**ARTICLE**

# Sentiment Analysis of Low-Resource Language Literature Using Data Processing and Deep Learning

**Aizaz Ali[1], Maqbool Khan[1,2], Khalil Khan[3], Rehan Ullah Khan[4] and Abdulrahman Aloraini[4,*]**

[1]Department of IT and Computer Science, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Haripur, 22620, Pakistan

[2]Software Competence Center Hagenberg, Softwarepark 32a, Hagenberg, 4232, Austria

[3]Department of Computer Science, School of Engineering and Digital Sciences, Nazarbayev University, Astana, 010000, Kazakhstan

[4]Department of Information Technology, College of Computer, Qassim University, P.O.Box 1162, Buraydah, Saudi Arabia

*Corresponding Author: Abdulrahman Aloraini. Email: a.aloraini@qu.edu.sa

**ABSTRACT**

Sentiment analysis, a crucial task in discerning emotional tones within the text, plays a pivotal role in understanding public opinion and user sentiment across diverse languages. While numerous scholars conduct sentiment analysis in widely spoken languages such as English, Chinese, Arabic, Roman Arabic, and more, we come to grappling with resource-poor languages like Urdu literature which becomes a challenge. Urdu is a uniquely crafted language, characterized by a script that amalgamates elements from diverse languages, including Arabic, Parsi, Pashtu, Turkish, Punjabi, Saraiki, and more. As Urdu literature, characterized by distinct character sets and linguistic features, presents an additional hurdle due to the lack of accessible datasets, rendering sentiment analysis a formidable undertaking. The limited availability of resources has fueled increased interest among researchers, prompting a deeper exploration into Urdu sentiment analysis. This research is dedicated to Urdu language sentiment analysis, employing sophisticated deep learning models on an extensive dataset categorized into five labels: Positive, Negative, Neutral, Mixed, and Ambiguous. The primary objective is to discern sentiments and emotions within the Urdu language, despite the absence of well-curated datasets. To tackle this challenge, the initial step involves the creation of a comprehensive Urdu dataset by aggregating data from various sources such as newspapers, articles, and social media comments. Subsequent to this data collection, a thorough process of cleaning and preprocessing is implemented to ensure the quality of the data. The study leverages two well-known deep learning models, namely Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), for both training and evaluating sentiment analysis performance. Additionally, the study explores hyperparameter tuning to optimize the models' efficacy. Evaluation metrics such as precision, recall, and the F1-score are employed to assess the effectiveness of the models. The research findings reveal that RNN surpasses CNN in Urdu sentiment analysis, gaining a significantly higher accuracy rate of 91%. This result accentuates the exceptional performance of RNN, solidifying its status as a compelling option for conducting sentiment analysis tasks in the Urdu language.

**KEYWORDS**

Urdu sentiment analysis; convolutional neural networks; recurrent neural network; deep learning; natural language processing; neural networks

## 1 Introduction

Sentiment analysis serves the purpose of determining the unlikeness of text and understanding people's perspectives on a given subject. Sentiment analysis is a way that can be used to analyze emotions in paragraphs, phrases, or even clauses. It can also be used to find indicators like grief, anger, fear, happiness, and more, which helps the reader better understand the mood that is being represented in the text. Its primary focus is on classifying text as positive, negative, neutral applicable to paragraphs, sentences, or clauses [1]. Many languages, especially highly resourced ones like English, Chinese, Arabic, and Roman, use sentiment analysis extensively. But the difficulties become more apparent when we focus on languages like Urdu that have little resources accessible. Given that a sizable portion of the population speaks Urdu, the lack of linguistic resources presents a special challenge to efficient sentiment analysis. To overcome these resource limitations, novel strategies and specialized models that are adapted to the subtleties of the Urdu language are needed. This will guarantee a more precise and perceptive analysis of the emotions represented in Urdu literature. It is crucial to examine Urdu text sentiment considering the rise in social media usage and the growing dependence on digital platforms for communication. Deep learning algorithms show great promise in fulfilling this need, they can classify sentiment in text with excellent accuracy by utilizing large datasets for training [2].

The Urdu language, widely spoken in Pakistan and sharing a script like Arabic, Hindi, and Pashtu [3], has seen an increased need for sentiment analysis. Urdu text is characterized by its right-to-left script, and the demarcation between words is not always clearly defined. For instance, the phrase 'یہ میری جگہ ہے' (this is my place) remains comprehensible despite the absence of spaces between words. Deep learning models have emerged as promising tools for analyzing Urdu text sentiments, leveraging extensive datasets for accurate sentiment classification [4]. The primary objective of this research is to perform sentiment analysis across varied data sources, encompassing social media comments, Facebook pages, and newspaper articles. Employing supervised deep learning techniques, the author endeavors to construct a resilient model with the ability to predict the polarity of Urdu text data. This approach underscores the comprehensive nature of the study, which seeks to address sentiment analysis across diverse linguistic contexts and platforms using advanced deep learning methodologies.

By employing natural language processing techniques for text preprocessing, this study embarks on training deep learning models using extensive datasets [5]. The resultant trained model exhibits the capacity to predict sentiment in new Urdu text, presenting a potential enhancement in the efficiency of sentiment analysis compared to conventional rule-based and lexicon-based systems. Furthermore, the utilization of deep learning models proves advantageous in mitigating challenges associated with the limited availability of labeled Urdu text data, as these models can adeptly learn from unannotated sources. This multifaceted approach underscores the potential of leveraging deep learning for more robust sentiment analysis in the Urdu language.

Examining the effectiveness of deep learning models in the sentiment analysis of Urdu literature, this paper conducts a comprehensive review of existing literature, delving into the challenges associated with this domain [6]. Through experimental endeavors on a curated dataset of Urdu text, the paper systematically compares the performance of various deep learning models. Notably, the study underscores the critical role of dataset selection and preprocessing techniques in enhancing the accuracy of these models. The findings contribute valuable insights into refining the application of deep learning in sentiment analysis, particularly within the context of Urdu literature. The study explores the potential applications of deep learning models in sentiment analysis of Urdu text, encompassing areas such as political and social media monitoring, customer feedback analysis, and brand reputation

management. The capability to accurately classify sentiment in Urdu text presents an opportunity for extracting valuable insights that can inform decision-making within these domains.

However, despite the promising outcomes exhibited by deep learning models in the analysis of Urdu text, there are notable challenges that persist. Although these models demonstrate remarkable accuracy in recognizing Urdu text, the field of text analysis, especially sentiment analysis (SA), grapples with significant obstacles. One such challenge is the absence of standardized Urdu datasets. Currently, there is a conspicuous lack of a suitable dataset for Urdu text, underscoring the need to develop a comprehensive dataset as an integral part of the proposed solution. The creation of a dedicated dataset is imperative for augmenting the accuracy and effectiveness of deep learning models applied to Urdu text analysis.

The paper is organized into distinct sections for clarity and coherence. Section 2 provides an overview of related work in the field. In Section 3, the focus shifts to introducing the Urdu literature dataset and detailing the methodology of the proposed model. Section 4 delves into the experiments conducted, presenting the results, and initiating a discussion around them. The paper concludes in the final section, summarizing key findings and implications. This structured approach ensures a systematic presentation of the research, facilitating a comprehensive understanding of the study's methodology, findings, and concluding remarks.

## 2  Related Work

Sentiment analysis has emerged as a highly researched topic in text analysis over the past decade. This section of the paper delves into recent endeavors and contributions pertaining to Urdu literature, exploring methodologies employed in analogous domains that exert an impact on this field. Various approaches have been put forth to augment the precision of classification results, and these are detailed below [7].

### 2.1  Urdu Language

The paper [8] approach to document-level text classification. Employing a single layer multi size filters convolutional neural network (CNN), the authors aim to address the challenges associated with processing entire documents and capturing contextual information crucial for accurate classification. The streamlined architecture of the CNN, featuring multisize filters, allows for the extraction of hierarchical features, contributing to a nuanced understanding of document content and potentially enhancing classification performance. While the specific experimental results and performance metrics are not detailed in the summary, the paper is expected to provide insights into the effectiveness of the proposed model compared to existing methods. Additionally, discussions on the model's strengths, potential limitations, and practical implications, as well as suggestions for future research directions, would likely contribute to the overall significance of the proposed document-level text classification method.

