Taylor & Francis
Taylor & Francis Group

Applied Artificial Intelligence
An International Journal

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/uaai20

Deep Sentiments Analysis for Roman Urdu Dataset Using Faster Recurrent Convolutional Neural Network Model

Arfan Ali Nagra, Khalid Alissa, Taher M. Ghazal, Saigeeta Kukunuru, Muhammad Mugees Asif & Muhammad Fawad

To cite this article: Arfan Ali Nagra, Khalid Alissa, Taher M. Ghazal, Saigeeta Kukunuru, Muhammad Mugees Asif & Muhammad Fawad (2022) Deep Sentiments Analysis for Roman Urdu Dataset Using Faster Recurrent Convolutional Neural Network Model, Applied Artificial Intelligence, 36:1, 2123094, DOI: 10.1080/08839514.2022.2123094

To link to this article: https://doi.org/10.1080/08839514.2022.2123094

Published online: 26 Sep 2022.

APPLIED ARTIFICIAL INTELLIGENCE
2022, VOL. 36, NO. 1, e2123094 (3201 pages)
https://doi.org/10.1080/08839514.2022.2123094

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Deep Sentiments Analysis for Roman Urdu Dataset Using Faster Recurrent Convolutional Neural Network Model

Arfan Ali Nagra[a], Khalid Alissa[b], Taher M. Ghazal[c,d], Saigeeta Kukunuru[e], Muhammad Mugees Asif[a], and Muhammad Fawad[f]

[a]Department of Computer Science, Garrison University, Lahore, Pakistan; [b]Networks and Communications Department, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia; [c]Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor, Malaysia; [d]School of Information Technology, Skyline University College, University City Sharjah, Sharjah, UAE; [e]Faculty of Management, Swiss Business School, Ras Al Khaima, UAE; [f]Riphah School of Computing and Innovation, Faculty of Computing, Riphah International University, Lahore, Pakistan

ABSTRACT

Urdu language is being spoken by over 64 million people and its Roman script is very popular, especially on social networking sites. Most users prefer Roman Urdu over English grammar for communication on social networking platforms such as Facebook, Twitter, Instagram and WhatsApp. For research, Urdu is a poor resource language as there are a few research papers and projects that have been carried out for the language and vocabulary enhancement in comparison to other languages especially English. A lot of research has been made in the domain of sentiment analysis in English but only a limited work has been performed on the Roman Urdu language. Sentiment analysis is the method of understanding human emotions or points of view, expressed in a textual form about a particular thing. This article proposes a deep learning model to perform data mining on emotions and attitudes of people using Roman Urdu. The main objective of the research is to evaluate sentiment analysis on Roman Urdu corpus containing RUSA-19 using faster recurrent convolutional neural network (FRCNN), RCNN, rule-based and *N*-gram model. For assessment, two series of experiments were performed on each model, binary classification (positive and negative) and tertiary classification (positive, negative, and neutral). Finally, the evaluation of the faster RCNN model is analyzed and a comparative analysis is performed for the outcomes of four models. The faster RCNN model outperformed others as the model achieves an accuracy of 91.73% for binary classification and 89.94% for tertiary classification.

ARTICLE HISTORY
Received 19 May 2022
Revised 11 August 2022
Accepted 2 September 2022

Introduction

As technology expands rapidly in the recent years, now every individual is using internet which provide the most significant way for e-shopping, e-learning,

CONTACT Arfan Ali Nagra arfan137nagra@gmail.com Department of Computer Science, Garrison University, Lahore 94777, Pakistan

This article has been republished with a minor change. This change does not impact on the academic content of the article.

telemedicine and provide a social platform for communication and interactions between different people by using different social networking sites like Facebook, Twitter, Instagram, Blogs and many other that allow people to engage in discussion group and express their opinions with anonymity (Al-Smadi et al. 2019; Lytos et al. 2019; Vuong et al. 2019). While dealing with social media, sentiment analysis is the most important task to understand the behavior and attitude of an individual toward a particular problem and get to know what the people actually think (Sailunaz and Alhajj 2019). Sentiment analysis analyze the emotions of the users behind text written by the user by using different machine learning technique like Natural Language Processing (NLP), text analytics, computational linguistics, and so on (Ananiadou, Thompson, and Nawaz 2013; Xing, Pallucchini, and Cambria 2019).

