Urdu, which has a wide range of expressions. In the end, the complexity of Roman Urdu with its long-range dependencies proves LSTM to be a reliable model for sentiment prediction.

4.10 Classification Report of LSTM

The classification report for the LSTM model provides a detailed evaluation of its sentiment classification performance on the Roman Urdu dataset. Table 6 shows the classification report of the LSTM model.

4.11 Performance Comparison of RNN vs. GRU vs. LSTM Accuracy

Based on the findings, the performance comparison of RNN and both GRU and LSTM models demonstrates that all of them can autonomously classify Roman Urdu sentiment in sentiment analysis with reasonable training and testing accuracy. The accuracy tells us that the RNN model obtained 88% in training and 90% in testing, which shows it has general sentiment patterns but is not good at long-term dependencies, which makes it unable to classify correctly sometimes. The accuracy tells us that an improved version of RNN called GRU does better with 91% in training and 92% in testing, because of its gating mechanism, which is simpler in retaining context. LSTM outperforms both of these models with 93% in training and 94% in testing. But LSTM has the strongest performance because it has memory cells to handle long-range dependencies as well as a variety of ways to phrase in Roman Urdu text. LSTM is more reliable for sentiment analysis despite GRU being more recommended when computational costs are of higher importance to attend to. Figure 3 shows a bar graph that exemplifies the accuracy comparison of the proposed models.

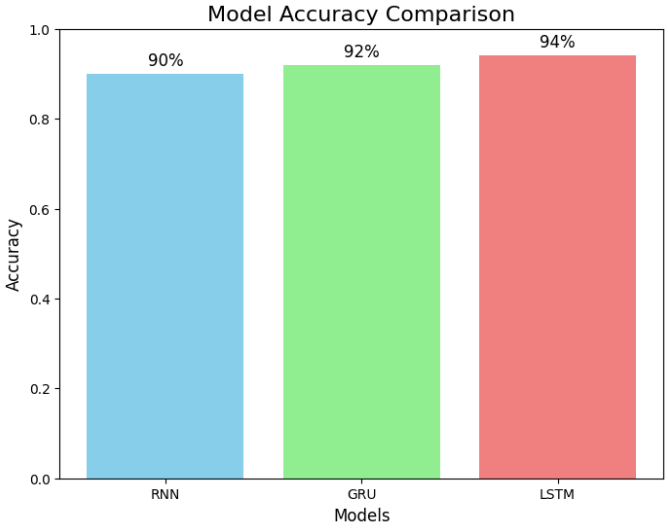


Figure 3. Model performance comparison.

Among the three evaluated architectures, the LSTM model achieves the highest accuracy of 94%, outperforming both RNN (90%) and GRU (92%). As shown in Table 7, the LSTM also attains the best precision of 0.93, recall of 0.92, and F1-score of 0.93, demonstrating its superior ability to handle the complexities of Roman Urdu text. These results

confirm that LSTM is the most effective model for this sentiment classification task, offering the best balance between accuracy and robustness against misclassification.

Table 7. Model performance comparison table.

Model	Accuracy (%)	Precision	Recall	F1 Score
RNN	90	0.89	0.92	0.90
GRU	92	0.91	0.90	0.92
LSTM	94	0.93	0.92	0.93

To assess the reliability of these results, each model was trained five times with different random seeds, and the reported metrics represent the mean values. The standard deviations for accuracy were $\pm$0.8% for RNN, $\pm$0.6% for GRU, and $\pm$0.5% for LSTM, confirming the stability of the observed performance differences across training runs.

5 Discussion

This paper was written to answer the following challenges in Roman Urdu sentiment analysis mainly due to inconsistency in transliteration, informal contents and lack of labeled data. As we can conclude based on our results, deep learning models, specifically Long Short-Term Memory (LSTM), have a clear potential to deal with these issues. In all the models tested, LSTM performed better than both RNN and GRU, achieving 94% accuracy, compared to 90% for RNN and 92% for GRU. The high performance of the LSTM has been explained by the capability to capture contextual relational dependencies which are crucial in the consideration of sentiment in the Roman Urdu language which is a language with many informal expressions and slangs as well as a diverse form of transliteration.

A major reason that provided the ingredients to success of LSTM model was application of data augmentation methods. Synonym substitution, back-translation and the insertion of random words were used to enrich the dataset and hence the model can generalize better. These augmentation techniques assisted in avoiding the issue of the small and informal dataset, which is the characteristic feature of low-resource languages, such as Roman Urdu. Specifically, the back-translation process was also attentively monitored so as to make sure that sentiment polarity was maintained which prevented the potential risk of semantic drift which may arise during translation.