The paper [9] addresses the growing significance of regional language data on the Internet, particularly focusing on Urdu. With an increasing interest in Urdu sentiment analysis, the study introduces the Urdu Text Sentiment Analysis (UTSA) framework, leveraging deep learning techniques and various word vector representations. Stacked layers are applied to sequential models, and the role of pre-trained and unsupervised self-trained embedding models is explored. The results highlight that BiLSTM-ATT outperforms other models, achieving a notable 77.9% accuracy and 72.7% F1 score. The study underscores the effectiveness of deep learning methods in Urdu sentiment analysis, emphasizing the need for further exploration in morphologically rich languages like Urdu.

The paper [10] explores text classification as a tool for assigning predefined categories to text documents through supervised machine learning algorithms. The study applies five well-known classification techniques to an Urdu language corpus comprising 21,769 news documents across seven categories. Utilizing the concept of majority voting, a class is assigned to each document. Preprocessing techniques such as tokenization, stop words removal, and a rule-based stemmer are applied to make the data suitable for algorithmic processing.

With 93,400 features extracted after preprocessing, the machine learning algorithms achieve impressive results, attaining up to 94% precision and recall through the majority voting approach. The study showcases the effectiveness of this methodology in the realm of Urdu text classification, with potential applications in diverse fields such as spam detection, sentiment analysis, and natural language detection.

Iqbal et al. [11] conduct a study on sentiment analysis of social media content in Pashtu, a low-resource language. Employing deep learning techniques including convolutional neural networks (CNN) and long short-term memory (LSTM), the authors aimed to classify Pashtu social media content into positive, negative, and neutral categories. The dataset, manually annotated by native Pashtu speakers, contributed to promising results. CNN achieved an accuracy of 71%, while LSTM outperformed with an accuracy of 73.8%, surpassing other existing models. This study holds significance for Pashtu language processing, offering valuable applications in areas such as opinion mining, market analysis.

Khattak et al. [12] conduct a comprehensive review of sentiment analysis in Urdu, a resource-poor language akin to Pashtu script. Analyzing over 50 studies, they highlighted challenges such as a scarcity of linguistic resources and annotated datasets. The study explores various approaches, including lexicon-based and machine learning methods, offering insights into the complexities of Urdu's morphology and syntax. The authors suggest future research directions, emphasizing the need for more annotated datasets and improved techniques, as well as exploring transfer learning and multilingual models.

Noor et al. [13] contribute to e-commerce sentiment analysis in Roman Urdu using Support Vector Machines (SVM). The study demonstrates SVM's effectiveness and provides insights into customer sentiments. It discusses evaluation metrics and calls for further research to compare SVM with other approaches in Roman Urdu text, aiming to enhance decision-making in the e-commerce domain.

Saad et al. [14] contribute to Pashtu language processing by developing a rule-based stemming algorithm. The proposed algorithm outperformed existing algorithms, achieving an accuracy of 78.6%. This robust stemming algorithm is essential for various natural language processing applications, addressing challenges in low-resource languages like Pashtu.

Khan et al. [15] present a new dataset of Pashtu handwritten numerals and a deep learning-based algorithm for numeral recognition. The proposed approach achieved a high accuracy of 98.64% on the PHND dataset, providing a valuable resource for researchers working in Pashtu language processing and computer vision, with practical applications in finance and commerce.

Khalil et al. [16] propose a deep learning-based approach for recognizing Pashtu numerals using CNNs. Their approach achieved an accuracy of 91.4%, outperforming other methods and demonstrating practical applications in digitizing Pashtu manuscripts and historical documents.

Janisar et al. [17] demonstrate the feasibility of using machine learning techniques to identify hate speech in Pashto language tweets. SVM demonstrated the highest performance with an accuracy of 74%, contributing to hate speech detection in languages beyond English.

Singh et al. [18] enhance our understanding of Punjabi language morphology using deep learning classification techniques for sentiment analysis. The study provides valuable insights into morphological patterns and demonstrates the effectiveness of deep learning models in analyzing sentiment in Punjabi text.

Neelam et al. [19] address the overlooked domain of sentiment analysis in resource-poor languages, with a specific focus on Urdu. The researchers collected data from various blogs across 14 different genres and annotated it with the assistance of human annotators. The study employed three well-known classifiers—Support Vector Machine, Decision Tree, and k-Nearest Neighbor (k-NN). However, the initial results were unsatisfactory, with all three classifiers achieving accuracy below 50%. Attempts to improve results through ensemble classifiers did not yield significant improvements. Ultimately, the study concludes that k-NN outperformed Support Vector Machine and Decision Tree in terms of accuracy, precision, recall, and F-measure.

This conclusion is drawn after a careful analysis of results and subsequent adjustments to enhance classifier performance. The findings highlight the importance of tailored approaches and feature extraction techniques in sentiment analysis for resource-poor languages like Urdu.

The paper [20] emphasizes the role of sentiment analysis (SA) in decision-making and problem-solving, highlighting the importance of reliable results in biologically inspired machine learning approaches. The research focuses on SA for Urdu, employing five classifiers. Through 10-fold cross-validation, three top classifiers—Lib SVM, J48, and IBK—are selected based on high accuracy, precision, recall, and F-measure. IBK emerges as the best classifier with 67% of accuracy result. To validate this, sentence labels are predicted using training and test data, and three standard evaluation measures—McNamara's test, kappa statistic, and root mean squared error—are applied. IBK consistently outperforms others, affirming its status as the most reliable classifier. The study's contribution lies in its meticulous approach to verification, enhancing confidence in IBK as the optimal choice for Urdu sentiment analysis.

Khan et al. [21] focus on talking about sentiment analysis (SA) in relation to English and Roman Urdu dialects, especially on social networking sites. To solve these issues, the research suggests a novel deep learning architecture that combines a one-layer CNN model for local feature extraction with Long Short-Term Memory (LSTM) for long-term dependence preservation. The results of the study, which includes comprehensive testing on four corpora, show that the suggested model performs remarkably well in sentiment categorization for both Roman and Urdu text. In particular, on the MDPI, RUSA, RUSA-19, and UCL datasets, the model obtains high accuracy ratings of 0.841, 0.740, and 0.748, respectively. The results of the studies show that the Word2Vec CBOW model and the SVM classifier work well for sentiment analysis in Roman Urdu.

The study [22] examines Opinion Mining within the framework of user evaluations, highlighting the new area of Sentiment Analysis as a means of comprehending consumer sentiment. The research focuses on reviews published in Roman Urdu and presents a methodology to categorize these reviews according to their polarity. A dataset of 24,000 Roman Urdu reviews is produced by gathering raw data from reviews of 20 songs in the Indo-Pak music industry. Nine machine learning techniques are used in this study: ID3, Gradient Boost Tree, k-Nearest Neighbors, Artificial Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks, Support Vector Machines, Naïve Bayes, and Logistic Regression. Of them, Logistic Regression performs better, with cross-validation and testing accuracies of 91.47% and 92.25%, respectively.

The paper [23] tackles the issue of sentiment analysis studies being underrepresented in regional or low-resource languages such as Urdu. It gives the Urdu Dataset for Sentiment Analysis-23

(UDSA-23) and presents USA-BERT (Urdu Text Sentiment Analysis using Bidirectional Encoder Representations from Transformers), a deep learning-based methodology. The procedure entails using BERT-Tokenizer to preprocess Urdu reviews, generating BERT embeddings for every review, optimizing a deep learning classifier (BERT), and evaluating USA-BERT on two datasets (UCSA-21 and UDSA-23) using the Pareto principle. The evaluation shows notable gains in f-measure and accuracy, outperforming current techniques by as much as 25.87% and 26.09%, respectively. This closes the research gap for low-resource languages and demonstrates the efficacy of USA-BERT in Urdu sentiment analysis.

In Table 1, a compilation of deep learning-related research discussed in this study is provided. While highlights recent research utilizing diverse deep learning models for sentiment analysis, it is notable that few studies have explored the effects of employing different dataset configurations and accuracy result. Specifically, variations in binary or multiclassification and balanced or imbalanced setups have been relatively underexplored. In this study, a range of datasets featuring five distinct data labels, along with diverse setups, were employed across various models. The research further delved into examining the influence of these different dataset configurations on model accuracy.