Due to growing interaction of people with social networking sites, companies and organization are more interested in sentiment analysis to deploy winning marketing strategies, to target more customers, overcome their weaknesses and to compete the market (Luo, Huang, and Zhu 2019). Companies are putting their efforts to understand what the individual consumer thinks about their product by extracting the emotions behind their words. Similarly, politicians are interested in what the public talk about them, their reputation, what media say about them, and so on, and this media text information is needed to calculate the probability to know who is going to win or lose the coming election. We can apply sentiment analysis on many other topics like sports, entertainment, medicine and harassment, and so on (Araque, Zhu, and Iglesias 2019; Jahangir et al. 2017). Sentiment analysis become popular now a days because of using advance methods of NLP, data mining, data analysis, predictive study and contextual analysis of a document become a rising research area (Al-Ayyoub et al. 2019).

European languages mainly English is considered as a rich language when we talk about the available of the resource and tools when desired to perform sentiment analysis, but there are several other languages that are consider poor resource language and unfortunately Roman Urdu is one of them. Pakistan national language is Urdu and Urdu is spoken and understood in five different areas of South Asia. So, it also requires standard corpora to perform any task related to NLP, but it is very difficult morphologically because the spoken of Urdu and Hindi are very similar and can be interpreted by the people of both the nations but their writing scripts are different. So, instead of using Urdu script, Roman Urdu (writing Urdu script using English alphabets) is preferred because the South Asian users (like Pakistan, India and other South Asian) usually interact on social media by using Roman Urdu scripts. For example, the Roman Urdu script for "what are you doing" goes as "ap kiya kr rahy hn." Hence, the sentiment analysis for the Roman Urdu language is also very important because it helps non Urdu speaking users to understand the emotions of the Urdu speaking users behind text. And it is very helpful for the companies to expand their market at international level by understanding the sentiments of the international consumers. Sentiment analysis for Roman Urdu is important if South Asian market is needed to be target, because majority of the Urdu speakers express their emotions, ideas, thoughts and opinion by using Roman Urdu scripts. And we can see it on Facebook, Twitter and many other websites.

The proposed method is a novel model that can perform sentiment analysis for a language script which have high diversity in writing, Roman Urdu. The main target of this study is to achieve better accuracy by consuming less computational time. For this purpose, RUSA-19 Roman Urdu corpus is used, which is a collection of 10,021 sentences extracted from five different topics: drama, technology, food & recipes, politics and sports. It's a human annotator that is annotated and the evaluation is divided into three different classification groups: positive, negative and neutral.This paper is ordered into six different sections. In the next section, we will study some previous research related to sentiment analysis most importantly for Urdu language. In the third section, we will discuss more about RUSA-19 Roman Urdu corpus. Section four, we mainly concentrated on the methodology. Section five, we discusses the experimental results of our model designed to perform sentiment analysis of Roman Urdu text classification models. In Section 6 we will perform a comparative analysis and finally in Section 7 we will conclude the paper.

Literature Review

In 2019, the class imbalance problem is addressed. This problem in encountered while using the Machine Learning algorithms, it divides the classes into 2 categories one is a minor class (it contains fewer instances) and the other is a major class (it contains more instances). In some cases, an imbalance class problem affects the accuracy and sometimes it improves the results. In this paper cluster-based under-sampling technique is used, in this approach instances of the major class are converted into minor class, and they used 5 different clustering methods with the combination of ML and classification algorithms and 3 ensemble learning models. And the results show that the combination of these 4 techniques improved the predictive performance of the model (Onan 2022). In 2016, the Auto-Keyword extraction problem is discussed and it's an important domain of research in NLP. It helps to extract important features or words from a text document and provide precise information about the document. In this paper performance of five different models (Text-Rank algorithm, most frequent, co-occurrence, term frequency and eccentricity keyword extraction model) are evaluated with the help of ensemble and classification algorithms. According to the results, the conjunction of most-frequent keyword extraction and ensemble of Random Forest outperformed and achieved the highest accuracy of 94% for the ACM data set

(Onan 2019). In 2019, the topic extraction problem is resolved, in the domain of data mining topic extraction plays an important role in the organization of the collected text. In this study, they proposed a two-stage topic extraction model (word embedding and cluster analysis). The conjunction of these 2 models was applied to the 160,421 abstracts with different genre like medical, data science, Artificial Intelligence, and so on. This model predicted the genre against each abstract and based on the results it is observed that the proposed model enhanced the accuracy and performance of the clustering algorithms (Onan, Korukoğlu, and Bulut 2016).