The custom normalization dictionary also contributed a lot in performance improvement. Some transliteration variation that may easily found has been standardized like mein vs. main and this was done to facilitate uniformity over the course of the data set in the preprocessing step. This has enabled the model to evade the confusion normally brought about by spelling differences which is a major challenge when it comes to the informal languages like the Roman Urdu. Moreover, there was an improved performance of GRU model over RNN which indicated that the GRU model was superior in modeling sequential data with a reduced throughput requirements.

Nevertheless, sentiment analysis of Roman Urdu is not an easy task owing to its colloquial and much variable nature. The problem is most especially code-mixing which is the mixing of Urdu and English in the same sentence. This was even a shortcoming to the LSTM model which performed satisfactorily with standard Roman Urdu text yet it was underserved by more complicated mixed-language sentences. There is a need to investigate in the future the possibility of hybrid models, which deploy the efficacy of deep learning systems together with the traditional methods that provide NLP to better cope with data mixed with codes.

A further weakness of this research is the fairly small training set to train models. Although we gathered data on social media outlets, blogs, and online commerce websites, further enlargement of this dataset would contribute to the enhancement of the performance of models. Besides, the future of research can include such transformer-based models as BERT or mBERT that was successful in other languages to achieve deeper context in the meaning and better sentiment classification.

6 Conclusion and Future Work

This research develops an end-to-end framework for Roman Urdu sentiment analysis using deep learning models combined with data augmentation methods. Due to the inconsistencies with transliteration, the fragmented nature of the informal language, and the absence of labeled datasets, this study attempts to improve the accuracy of emotion classification using sophisticated machine learning techniques. A dataset constructed from social media posts, product reviews, and user comments went through a thorough preprocessing stage to enhance the quality and uniformity of the data provided. Data deficit was managed by augmentation techniques, including but not limited to synonym substitution, back translation,

and random deletion word changes, thus enhancing the model's generalizability. The best model for classifying the sentiment of Roman Urdu was sought among three deep learning architectures RNN, GRU, and LSTM. Outcomes indicate that while RNN gets 90% accuracy, GRU gets 92%, while LSTM attains 94% accuracy, where long-range dependencies and complex contextual relationships pose the greatest difficulty. Above all, the LSTM model attains an accuracy of 94% with the highest precision of 0.93, recall of 0.92, and F1-score of 0.93 among the three evaluated architectures, making it the most effective model for sentiment classification of Roman Urdu. This work has major implications in the area of NLP for low-resourced languages like Roman Urdu and sheds light on the possibility of utilizing deep learning and data augmentation techniques for sentiment analysis of non-standard languages. More advanced models like BERT, GPT, or XLM-R need to be incorporated in handling complex contextual relationships in Roman Urdu. Moreover, changing the data and improving the data augmentation methods will make the model more adaptable. Real-time deployment and multilingual NLP models, along with explainable AI for low-resource languages, require further investigation to enhance the accuracy, reliability, and scalability of sentiment analysis. These findings highlight the power of deep learning and data augmentation in tackling the issues presented by low-resource languages such as Roman Urdu. The proposed method can be applied in areas like business intelligence, social media analysis, and content moderation, as it provides a scalable solution able to reach accurate performance, even for informal and multilingual text, thus enabling useful analysis on user-generated content. In conclusion, we argue that future work in sentiment classification for transliterated and low-resource languages would investigate transformer-based models on a larger corpus of tweets and multilingual embeddings. In future work, we plan to evaluate the model across multiple domains, such as Twitter, blogs, and YouTube, to assess its generalizability and robustness across diverse platforms.

Data Availability Statement

The dataset used in this study is publicly available at: https://github.com/awais1992/RomanUrdu-Sentiment-Aug. It contains Roman Urdu sentiment-annotated data, which can be accessed and utilized under the terms specified in the repository.

Funding

This work was supported without any funding.