**Table 1:** Comparison of literature by different models

| Paper reference | Embedding technique | Models | Dataset Text/ Language | Accuracy (%) |
| --- | --- | --- | --- | --- |
| [8] | Word2Vec | CNN | Urdu | The accuracy of the model is 73% and 71%, respectively |
| [9] | Word2Vec | BiLSTM-ATT | Urdu | The accuracy of the model is 72.75% |
| [10] | Word2Vec | SVM, Naïve Bayes | Urdu | The accuracy of the model is 75% to 94% |
| [11] | Word2Vec | LSTM, CNN | Urdu | The accuracy of the model is 73% |
| [12] | Word2Vec | ML, DL | Urdu | The accuracy of the model is 71% to 91% |
| [13] | Word2Vec | CNN | Pashtu | The accuracy of the model is 91% |
| [14] | Word2Vec | SVM | Roman Urdu | The accuracy of the model is 76% |
| [15] | Word2Vec | CNN, KNN | Handwritten Numerals | The accuracy of the model is 98.45% |
| [16] | Word2Vec | CNN | Letter Detection | The accuracy of the model is 91% |
| [17] | Word2Vec | SVM | English Text | The accuracy of the model is 74% |
| [18] | Word2Vec | Deep neural network model | Panjabi | The accuracy of the model is 83% |
| [19] | Word2Vec | k-NN, Decision tree, SVM | Urdu | The accuracy of the model is 50% |
| [20] | Word2Vec | SVM, lib, IBK | Urdu | The accuracy of the model is 67% |
| [21] | Word2Vec, GloVe | Deep neural network model | Roman Urdu to English | The accuracy of the model is 76% |

(Continued)

**Table 1 (continued)**

| Paper reference | Embedding technique | Models | Dataset Text/Language | Accuracy (%) |
|---|---|---|---|---|
| [22] | FastText | 9 ML Models | Roman Urdu | The accuracy of the model is 91% |
| [23] | BERT | Deep Learning (USA-BERT) | Roman Urdu | The Improve accuracy of the model is 26% |

## 3 Materials and Methods

This section is dedicated to elucidating two prominent deep learning models extensively utilized in sentiment analysis: Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN). In this section, the essential components of the proposed solution are delineated. Initially, a comprehensive Urdu literature dataset is created from diverse sources such as Mashriq Newspaper, Seyaq Newspaper, Urdu literature from UOP and different social media content. Subsequently, Urdu sentences undergo a preprocessing and cleaning phase, wherein extraneous text, symbols, and tokens are systematically removed. To facilitate input into the chosen deep neural networks, the cleaned dataset is then prepared for the word embedding step, which was a challenging section of the research. This involves converting texts into vectors using a word embedding method, with Word2Vec being the technique employed in the proposed model [24]. Finally, for the training phase, various machine learning models are employed, specifically the CNN model and the RNN model. These models undertake the task of sentiment classification, categorizing input sentences as positive, negative, neutral, mixed, or ambiguous as applicable.

In the Fig. 1, the data is collected from different source and make dataset. Then Urdu sentences underwent a thorough process of cleaning, normalization, and lemmatization [25,26]. Subsequently, these cleaned sentences were input into word embeddings, specifically employing Word2Vec in this study. The data vectors generated were then fed into a deep architecture for sentiment analysis to determine polarity. Finally, one of the five labels positive, negative, neutral, mixed, or ambiguous was assigned to characterize the polarity of the analyzed content.
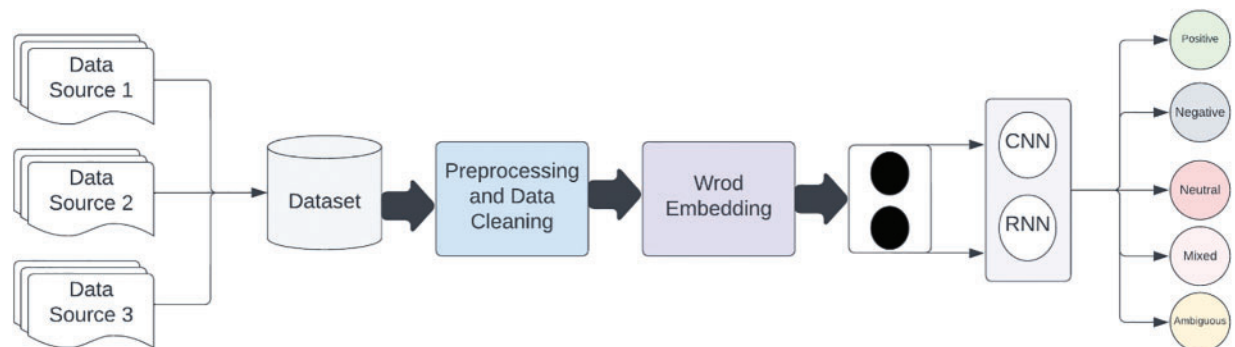


**Figure 1:** Deep architecture pipeline

### 3.1 Dataset Creation

In the realm of deep learning, the presence and the use of standard dataset wield significant influence in a proficient sentiment analysis model [27]. Prior research on sentiment analysis in Urdu, the imperative arises to construct a dataset from the ground up. The process of constructing a well-organized dataset entail navigating through several crucial steps.

### 3.2 Dataset Collection

Data collection stands as a pivotal phase in research, facilitating the capture of historical records and patterns that empower models to predict future events. Devising a well-defined strategy for data collection is paramount, especially in the creation of a dataset, involving the identification of data sources and the selection of appropriate methodologies [28]. In our research, data collection presented challenges, primarily due to the scarcity of Urdu script data. Given the prevalence of Roman Urdu in communication among Urdu speakers, a deviation from our study's target, obtaining authentic Urdu language data became a challenge. To overcome this hurdle and secure genuine Urdu language data, we focused on specific sources, namely "Mashriq Newspaper" and "Seyaq Newspaper," widely read across all provinces of Pakistan.

Additionally, we incorporated Urdu text from articles sourced from the Urdu Department at the University of Peshawar. In total, approximately 45,000 sentences were gathered to construct the dataset shown in Table 2. This dataset stands as a significant accomplishment, representing one of the initial and most extensive collections of Urdu language data tailored specifically for sentiment analysis. This comprehensive dataset serves as a crucial resource for training and evaluating the sentiment analysis model, empowering it to proficiently analyze and comprehend sentiments expressed in Urdu text.

**Table 2:** Collected dataset

| English text | Urdu text |
| --- | --- |
| Increase in regional trade, growth in exports as well ☑ 🌐 | علاقائی تجارت میں اضافہ، صادرات میں بھی بڑھوتری ☑ 🌐 |
| New government approach: Beneficial development projects for the people ☺ 👥 | حکومت کا نیا منہاج: عوام کے لئے مفید فلاحی منصوبے ☺ 👥 |
| Government initiatives for# COVID-19 prevention nationwide 🦠 🏢 | ملک بھر میں کورونا وائرس کے# بچاؤ کے لئے حکومتی اقدامات □ 🏢 |
| Improvement in social culture: Educational projects and arts businesses ✍ 🎨 | معاشرتی ثقافت میں بہتری: تعلیمی منصوبے اور ہنر کے کاروبارات 🎨 |
| Historical judiciary decision: Significant legal changes in the country ⚖ 📑 | تاریخی عدلیہ کا فیصلہ: ملک میں اہم قانونی تبدیلیاں ⚖ |
| Climate change: The impact of local variations 🌦 🌍 | موسمیاتی تبدیلی: مقامی تبدیلیوں کا اثر 🌦 □ 🌍 |
| New projects for the improvement of the education system in Country% 🏫📑 | Country میں تعلیمی نظام% کی بہتری کے لئے نئے منصوبے 🏫 |
| Rumors and reality on social media: Who can be trusted? 😷📱 | سوشل میڈیا پر افواہیں اور حقیقت: کس کو یقین ہے؟ 📱😷 |

(Continued)

**Table 2 (continued)**

| English text | Urdu text |
| --- | --- |
| Historical fame: Renowned personalities of Country! 🎉🧎 | تاریخی شہرت: ملک اکے شہرت یافتہ شخصیات 🎉🧎 |
| Go to news way: The secret of a healthy & life 🍎🧎🚶⬜ | Go to news way: ایک صحتمند 🍃زندگی کا راز 🍎 🧎 🍃 ⬜ |

### 3.3 Data Cleaning

Data cleaning is an essential step in preparing a dataset for analysis, involving the identification and handling of incorrect, incomplete, or irrelevant data. The quality of the dataset directly impacts model accuracy. In the proposed solution, data observation revealed a mixture of languages and emoticons/emojis in the collected data which is not our target data [29]. To ready the dataset for sentiment analysis, a meticulous data cleaning process was executed, including removing English and unwanted Urdu text, as well as symbols (e.g., "#$%&´() *+, -. /:;?@[\]^_`{|}~). The dataset, initially containing mixed and noisy data, was successfully refined, resulting in a focused and high-quality dataset of 38,600 sentences. This reduction in size underscores the effectiveness of the cleaning process in eliminating irrelevant content.