In 2015, research was carried out to compress the data obtained from social web mining by extracting important features to reduce processing time and improve the performance of the model. In this paper, they introduced the ensemble feature selection method to address this problem, they used different feature selection techniques and performed aggregation on each extracted feature list. For aggregation, they used a genetic algorithm. For the evaluation, k-mean cluster and Naive Bayes, algorithms are used, and the evaluation results show that the proposed technique improves the performance and efficiency of the genetic algorithm, and achieved 95% accuracy for the Naive Bayes classifier with the conjunction of the genetic algorithm (Onan 2019). In 2017, the extraction of subjective information from a robust data set is quite difficult to perform, but this information plays a key role in the performance of language processing models. In this study, they tried to overcome that problem by introducing the hybrid ensemble pruning technique with the combination of a clustering algorithm and randomized search. First, the classifier cluster the data based on their predictive features and then applied classification algorithms to each cluster, which ultimately reduced the search space. For the evaluation, they compared the proposed model with 3 different ensemble models and proves that their model outperformed the remaining models in terms of performance and accuracy (Onan and Korukoğlu 2016). In 2020, an article was published that addressed sentiment analysis problem, and they combined different models like TF-IDF with CNN-LSTM model. This model comprises 5 different layers, and each layer playing an important role in better performance of the model, and these layers are weighted word embedding, CNN, max pooling and dense layer. For the model evaluation, different performance predictive techniques like GloVe, word2Vec, LDA2vec, fastText and DOC2vec were used. And they have observed that the combination of CNN and LSTM worked really good together, and they have achieved higher accuracy against their proposed model, which is 94% and its a huge achievements (Onan, Korukoğlu, and Bulut 2017).

In 2019, sentiment analysis was performed on student evaluations of teaching (SET) review dataset to improve their quality of education. They have used Recurrent Neural Network (RNN) model for sentiment analysis, with the conjunction of Convolutional ML algorithms, DL algorithms, and ensemble learning methods, and they have achieved 98% accuracy as a result (Onan 2020). In 2020, massive open online courses (MOOC) review dataset were to understand the sentiments on the MOOC users. They have performed a comparative analysis between deep learning and ensemble learning models to present an efficient model in term of productivity and accuracy. And as a result they have achieved 95% accuracy for LSTM model with the conjunction of GolVe word embedding scheme (Onan 2019). In 2021, another study was carried out to identify sarcasm in the text document and they have introduced a weighted word embedding technique using tri-grams, to assign higher values to important words without losing word order. For this purpose, three layers of bi-directional LSTM were used for sarcastic text identification. To evaluate this model sarcasm text corpus was used with the help of fastText, word2Vec, and GloVe model. Term frequency and TF-IDF weighting functions were used for unsupervised learning. And they have achieved 95% accuracy for classification (Onan 2020). In 2016, for efficient text genre, classification problem is discussed, and proposed a technique through which they can overcome this problem. And they have used an ensemble learning technique that combines the output of different classifiers to achieve a better classification, for its evaluation comparative analysis is performed among different feature extraction techniques like linguistic features, *n*-gram feature extraction, and authorship attribution with the help of 5 different learning algorithms like logistic regression (LR), support vector machines (SVM), Naive Bayes, Random Forest, k-mean cluster using LFA corpus. And they have achieved high predictive performance against their proposed scheme, that is 94% (Onan 2016).

In 2019, research was carried out to handle sarcasm sentences while performing sentiment analysis. It's a type of non-literal language, which is used to express negative emotions using positive literal meaning words. This study was mainly designed to improve the quality or performance of sentiment classification models by incorporating sarcasm sentences. For this purpose deep learning techniques like Word2vec, LDA2vec, GloVe word embedding and FastText were used. They have achieved very good accuracies and it was observed that LDA2Vec outperformed as compared to other embedding techniques and they have achieved 75–85% accuracy for subset number 1 to 6 (Onan and Tocoglu 2021). In the domain of text mining, another research was carried out in 2018, and they proposed a multi-functional classifier based n the swarm intelligence optimization (SO) technique. This model was mainly introduced to overcome the limitations of LDA (Latent Dirichlet Allocation), as LDA faces difficulty in handling high dimensional data. A hybrid approach was proposed that can handle diversity and multi-class clustering. The ensemble was combined to manage diversity among the classifiers and employed the SI algorithm on the combined diversity matrix. For the evaluation of SO-LDA different algorithms (GA, PSO, CSA) were used and it was observed that SO-

LDA performed very well and efficiently overcome the limitations of the previous model (Onan 2022). In 2016, sentiment analysis is performed on the micro blogging data set. Micro blogging data is collected from social networking sites it could be a short post, comment or message, and so on. Many techniques have been implemented in the literature that performed sentiment analysis on these micro blogging websites, but the results were not that satisfactory. In this study, they tried to achieve better accuracies in term of classification and topic identification, and they proposed a multi-model joint sentiment topic (MJST) with the conjunction of LDA that deeply analyze and classify based on the hidden topic. This technique not only classifies the text but also defines the personality of the author based on the writing. The experimental results prove the outstanding performance of the model in terms of accuracy (Onan 2018).