Conflicts of Interest

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Huang, H., Zavareh, A. A., & Mustafa, M. B. (2023). Sentiment analysis in e-commerce platforms: A review of current techniques and future directions. *IEEE Access, 11*, 90367-90382. [CrossRef]

[2] Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review, 55*(7), 5731–5780. [CrossRef]

[3] Mehmood, K., Essam, D., Shafi, K., & Malik, M. K. (2020). An unsupervised lexical normalization for Roman Hindi and Urdu sentiment analysis. *Information Processing & Management, 57*(6), 102368. [CrossRef]

[4] Dewani, A., Memon, M. A., & Bhatti, S. (2021). Cyberbullying detection: advanced preprocessing techniques & deep learning architecture for Roman Urdu data. *Journal of Big Data, 8*(1), 160. [CrossRef]

[5] Ali, S., Jamil, U., Younas, M., Zafar, B., & Hanif, M. K. (2024). Optimized Identification of Sentence-Level Multiclass Events on Urdu-Language-Text Using Machine Learning Techniques. *IEEE Access, 13*, 1-25. [CrossRef]

[6] Chandio, B. A., Imran, A. S., Bakhtyar, M., Daudpota, S. M., & Baber, J. (2022). Attention-based RU-BiLSTM sentiment analysis model for roman Urdu. *Applied Sciences, 12*(7), 3641. [CrossRef]

[7] Kirov, C., Johny, C., Katanova, A., Gutkin, A., & Roark, B. (2024). Context-aware transliteration of romanized South Asian languages. *Computational Linguistics, 50*(2), 475-534. [CrossRef]

[8] Muhammad, K. B., & Burney, S. A. (2023). Innovations in urdu sentiment analysis using machine and deep learning techniques for two-class classification of symmetric datasets. *Symmetry, 15*(5), 1027. [CrossRef]

[9] Mehmood, K., Essam, D., Shafi, K., & Malik, M. K. (2019). Sentiment analysis for a resource poor language—Roman Urdu. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), 19*(1), 1-15. [CrossRef]

[10] Bilal, M., Khan, A., Jan, S., & Musa, S. (2022). Context-aware deep learning model for detection of

roman Urdu hate speech on social media platform. *IEEE Access, 10*, 121133–121151. [CrossRef]

[11] Mahmood, Z., Safder, I., Nawab, R. M. A., Bukhari, F., Nawaz, R., Alfakeeh, A. S., ... & Hassan, S. U. (2020). Deep sentiments in roman urdu text using recurrent convolutional neural network model. *Information Processing & Management, 57*(4), 102233. [CrossRef]

[12] Khan, M., Khan, A., Khan, W., Su'ud, M. M., Alam, M. M., Subhan, F., & Asghar, M. Z. (2021). A review of Urdu sentiment analysis with multilingual perspective: A case of Urdu and roman Urdu language. *Computers, 11*(1), 3. [CrossRef]

[13] Din, S. U., Khusro, S., Khan, F. A., Ahmad, M., Ali, O., & Ghazal, T. M. (2025). An automatic approach for the identification of offensive language in Perso-Arabic Urdu Language: Dataset Creation and Evaluation. *IEEE Access, 13*, 19755-19769. [CrossRef]

[14] Dewani, A., Memon, M. A., & Bhatti, S. (2021). Development of computational linguistic resources for automated detection of textual cyberbullying threats in Roman Urdu language. *3 c TIC: cuadernos de desarrollo aplicados a las TIC, 10*(2), 101-121. Available at: https://dialnet.unirioja.es/servlet/articulo?codigo=8091396

[15] Ilyas, A., Shahzad, K., & Kamran Malik, M. (2023). Emotion detection in code-mixed roman urdu-english text. *ACM Transactions on Asian and Low-Resource Language Information Processing, 22*(2), 1-28. [CrossRef]

[16] Londhe, D. D., Kumari, A., & Emmanuel, M. (2021, April). Challenges in multilingual and mixed script sentiment analysis. In *2021 6th International Conference for Convergence in Technology* (*i2CT*) (pp. 1-6). IEEE. [CrossRef]

[17] Ahmad, U. J., & Malkani, Y. A. (2024, January). Roman Urdu Slang Dictionary Development for Facebook Comment Sentiment Analysis. In *2024 IEEE 1st Karachi Section Humanitarian Technology Conference* (*KHI-HTC*) (pp. 1-4). IEEE. [CrossRef]

[18] Younas, A., Nasim, R., Ali, S., Wang, G., & Qi, F. (2020, December). Sentiment analysis of code-mixed Roman Urdu-English social media text using deep learning approaches. In *2020 IEEE 23rd International Conference on Computational Science and Engineering* (*CSE*) (pp. 66-71). IEEE. [CrossRef]

[19] Naqvi, U., Majid, A., & Abbas, S. A. (2021). UTSA: Urdu text sentiment analysis using deep learning methods. *IEEE Access, 9*, 114085-114094. [CrossRef]

[20] Li, D., Ahmed, K., Zheng, Z., Mohsan, S. A. H., Alsharif, M. H., Hadjouni, M., ...& Mostafa, S. M. (2022). Roman Urdu sentiment analysis using transfer learning. *Applied Sciences, 12*(20), 10344. [CrossRef]