### 3.4 Data Preprocessing

In deep learning, data must be in numeric format. Before encoding text into numerical representations, a pivotal step known as preprocessing text is essential [30]. This involves multiple stages, such as:

1. Eliminate null values from the dataset.

2. Retain only the "sentiment" and "polarity" columns, discarding any non-required columns.

3. Remove duplicate entries within the dataset.

4. Convert values such as "positive," "negative," "neutral," "mixed," or "ambiguous" into numerical representations.

5. Tokenize the text column, creating separate lists using NLTK's sent tokenize.

6. Exclude stop words from the lists.

7. Remove non-Urdu characters from the text.

8. Utilize NLTK's word tokenize to convert the lists into tokens.

9. Split the data into training and testing sets.

Table 3 explains the tokenization, which is implemented during the data preprocessing stage, tokenization emerges as a crucial task. Tokenization, a pivotal step in Natural Language Processing (NLP), entails breaking down a text into distinct units, whether they are words, symbols, phrases, or tokens. This process serves as a foundational element for deep learning models, empowering them to proficiently handle and analyze textual data.

**Table 3:** Tokenization of dataset

| English text | Urdu text | Token words |
|---|---|---|
| Increase in regional trade, growth in exports as well | علاقائی تجارت میں اضافہ، صادرات میں بھی بڑھوتری | علاقائی، تجارت، میں، اضاف، ، صادرات , میں، بھی، بڑھوتری |
| New government approach: Beneficial development projects for the people | حکومت کا نیا منہاج: عوام کے لئے مفید فلاحی منصوبے | حکومت , کا , نیا , منہاج , عوام , کے , لئے , مفید , فلاحی , منصوبے |
| Government initiatives for# COVID-19 prevention nationwide | ملک بھر میں کورونا وائرس کے بچاؤ کے لئے کچھ نہی ہورہا | ملک , بھر , میں , کورونا , وائرس , کے , بچاؤ , کے , لئے , کچھ , نہی ، ہورہا |

### 3.5 Data Labeling

Data labeling is a vital process in categorizing information, crucial for training deep learning models to make predictions based on provided labels. In the proposed sentiment analysis, sentences were labeled to indicate positive, negative, neutral, mixed, or ambiguous sentiments. A numeric labeling scheme, where positive is 1, negative is $-1$, neutral is 0, mixed is 2, and ambiguous is $-2$, was adopted for efficient processing by deep learning models. Among 38,600 sentences, 16,212 were positive, 10,422 were negative, 6,176 were neutral, 3,474 were mixed, and 2,316 were ambiguous. Table 4 details the breakdown of labeled classes, revealing sentiment distribution, while Table 5 showcases annotated comments, providing insights into data annotation in the research.

**Table 4:** Number of label data

| Label | Number | Percentage | Label class |
|---|---|---|---|
| 1 | 16212 | 42% | Positive |
| −1 | 10422 | 27% | Negative |
| 0 | 6176 | 16% | Neutral |
| 2 | 3474 | 9% | Mixed |
| −2 | 2316 | 6% | Ambiguous |

**Table 5:** Dataset sample

| English text | Urdu text | Label data |
|---|---|---|
| Today's weather is very beautiful, everywhere is lush green and the air is very pleasant | آج کا موسم بہت خوبصورت ہے، ہر طرف ہری بھری ہے اور ہوا بہت خوشگوار ہے۔ | 1 (Positive) |
| My day is going very bad, my friends have behaved rudely with me | میرا دن بہت برا گزر رہا ہے، میرے ساتھیوں نے میرے ساتھ بے ادبی کی ہے۔ | −1 (Negative) |
| The metro journey was very ordinary, nothing special happened | میٹرو کا سفر بہت عام تھا، کچھ خاص نہیں ہوا۔ | 0 (Neutral) |
| I went to a new restaurant, the food was beautiful but the service was average | میں نئے ریستوراں گئی تھی، خوراک خوبصورت تھی لیکن خدمت متوسط تھی۔ | 2 (Mixed) |

(Continued)

**Table 5 (continued)**

| English text | Urdu text | Label data |
|---|---|---|
| This new movie seems good, some people like it but my friends are feeling somewhat mismatched | اس نئے فلم کا خیال ہے، کچھ لوگوں کو پسند آئی ہے لیکن میرے دوستوں کو کچھ کچھ غیر مطابقت محسوس ہو رہی ہے۔ | −2 (Ambiguous) |

### 3.6 Model Embedding Word to Vector (Word2Vec)

Word2Vec employs neural networks to create distributed representations of words in a way that words with similar meanings or contexts are closer together in the vector space. The training process involves adjusting the weights of the neural network to optimize the likelihood of observed word co-occurrences [31,32]. It finds applications in various domains, including sentiment classification, named-entity recognition (NER), POS-tagging, and document analysis. Word2Vec employs two methodologies: Skip gram (SG) and continuous bag of words (CBOW) [33]. In this research, CBOW was employed following exhaustive experiments, determining its superior performance over SG. Specifically, SG was trained to anticipate the context (surrounding words) of a given word, while CBOW was trained to predict the current word based on its context (given words) [34]. In this study, every review was depicted as a 2D vector with dimensions of n x 400, where n represents the number of words in the review, and 400 signifies the length of the vector's dimension for each word. This approach was implemented to maintain uniform size for all reviews, and each review's representation was padded with zeros to ensure consistent length across the dataset [35]. The methodology aligns with the approach employed in newspaper sentiment analysis using CNNs and RNN [36].

### 3.7 Model Implementation

To do Sentiment Analysis on Urdu literature data, must train a model. This model, tailored for determining the polarity of Pashtu text, is a file that undergoes a learning process utilizing a specific algorithm, enabling it to discern patterns [37]. Model creation in deep learning involves two pivotal phases: The initial phase encompasses training the model with a designated algorithm, while the subsequent phase involves testing the model. Model training in deep learning mandates a labeled dataset [38]. In this study, three deep learning algorithms—Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) were employed for model training.

The process begins with a clean and well-structured labeled dataset. The dataset is then loaded, and data preprocessing steps, including tokenization, stemming, and lemmatization using the Hazm library (a python library), are executed to ready the data for training. Prior to training, the dataset is divided into two segments: 80% for training and 20% for testing. The model is initially trained using CNN and then through RNN algorithm. This methodology empowers the model to learn from the labeled dataset and subsequently apply this acquired knowledge to analyze the sentiment of Pashtu text data.

#### 3.7.1 Sentiment Analysis Using CNN

In this section, the implementation of the proposed Urdu Sentiment Analysis model utilizing the CNN architecture is elucidated. The depicted CNN architecture, as illustrated in Fig. 2, delineates

the components of the three crucial stages within the proposed system: Dataset insertion, feature extraction, and classification [39].
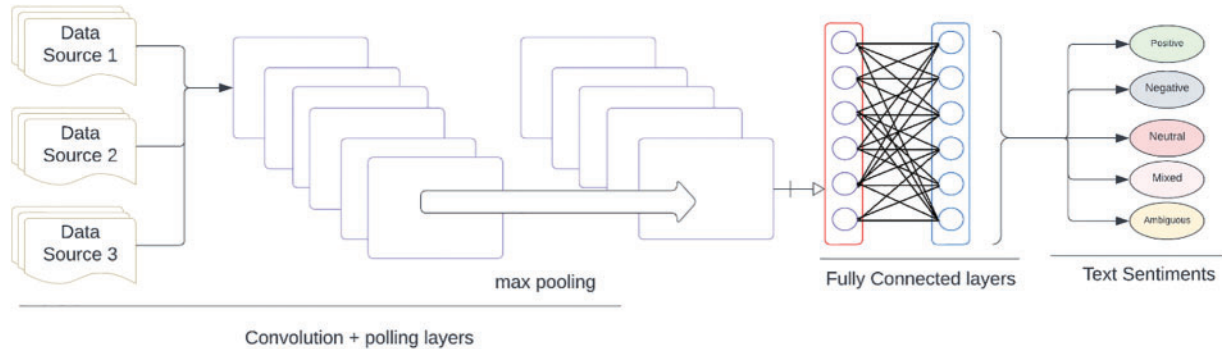


**Figure 2:** CNN architecture of proposed solution

The flow of the CNN model commences with an embedding layer, responsible for representing words as dense vector representations. Subsequently, a series of convolutional, normalization, max-pooling, and dropout layers are employed in sequential blocks [40]. Originally developed for image recognition, the CNN model has proven to be highly effective in Natural Language Processing (NLP) tasks, including text classification [41].