In 2016, sentiment analysis is performed on the micro blogging data set. Micro blogging data is collected from social networking sites it could be a short post, comment or message, and so on. Many techniques have been implemented in the literature that performed sentiment analysis on these micro blogging websites, but the results were not that satisfactory. In this study, they tried to achieve better accuracies in term of classification and topic identification, and they proposed a multi-model joint sentiment topic (MJST) with the conjunction of LDA that deeply analyze and classify based on the hidden topic. This technique not only classifies the text but also defines the personality of the author based on the writing. The experimental results prove the outstanding performance of the model in terms of accuracy (Huang et al. 2022). In 2021, a really interesting study was performed, in which long short-term memory (LSTM) tried to improve. In literature, LSTM played an important role in the domain of sentiment analysis and text classification, but this model is not capable enough to capture emotion modulation effects while extracting features for sentiment analysis. To overcome its limitations, a new variant of LSTM was introduced by integrating emotional intelligence (EI) and named ELSTM. This model improved the feature extraction ability of the model and effectively performed emotion modulation by capturing structure patterns in the text (Huang et al. 2021).

Brief Discussion on SemEval and Semantic Analysis

In the literature a lot of work has been done in the domain of sentiment analysis and a series of SemEval corpora compilations are the most prominent attempt of literature that mainly work in the field of semantic analysis. For this purpose, huge corpora have been generated in Arabic and English language (Al-Ayyoub et al. 2019). These corpora are mostly based on reviews, tweets, comments, messages, entertainment, technical products, food and restaurants. With time, the size of SemEval corpora increases by introducing versions of garner like 2013 the data-sets based on Twitter (split into 9728 training, 1654 development, and 3813 testing data-set) and 2093 SMS (sentenced based text). In 2014, version is based on Twitter data set with 86 bitterness tweets and a Live journal data set of about 1142. In 2016 and 2017, the data set is divided into 5 different subtasks and performed these four operations on each data set ((1) training, (2) development, (3) development testing and (4) testing). For A subtask 30,632 sized data-set was used, 17,639 for B and D, and 30,632 sized dataset was used for C and E subtask (Kiritchenko, Mohammad, and Salameh 2016; Nawaz, Thompson, and Ananiadou 2013).

The research of sentiment analysis has also been carried out in many other different languages like German, Italian, Korean and Indonesian. KOSAC is the Korean corpus based on 332 news articles. Parser1 is the German corpus created using Amazon reviews and it is a collection of 63,067 sentences (Safder and Hassan 2019). The Indonesian corpus was created using 5.3 Twitter tweets with the help of the twitter streaming API (Boland, Wira-Alam, and Messerschmidt 2013). Italian corpus is a collection of 2,648 sentences related to entertainment media like films and dramas. And the whole corpus was manually annotated and classified into five different sentiments like a negative, strong negative, neutral, positive, and strong positive (Zhang, Wang, and Liu 2018). Different methods have been used in previous work to process sentiment analysis. Both supervised (support vector machine (SVM)) and unsupervised (rule based) machine learning techniques have been applied to SemEval in 2014. It used the technique of determining the sentiment polarity of the sentence and based on that polarity score it classifies the data. If the polarity score is greater than 0, then it is positive; if it is smaller than 0, then it is negative; otherwise, it is neutral (Chen, Liu, and Chiu 2011). In version 2016 of SemEval, Gaussian Regression, Random Forest and Linear Regression techniques were used. To represent the influence of the positive emotion in a sentence, a supervised learning method is used to score between 0 and 1. Further, Spearman's rank and Kendall's rank were used to access the results (Attardi and Sartiano 2016). In the publication of SemEval 2017, two different classification methods were used long short-term memory RNN (LSTM_RNN). It can study long-term dependencies and uses word indexes as input chains and machine-based learning used for feature representation by embedding words. Embedding is an advanced NLP system that relates and represents the phrases or words in a real number vector. For separating the data-set into classes SVM is used. Further, random forest technique is used to produce decision trees (El-Beltagy, Kalamawy, and Soliman 2017).