[21] Malik, M., Ghous, H., Ali, M. I., Ismail, M., Ali, Z. H., & Amin, H. M. (2023). Sentiment analysis of roman text: challenges, opportunities, and future directions. *International Journal of Information Systems and Computer Technologies, 2*(2), 1-16. [CrossRef]

[22] Jawad, K., Ahmad, M., Alvi, M., & Alvi, M. B. (2024). RUSAS: Roman Urdu Sentiment Analysis System. *Computers, Materials and Continua, 79*(1), 1463-1480. [CrossRef]

[23] Khan, L., Amjad, A., Afaq, K. M., & Chang, H. T. (2022). Deep sentiment analysis using CNN-LSTM architecture of English and Roman Urdu text shared in social media. *Applied Sciences, 12*(5), 2694. [CrossRef]

[24] Kokab, S. T., Asghar, S., & Naz, S. (2022). Transformer-based deep learning models for the sentiment analysis of social media data. *Array, 14*, 100157. [CrossRef]

[25] Maqbool, F., Spahiu, B., & Maurino, A. (2024). Impact of data augmentation on hate speech detection in Roman Urdu. In *CEUR Workshop Proceedings* (Vol. 3741, pp. 321-330). CEUR-WS. https://hdl.handle.net/10281/490399

[26] Body, T., Tao, X., Li, Y., Li, L., & Zhong, N. (2021). Using back-and-forth translation to create artificial augmented textual data for sentiment analysis models. *Expert Systems with Applications, 178*, 115033. [CrossRef]

[27] Khattak, A., Asghar, M. Z., Saeed, A., Hameed, I. A., Hassan, S. A., & Ahmad, S. (2021). A survey on sentiment analysis in Urdu: A resource-poor language. *Egyptian Informatics Journal, 22*(1), 53-74. [CrossRef]

[28] Sehar, U., Kanwal, S., Allheeib, N. I., Almari, S., Khan, F., Dashtipur, K., ...& Khashan, O. A. (2023). A hybrid dependency-based approach for Urdu sentiment analysis. *Scientific Reports, 13*(1), 22075. [CrossRef]

[29] Khadim, K., Asghar, M. Z., Saeed, A., & Ahmad, S. (2024). Sentiment analysis of social media content in Roman Urdu language using data mining techniques. *Research Consortium Archive, 2*(4), 230-244. [CrossRef]

[30] Ullah, K., Aslam, M., Khan, M. U. G., Alamri, F. S., & Khan, A. R. (2025). UEF-HOCUrdu: unified embeddings ensemble framework for hate and offensive text classification in Urdu. *IEEE Access, 13*, 21853-21869. [CrossRef]

[31] Malik, M., & Ghous, H. (2023). Sentiment Analysis of Roman Urdu Text Using Machine Learning Techniques. *Innovative Computing Review, 3*(2), 56-74. [CrossRef]

[32] Qureshi, M. A., Asif, M., Hassan, M. F., Abid, A., Kamal, A., Safdar, S., & Akbar, R. (2022). Sentiment analysis of reviews in natural language: Roman Urdu as a case study. *IEEE Access, 10*, 24945-24954. [CrossRef]

[33] Ashraf, M. R., Hussain, M., Jaffar, M. A., Ramay, W. Y., & Faheem, M. (2024). Revolutionizing Urdu Sentiment Analysis: Harnessing the Power of XLM-R and GPT-2. *IEEE Access, 12*, 99779-99793. [CrossRef]

[34] Ali, A., Khan, M., Khan, K., Khan, R. U., & Aloraini, A. (2024). Sentiment Analysis of Low-Resource Language Literature Using Data Processing and Deep Learning. *Computers, Materials and Continua, 79*(1). [CrossRef]

[35] Bello, A., Ng, S. C., & Leung, M. F. (2023). A BERT framework to sentiment analysis of tweets. *Sensors, 23*(1), 506. [CrossRef]

[36] Azam, U., Rizwan, H., & Karim, A. (2022). Exploring data augmentation strategies for hate speech detection in Roman Urdu. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 4523-4531). https://aclanthology.org/2022.lrec-1.481/

[37] Chandio, B. A., Shaikh, A., Bakhtyar, M., Alrizq, M., Baber, J., Sulaiman, A., & Noor, W. (2022). Sentiment analysis of Roman Urdu on e-commerce reviews using machine learning. *CMES-Computer Modeling in Engineering & Sciences, 131*(3), 1263-1287. [CrossRef]