In this study, the CNN sequential model was constructed, featuring an embedding layer as the model input, one output layer, and convolutional layer blocks. Two fully connected hidden layers (FCLs) were incorporated for text classification. For the input layer word embeddings were utilized based on pre-trained Word2Vec models. ReLU activation functions were applied to both 1D convolutional layers, and batch normalization layers were included to mitigate overfitting. Max pooling 1D layers were employed to reduce the dimensionality of feature maps, with pool sizes of three and two after the first and second convolutions, respectively. The final layer consisted of two dense layers: One in the hidden layer with 1,024 neurons and another in the output layer.

The number of neurons in the output layer was configured based on the dataset's class count, using the sigmoid function for binary classification and the Softmax activation function for multi-classification.

### 3.7.2 Sentiment Analysis with RNN

In this study, the RNN algorithm takes center stage for sentiment classification in Urdu literature. What sets RNN apart is its temporal aspect, distinguishing it from other neural networks [42]. The algorithm shares an equal number of time steps with both CNN and LSTM, emphasizing its temporal sensitivity in capturing sequential dependencies within the data. This distinctive feature positions RNN as a valuable tool for analyzing the temporal dynamics inherent in Urdu text sentiment, showcasing its relevance and effectiveness in the realm of deep learning.

The Urdu sentiment analysis model, implemented through RNN models within deep neural networks, follows a systematic flow as shown in Fig. 3. Initiated with the input of data, the process includes preprocessing, word embedding using the Word2Vec algorithm, and feature extraction from the dataset through a meticulous selection process [43,44]. The subsequent phase involves the neural layer, where activated functions scrutinize the model within hidden layers. This comprehensive approach results in an output that discerns sentiment in Urdu text, categorizing it into five classes:

Positive, negative, mixed, or ambiguous. The RNN models contribute to a nuanced understanding of sentiment, showcasing the effectiveness of deep learning in analyzing Urdu text.
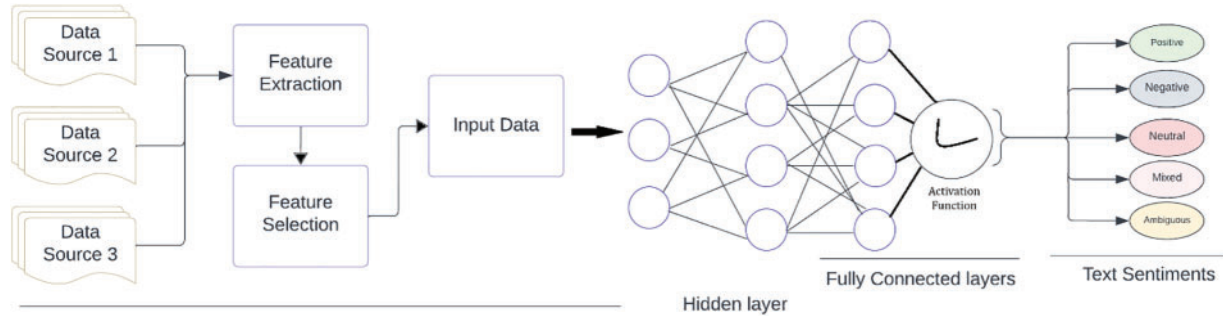


**Figure 3:** CNN architecture of proposed solution

## 4 Experimental Result

To assess the efficacy of our proposed approach in the realm of multilingual sentiment analysis, we conducted a series of experiments. This section outlines the experimental setup of our solution and subsequently introduces the evaluation of model employed in the assessment. The structured presentation aims to provide clarity on the methodology and criteria used to gauge the performance and robustness of our multilingual sentiment analysis approach.

### 4.1 Experimental Setup

In our experimental setup, we employed a dataset comprising Urdu newspaper content. For the implementation of machine learning tasks, we utilized the Anaconda open-source tool with the Python language. The research tasks were performed using Jupyter Notebook, leveraging TensorFlow—an open-source framework that integrates Keras for implementing deep learning models [45]. Specifically, we utilized Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) architectures, deploying them with TensorFlow and taking advantage of the open-source deep learning library for Python [46].

This choice of tools and frameworks ensured a robust and flexible environment for our experimentation, encompassing training, implementation, and deployment of the sentiment analysis models.

### 4.2 Model Evaluation

The assessment of RNN and CNN models aimed to measure their performance in sentiment analysis within resource-poor languages. This involved utilizing diverse metrics and criteria to measure their accuracy in classifying sentiments within Urdu literature. Factors like accuracy, precision, recall, F1 score, and confusion matrices were considered in the evaluation process. Through a systematic analysis of these metrics, insights were gained into the performance of RNN and CNN models in sentiment capture and classification, offering a thorough assessment of their capabilities in handling the intricacies of Urdu sentiment analysis. Three distinct data sources were collected and used to create a dataset for the proposed approaches, each with varying configurations. The dataset was divided into training and testing sets, with 80% allocated for training and 20% for testing. This partitioning facilitated a comprehensive evaluation of the models' performance under different conditions and contributed to a deeper understanding of their capabilities across diverse datasets.

The main objective of this experiment was to compare the performance of CNN and RNN classifiers. Various metrics such as Accuracy, Precision, Recall, and F1 score were employed to assess these classifiers, providing a comprehensive evaluation of their performance across multiple dimensions. The equations for precision, recall, F-measure, and accuracy were sequentially presented to illustrate their calculation.

$$Accuracy = \frac{True_{Positive} + True_{negative}}{True_{Positive} + True_{negative} + False_{Positive} + False_{Negative}} \tag{1}$$

$$Precision = \frac{True_{Positive}}{True_{Positive} + False_{Positive}} \tag{2}$$

$$Recall = \frac{True_{Positive}}{True_{Positive} + False_{Negative}} \tag{3}$$

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

These equation metrics collectively offer a comprehensive evaluation of the model's performance by considering different aspects of prediction accuracy and error.

## 5  Results and Discussions

In this section, the presentation of performance evaluation results for sentiment analysis of Urdu literature using RNN and CNN of DL models is showcased. To perform sentiment analysis, the dataset is categorized into five labels: Label 1 for positive, label −1 for negative, label 0 for neutral, label 2 for mixed, and label −2 for ambiguous sentiments. Deep learning methods are employed for data training, distinguishing valid data for classifier testing post-training. NLTK utilizes various algorithms to eliminate punctuation, stop words, and spaces to identify unique words [47,48]. Sklearn employs techniques like text and Word2Vec for feature extraction, incorporating data preprocessing methods. Fig. 4 illustrates the model training evaluation on the dataset.
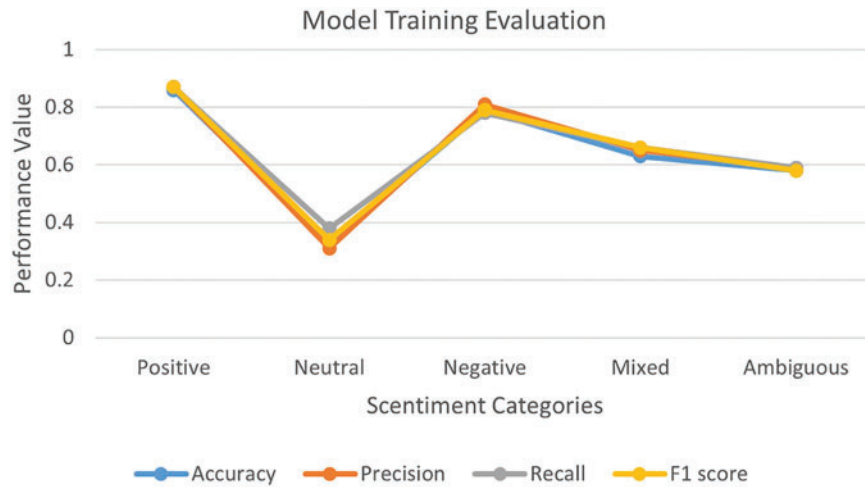


**Figure 4:** Model training evaluation

The ROC curve visually represents the performance of a classification model across various classification thresholds [49,50]. In this analysis, the model's journey is illustrated, starting from 0%

predictions, and progressing towards true positive predictions, which signify correct classifications shown in Fig. 5. The ROC curve is a graphical representation plotting the True Positive Rate (correct predictions/classifications) against the False Positive Rate (incorrect predictions/classifications) [51]. The curve begins at the 0% prediction point and moves towards the upper-left corner as true positive predictions increase. The Receiver Operating Characteristic (ROC) curve serves as a comprehensive indicator of a classification model's performance at different thresholds, providing valuable insights into its accuracy and capabilities.
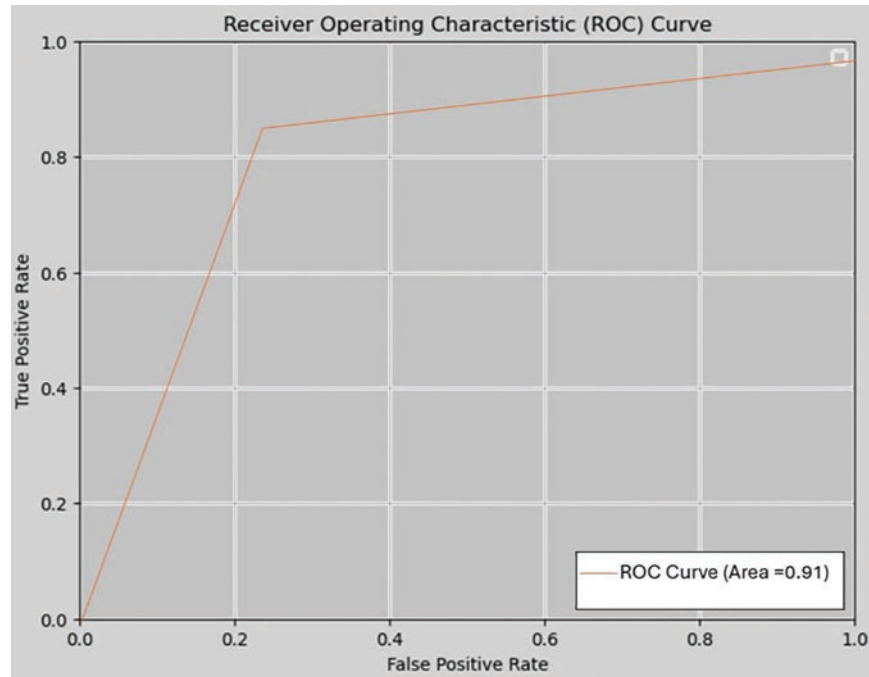


**Figure 5:** ROC curve model classification

### 5.1 Performance Measure

In this trial, we employed both CNN and RNN learning algorithms to train the model using the identical dataset. We evaluated classifier performance utilizing Accuracy, precision, recall, and F1 score, and the outcomes are presented in Tables 6 and 7, respectively.

**Table 6:** CNN performance

| CNN Model of DL | Accuracy | Precision | Recall | F1 score |
|---|---|---|---|---|
| Positive | 0.86 | 0.87 | 0.87 | 0.87 |
| Neutral | 0.33 | 0.31 | 0.38 | 0.34 |
| Negative | 0.79 | 0.81 | 0.78 | 0.79 |
| Mixed | 0.63 | 0.65 | 0.66 | 0.66 |
| Ambiguous | 0.58 | 0.59 | 0.59 | 0.58 |

**Table 7:** RNN performance

| RNN Model of DL | Accuracy | Precision | Recall | F1 score |
| --- | --- | --- | --- | --- |
| Positive | 0.89 | 0.91 | 0.89 | 0.91 |
| Neutral | 0.87 | 0.89 | 0.87 | 0.89 |
| Negative | 0.89 | 0.89 | 0.84 | 0.82 |
| Mixed | 0.73 | 0.75 | 0.76 | 0.66 |
| Ambiguous | 0.58 | 0.79 | 0.78 | 0.78 |

Table 6 and Fig. 6 illustrate the performance of the CNN and RNN classifiers for the negative class. The CNN classifier achieved high precision, indicating a minimal number of false positives. It also exhibited an 87% recall rate for detecting negative sentiment, accurately classifying 87% of cases while misclassifying 13%. In the neutral class, recall exceeded precision, suggesting high sensitivity and a reduced number of false negatives. For the positive class, both precision and recall were equal, indicating that the algorithm made an equivalent number of false positive and false negative predictions. This equilibrium is advantageous, signifying that the dataset featured an even distribution of negative and positive sentiments. When precision and recall are equivalent, the F1 score also shares the same value, resulting in identical metrics for all evaluation measures.
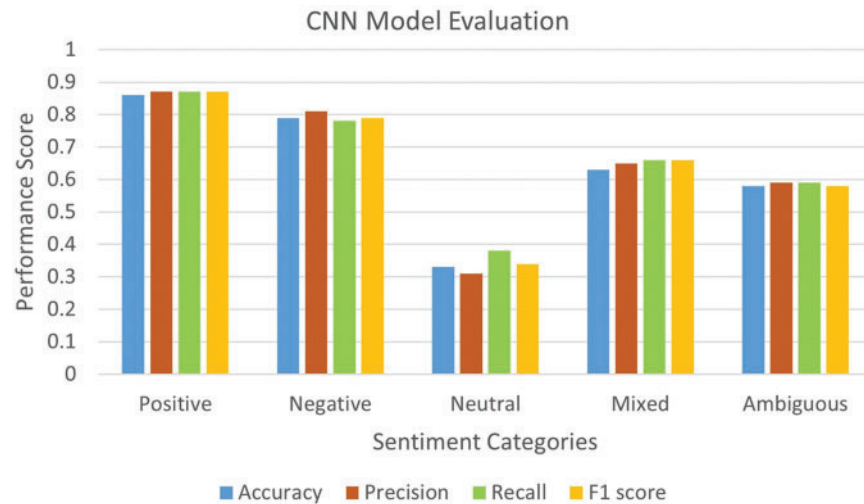


**Figure 6:** CNN evaluation

In Table 7 and Fig. 7, the RNN classifier demonstrated 89% precision in accurately identifying the negative class, with an 11% error rate. Additionally, the classifier achieved an 89% recall and accuracy rate for correctly recognizing the negative label, along with an 11% error rate. However, for neutral labels, the classifier exhibited an accuracy, precision, and recall rate of 87% and an F1 score of 89%. In contrast, the classifier performed admirably for the positive label, with 91% precision and F1 score, and 89% recall and accuracy. Overall, spanning all five classes, the RNN model consistently achieved a 91%.

Fig. 8 depicts that RNN surpassed CNN, exhibiting higher precision, recall, and F1 scores. Notably, RNN achieved a precision of 91%, outperforming CNN's 87%, implying superior precision

with fewer false positives for RNN. Additionally, RNN demonstrated a 4% higher sensitivity compared to CNN, suggesting a reduction in false negatives. Furthermore, RNN exhibited a 4% higher F1 score, indicative of superior overall performance.
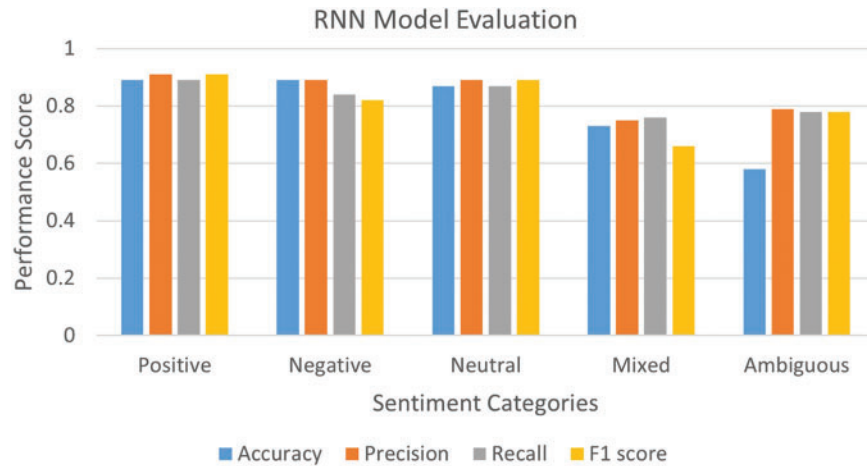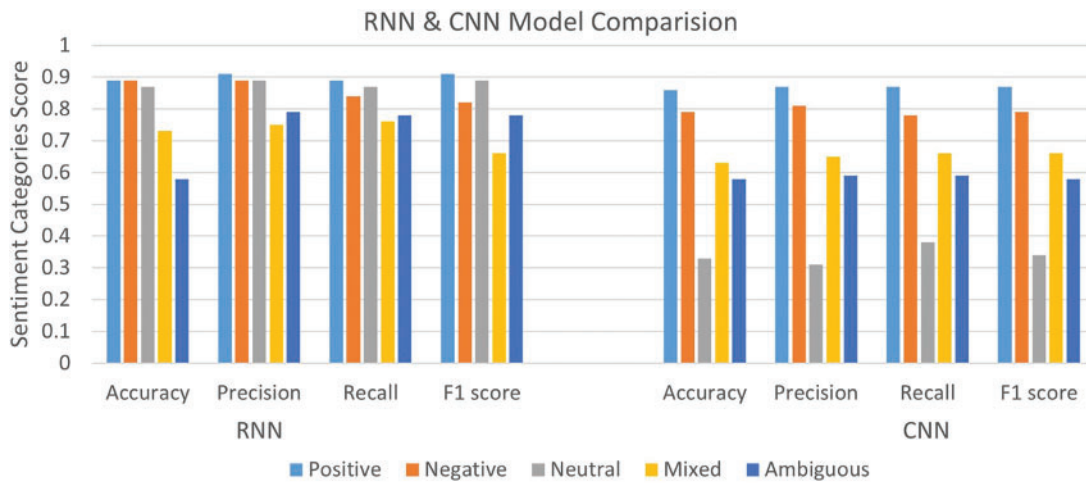