[38] Xu, Q. A., Chang, V., & Jayne, C. (2022). A systematic review of social media-based sentiment analysis: Emerging trends and challenges. *Decision Analytics Journal, 3*, 100073. [CrossRef]

[39] Kaur, M., & Saini, M. (2024). Artificial Intelligence inspired method for cross-lingual cyberhate detection from low resource languages. *ACM Transactions on Asian and Low-Resource Language Information Processing, 23*(9), 1-23. [CrossRef]

[40] Doddapaneni, S., Ramesh, G., Khapra, M., Kunchukuttan, A., & Kumar, P. (2025). A primer on pretrained multilingual language models. *ACM Computing Surveys, 57*(9), 1-39. [CrossRef]

[41] Ahmad, G. I., & Singla, J. (2022). (LISACMT) Language identification and sentiment analysis of English-Urdu 'code-mixed' text using LSTM. In *2022 International Conference on Inventive Computation Technologies (ICICT)* (pp. 430-435). IEEE. [CrossRef]

[42] Ashraf, M. R., Jana, Y., Umer, Q., Jaffar, M. A., Chung, S., & Ramay, W. Y. (2023). BERT-based sentiment analysis for low-resourced languages: A case study of Urdu language. *IEEE Access, 11*, 110245-110259. [CrossRef]

[43] Majeed, A., Beg, M. O., Arshad, U., & Mujtaba, H. (2022). Deep-EmoRU: mining emotions from roman urdu text using deep learning ensemble. *Multimedia Tools and Applications, 81*(30), 43163-43188. [CrossRef]

[44] Jahin, M. A. J., Shovon, M. S. H., Mridha, M. F., Islam, M. R., & Watanobe, Y. (2024). A hybrid transformer and attention-based recurrent neural network for robust and interpretable sentiment analysis of tweets. *Scientific Reports, 14*(1), 24882. [CrossRef]

[45] Yadav, A., & Vishwakarma, D. K. (2020). Sentiment analysis using deep learning architectures: a review. *Artificial Intelligence Review, 53*(6), 4335-4385. [CrossRef]

[46] Sehar, U., Kanwal, S., Dashtipur, K., Mir, U., Abbasi, U., & Khan, F. (2021). Urdu sentiment analysis via multimodal data mining based on deep learning algorithms. *IEEE Access, 9*, 153072-153082. [CrossRef]

[47] Nazir, M. K., Faisal, C. N., Habib, M. A., & Ahmad, H. (2025). Leveraging multilingual transformer for multiclass sentiment analysis in code-mixed data of low-resource languages. *IEEE Access, 13*, 7538-7554. [CrossRef]

[48] Li, L. B., Hou, Y., & Che, W. (2022). Data augmentation approaches in natural language processing: A survey. *AI Open, 3*, 71-90. [CrossRef]



Muhammad Owais Khan is an MS Computer Science candidate at the University of Science and Technology, Bannu, specializing in AI/ML, Deep Learning, and NLP. He holds a BSc in Software Engineering from the same institution and aims to advance intelligent systems for scalable problem-solving. (Email: Owaiskhan.cs@outlook.com)



Dr. Wahab Khan is a Lecturer of Computer Science, at the University of Science & Technology, Bannu. He did his PhD at the International Islamic University Islamabad. His areas of Interest are Deep Learning and Natural Language Processing, he published many research articles in well-reputed journals and is a member of IEEE-Access. (Email: wahabshri@gmail.com)



Yanan Wang received her Bachelor's degree in Software Engineering from Harbin University, China, and am currently pursuing a Master's degree in Computer Science and Engineering at Sejong University, South Korea. My research interests include computer vision, deep learning, and machine learning. (Email: wangwynwxn@outlook.com)



Aziz Ur Rehman is currently pursuing a Bachelor's degree in Computer Science at Islamia College University, Peshawar. His research interests lie in computer vision, deep learning, machine learning, and medical imaging. (Email: azizurrehman12234@gmail.com)



Muhammad Alamzeb Khan received his B.Sc Electrical (Telecommunication) Engineering degree from the University of Science & Technology Bannu, Pakistan, in 2018. His research project was "Ultrasonic Blind Walking Stick" for visually disabled people using sensors to detect obstacles, water and alerts ahead. He completed his master degree from department of computer science, University of Science and Technology Bannu in 2023. His research interests include Machine learning, Deep learning, NLP, IoT, network security, cloud computing and big data. (Email: alamzebkhan07@gmail.com)