**Figure 7:** RNN evaluation



**Figure 8:** Model comparison

## 6  Conclusion

Sentiment analysis involves extracting sentiments, attitudes, emotions, and opinions from a given sentence. This research paper explores various sentiment classification techniques, employing two deep learning models, RNN and CNN, along with the Word2Vec algorithm for embedding, to conduct sentiment analysis. The paper specifically emphasizes the use of deep learning models to design a sentiment analysis model for Urdu literature, addressing the challenges posed by a resource-poor language. One significant challenge is the lack of a proper dataset for model training. In the proposed system, the initial step involves preparing a dataset collected from diverse sources, including newspapers, articles, and social media content. A total of 45,000 sentences were collected and

underwent cleaning and preprocessing, utilizing regular expressions to remove English and unwanted text, and manual deletion of some Urdu text and emoticons. Text preprocessing employed the Python library for tokenization, stemming, and lemmatization. After cleaning and preprocessing, the dataset comprised 38,600 sentences with five labels: Positive, Negative, Neutral, Mixed, and Ambiguous. Model evaluation focused on precision, recall, and F1 score, revealing that RNN outperformed CNN, achieving an accuracy approximately 0.4 times higher. In conclusion, the research suggests that RNN is a more suitable technique for sentiment analysis in Urdu literature compared to CNN.

## 7  Future Work

In the future, we are ensuring the model's evaluation is conducted with an equal distribution of labels, thereby maintaining balance in the assessment process. Also make sure the proposed methodology effectively operates across five distinct label classes, namely Positive, Neutral, Negative, Positive curse, and Negative curse. This refinement aims to broaden the applicability and robustness of the technique across a diverse range of sentiments and expressions. Furthermore, exploring innovative strategies to optimize performance within each specific label category will be a crucial aspect of this future work. Additionally, investigating the impact of these modifications on overall model accuracy and reliability will contribute valuable insights to the ongoing research.

**Author Contributions:** Conceptualization, A. Ali and M. Khan; methodology, K. Khan; formal analysis, A. Ali; investigation, Y. Y. Ghadi and R. Ullah; resources, M. Khan; data curation, K. Khan; writing—original draft preparation, A. Ali; writing—review and editing, R. Ullah and Y. Y. Ghadi; visualization, A. Ali; supervision, M. Khan and K. Khan; project administration, R. Ullah and A. Aloraini; funding acquisition, A. Aloraini and A. Ali. All authors have read and agreed to the published version of the manuscript.

**Availability of Data and Materials:** The data used for the study is currently not available for readers due to privacy and will be published separately after completion of the project.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1]  S. Chetanpal, T. Imam, S. Wibowo, and S. Grandhi, "A deep learning approach for sentiment analysis of COVID-19 reviews," *Appl. Sci.*, vol. 12, no. 8, pp. 3709, 2022. doi: 10.3390/app12083709.

[2]  A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in *Proc. 2017 3rd Int. Conf. Control, Autom. Robotics (ICCAR)*, IEEE, Apr. 2017, pp. 705–710.

[3]  L. Khan, A. Amjad, N. Ashraf, H. T. Chang, and A. Gelbukh, "Urdu sentiment analysis with deep learning methods," *IEEE Access*, vol. 9, pp. 97803–97812, 2021. doi: 10.1109/ACCESS.2021.3093078.

[4]  S. Kardakis, I. Perikos, F. Grivokostopoulou, and I. Hatzilygeroudis, "Examining attention mechanisms in deep learning models for sentiment analysis," *Appl. Sci.,* vol. 11, no. 9, pp. 3883, 2021.

[5]  A. H. Alamoodi *et al.*, "Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review," *Expert Syst. Appl.*, vol. 167, pp. 114155, 2021. doi: 10.1016/j.eswa.2020.114155.

[6]    K. K. Agustiningsih, E. Utami, and H. Al Fatta, "Sentiment analysis of COVID-19 vaccine on Twitter social media: Systematic literature review," in *2021 IEEE 5th Int. Conf. Inform. Technol., Inform. Syst. Elect. Eng. (ICITISEE)*, IEEE, Nov. 2021, pp. 121–126.

[7]    T. A. Javed, W. Shahzad, and U. Arshad, "Hierarchical text classification of Urdu news using deep neural network," arXiv preprint arXiv:2107.03141, 2021.

[8]    M. P. Akhter, J. B. Zheng, I. R. Naqvi, M. Abdelmajeed, A. Mehmood and M. T. Sadiq, "Document-level text classification using single-layer multisize filters convolutional neural network," *IEEE Access*, vol. 8, pp. 42689–42707, 2020. doi: 10.1109/ACCESS.2020.2976744.

[9]    U. Naqvi, A. Majid, and S. A. Abbas, "UTSA: Urdu text sentiment analysis using deep learning methods," *IEEE Access*, vol. 9, pp. 114085–114094, 2021. doi: 10.1109/ACCESS.2021.3104308.

[10]   M. Usman, Z. Shafique, S. Ayub, and K. Malik, "Urdu text classification using majority voting," *Int. J. of Adv. Comput. Sci. Appl.*, vol. 7, no. 8, pp. 265–273, 2017.

[11]   S. Iqbal, F. Khan, H. U. Khan, T. Iqbal, and J. H. Shah, "Sentiment analysis of social media content in pashto language using deep learning algorithms," *J. Int. Technol.*, vol. 23, no. 7, pp. 1669–1677, 2022. doi: 10.53106/160792642022122307021.

[12]   A. Khattak, M. Z. Asghar, A. Saeed, I. A. Hameed, S. Asif Hassan and S. Ahmad, "A survey on sentiment analysis in Urdu: A resource-poor language," *Egypt. Inform. J.*, vol. 22, no. 1, pp. 53–74, 2021. doi: 10.1016/j.eij.2020.04.003.

[13]   F. Noor, M. Bakhtyar, and J. Baber, "Sentiment analysis in e-commerce using SVM on Roman Urdu text," in *Emerg. Technol. Comput.: Second Int. Conf., iCETiC 2019*, London, UK, Springer International Publishing, Aug. 19–20, 2019.

[14]   S. Aslamzai and S. Saad, "Pashto language stemming algorithm," *Asia Pac. J. Inf. Technol. Multimed.*, vol. 4, no. 1, pp. 25–37, 2015. doi: 10.17576/apjitm-2015-0401-03.

[15]   K. Khan *et al.*, "PHND: Pashtu handwritten numerals database and deep learning benchmark," *PLoS One*, vol. 15, no. 9, pp. e0238423, 2020. doi: 10.1371/journal.pone.0238423.

[16]   K. Khan, "Pashtu numerals recognition through convolutional neural networks," *J. Appl. Emerg. Sci.*, vol. 9, no. 2, pp. 91–92, 2019. doi: 10.36785/jaes.92338.

[17]   A. A. Janisar, H. Afzal, and G. Kumar, "Identification of HATE speech tweets in Pashto language using Machine Learning techniques," *Int. J. of Adv. Comput. Sci. Appl.*, vol. 10, no. 3, pp. 2278–3091, 2021.

[18]   J. Singh, G. Singh, R. Singh, and P. Singh, "Morphological evaluation and sentiment analysis of Punjabi text using deep learning classification," *J. King Saud Univ.-Comput. Inform. Sci.*, vol. 33, no. 5, pp. 508–517, 2021. doi: 10.1016/j.jksuci.2018.04.003.

[19]   N. Mukhtar and M. A. Khan, "Urdu sentiment analysis using supervised machine learning approach," *Int. J. Pattern Recognit.*, vol. 32, no. 2, pp. 1851001, 2018. doi: 10.1142/S0218001418510011.

[20]   N. Mukhtar, M. A. Khan, N. Chiragh, and S. Nazir, "Identification and handling of intensifiers for enhancing accuracy of Urdu sentiment analysis," *Expert. Syst.*, vol. 35, no. 6, pp. e12317, 2018. doi: 10.1111/exsy.12317.

[21]   L. Khan, A. Amjad, K. M. Afaq, and H. T. Chang, "Deep sentiment analysis using CNN-LSTM architecture of English and Roman Urdu text shared in social media," *Appl. Sci.*, vol. 12, no. 5, pp. 2694, 2022. doi: 10.3390/app12052694.

[22]   M. A. Qureshi *et al.*, "Sentiment analysis of reviews in natural language: Roman Urdu as a case study," *IEEE Access*, vol. 10, no. 3, pp. 24945–24954, 2022. doi: 10.1109/ACCESS.2022.3150172.

[23]   M. R. Ashraf, Y. Jana, Q. Umer, M. A. Jaffar, S. Chung and W. Y. Ramay, "BERT based sentiment analysis for low-resourced languages: A case study of Urdu language," *IEEE Access*, vol. 11, pp. 110245–110259, 2023. doi: 10.1109/ACCESS.2023.3322101.

[24]   T. Walkowiak, S. Datko, and H. Maciejewski, "Bag-of-words, bag-of-topics and word-to-vec based subject classification of text documents in polish–a comparative study," in *Contemporary Complex Syst. Depend.: Proc. Thirteenth Int. Conf. Depend. Complex Syst. DepCoS-RELCOMEX*, Brunów, Poland, Springer International Publishing, 2019, pp. 526–535.

[25] L. C. Yu *et al.*, "Refining word embeddings for sentiment analysis," in *Proc. 2017 Conf. Emp. Methods Natural Language Process.*, 2017, pp. 534–539.

[26] S. Sivakumar, L. S. Videla, T. R. Kumar, J. Nagaraj, S. Itnal and D. Haritha, "Review on word2vec word embedding neural net," in *Proc. 2020 Int. Conf. Smart Elect. Commun. (ICOSEC)*, IEEE, Sep. 2020, pp. 282–290.

[27] U. Khalid, A. Hussain, M. U. Arshad, W. Shahzad, and M. O. Beg, "Co-occurrences using Fasttext embeddings for word similarity tasks in Urdu," arXiv preprint arXiv:2102.10957, 2021.

[28] M. Cliche, "BB_twtr at SemEval-2017 task 4: Twitter sentiment analysis with CNNs and LSTMs," arXiv preprint arXiv:1704.06125, 2017.

[29] A. Abdulsalam, A. Alhothali, and S. Al-Ghamdi, "Detecting suicidality in Arabic Tweets using machine learning and deep learning techniques," arXiv preprint arXiv:2309.00246, 2023.

[30] S. Rani and P. Kumar, "Deep learning-based sentiment analysis using convolution neural network," *Arab. J. Sci. Eng.*, vol. 44, no. 4, pp. 3305–3314, 2019. doi: 10.1007/s13369-018-3500-z.

[31] S. Minaee, E. Azimi, and A. Abdolrashidi, "Deep-sentiment: Sentiment analysis using ensemble of CNN and bi-LSTM models," arXiv preprint arXiv:1904.04206, 2019.

[32] L. Kurniasari and A. Setyanto, "Sentiment analysis using recurrent neural network," *J. Phys. Conf. Ser.*, vol. 1471, pp. 12018, Feb. 2020, IOP Publishing.

[33] A. Patel and A. K. Tiwari, "Sentiment analysis by using recurrent neural network," in *Proc. 2nd Int. Conf. Advanced Comput. Software Eng. (ICACSE)*, Feb. 2019.

[34] N. Elhassan *et al.*, "Arabic sentiment analysis based on word embeddings and deep learning," *Computers*, vol. 12, no. 6, pp. 126, 2023. doi: 10.3390/computers12060126.

[35] N. Durrani and S. Hussain, "Urdu word segmentation," in *Human Lang. Technol.: The 2010 Annual Conf. North American Chapter Assoc. Comput. Linguist.*, Jun. 2010, pp. 528–536.

[36] H. Ghulam, F. Zeng, W. Li, and Y. Xiao, "Deep learning-based sentiment analysis for Roman Urdu text," *Procedia Comput. Sci.*, vol. 147, no. 8, pp. 131–135, 2019. doi: 10.1016/j.procs.2019.01.202.

[37] F. Hashim and M. A. Khan, "Sentence level sentiment analysis using Urdu nouns," in *Proc. Conf. Language & Technol.*, 2016, pp. 101–108.

[38] Q. Huang, R. Chen, X. Zheng, and Z. Dong, "Deep sentiment representation based on CNN and LSTM," in *Proc. 2017 Int. Conf. Green Inform. (ICGI)*, IEEE, 2017, pp. 30–33.

[39] S. Hussain, "Resources for Urdu Language Processing," in *Proc. IJCNLP 2008*, Jan. 2008, pp. 99–100.

[40] S. Iqbal, M. W. Anwar, U. I. Bajwa, and Z. Rehman, "Urdu spell checking: Reverse edit distance approach," in *Proc. 4th Workshop on South and Southeast Asian Natural Language Process.*, 2013, pp. 58–65.

[41] I. Javed, H. Afzal, A. Majeed, and B. Khan, "Towards creation of linguistic resources for bilingual sentiment analysis of twitter data," in *Natural Language Processing and Information Systems. NLDB 2014*, Montpellier, France: Springer International Publishing, 2014, vol. 19, pp. 232–236.

[42] M. Kamran Malik *et al.*, "Transliterating Urdu for a broad-coverage Urdu/Hindi LFG grammar," in *LREC 2010, Seventh Int. Conf. Language Resources Evalu.*, 2010, pp. 2921–2927.

[43] A. Khan, M. Z. Asghar, H. Ahmad, F. M. Kundi, and S. Ismail, "A rule-based sentiment classification framework for health reviews on mobile social media," *J. Med. Imag. Health. In.*, vol. 7, no. 6, pp. 1445–1453, 2017. doi: 10.1166/jmihi.2017.2208.

[44] W. Khan, A. Daud, J. A. Nasir, and T. Amjad, "Named entity dataset for Urdu named entity recognition task," *Organization*, vol. 48, pp. 282, 2016.

[45] W. Khan *et al.*, "Urdu part of speech tagging using conditional random fields," *Lang. Resour. Eval.*, vol. 53, pp. 1–32, 2018.

[46] A. Laukaitis, O. Vasilecas, R. Laukaitis, and D. Plikynas, "Semi-automatic bilingual corpus creation with zero entropy alignments," *Inform.*, vol. 22, no. 2, pp. 203–224, 2011. doi: 10.15388/Informatica.2011.323.

[47] G. S. Lehal, "A word segmentation system for handling space omission problem in Urdu script," in *23rd Int. Conf. Comput. Linguist. 2010*, Aug. 2010, pp. 43–50.

[48] D. Lindemann, "Bilingual lexicography and corpus methods. The example of German-basque as language pair," *Procedia Soc. Behav. Sci.*, vol. 95, no. 4, pp. 249–257, 2013. doi: 10.1016/j.sbspro.2013.10.645.

[49] S. L. Lo, E. Cambria, R. Chiong, and D. Cornforth, "Multilingual sentiment analysis: From formal to informal and scarce resource languages," *Artif. Intell. Rev.*, vol. 48, no. 4, pp. 499–527, 2017. doi: 10.1007/s10462-016-9508-4.

[50] M. Bilal, H. Israr, M. Shahid, and A. Khan, "Sentiment classification of Roman-Urdu opinions using naïve Bayesian, decision tree and KNN classification techniques," *J. King Saud Univ.-Comput. Inform. Sci.*, vol. 28, no. 3, pp. 330–344, 2016. doi: 10.1016/j.jksuci.2015.11.003.

[51] M. K. Malik, "Urdu named entity recognition and classification system using artificial neural network," *ACM Trans. Asian Low-Resour. Lang. Inf. Process. (TALLIP)*, vol. 17, no. 1, pp. 2, 2017.