Turney's approach is another unsupervised technique called Thumbs up/ thumbs down is used to perform semantic analysis. This method deals with adverbs and adjectives in the sentences and then further classified based on the word's polarity (Chen, Liu, and Chiu 2011). Artificial Neural networks (ANN) also play an important role in the research of semantic analysis, ANN is also a retrieval technique, use to retrieve semantic information from the data-set and use to separate the data-set into positive, negative, and neutral or fuzzy classes (Naseem and Hussain 2007). Gatti and Dos Santos suggested a technique based on a deep Convolutional Neural Network (deep CNN), this method performed semantic analysis on SemEval corpus by using the character to sentence-level information (Al-Amin, Islam, and Uzzal 2017). In 2017, three classifier techniques have been used CNN, logistic regression (LR) and multi-layer perceptron model; this technique was implemented on the Twitter data-set which classifies the data based on message topics and message quantification (Attardi and Sartiano 2016). In Feb 2020, Zainab and Iqra performed sentiment analysis on a manually generated Roman Urdu dataset and named it RUSA-19. They applied recurrent convolutional neural network (RCNN) model, rule-based approach, and Ngram model and they achieve very good accuracy. And we will discuss this in detail in the comparative analysis section. In 2020, another research was carried out by A. Arwa and B. Reem to investigate people's opinion and utilize it to make better decisions. For this purpose, they used a Twitter dataset and applied machine learning algorithms to classify balanced and unbalanced datasets by using different classification algorithms. And it was observed that the performance of the model is directly proportional to the number of classes, the performance of the model decreases with the increase in the number of classes. And they have achieved very good accuracies for the balanced dataset by using Naïve Byes and ID3 models and for unbalanced datasets, classifiers like Random Forest, K-mean classifier, Decision and Random Tree performed very well (El-Beltagy, Kalamawy, and Soliman 2017). In 2011, another study was carried out to investigate Twitter user's demographic information by using Twitter dataset. And the dataset consists of 184,000 Twitter users with labeled gender. And the dataset consists of 4 fields (text of the tweets, full name, screen name, and description). After applying classification algorithms, they have achieved 92% accuracy (Shamsi et al. 2021).

Semantic Analysis at Linguistic Level

Based on the studies performed at the linguistic level to work on sentiment analysis, many poor-resource languages (like Hindi, Urdu, Arabic and Thai language) take part in this research. A Hindi annotated database has been created by using the polarity technique. With the help of this algorithm, they achieved 80.21% accuracy (Burger et al. 2011). That is also considered an under-resourced language; in 2017, they proposed a machine learning technique to detect abusive words in the Thai language and they achieve 86.01% F-measure. Research in sentiment analysis based on the Bengali language was carried out on the Bengali comment data-set, and they used the Word2Vec technique to achieve 75.5% accuracy (Naseem and Hussain 2007). In 2019, a machine learning approach was conducted for the Arabic dataset to perform sentiment analysis, and the dataset consists of more than 151,000 Arabic tweets labeled into two classes positive and negative. Algorithms like SVM, AdaBoost, Naive Bayes (NB) and Round Robin (RR) were used, a comparative analysis was performed between all of them and it is observed that RR outperformed in terms of accuracy if we compare it with other models, and AdaBoost is the least performer and achieved 80% accuracy (Al-Amin, Islam, and Uzzal 2017).

A study was executed by M. Nabel in 2015, to perform Sentiment Analysis of the Arabic dataset ASTD, which consists of more than 94,000 Arabic tweets. And sentiment analysis is performed with help of the SVM model and they have achieved 81% accuracy (Gamal et al. 2019). In 2016, sentiment analysis was performed based on entertainment media mainly movie reviews. They used a dataset of more than 21,100 tweets of movie reviews. Date divided to perform training and testing then preprocessing techniques were applied and achieved 75% accuracy for SVM and 65% accuracy for NB model (Nabil, Aly, and Atiya 2015). In 2013, sentiment analysis and subjectivity analysis were performed to analyze the French tweets dataset to predict the French CAC40 stock market. The dataset consists of 1000 positive and negative French book reviews. For further processing, corpus is split into ¾ for training and testing and achieved 80% accuracy by applying the Neural Network model (Amolik et al. 2016).

Research Gap and Limitations

Based on the literature review, none of the poor resource languages (like Hindi, Khmer, Arabic and Thai language) implemented the combinations of different variants of the Neural Network (NN) algorithm for task classification. Therefore, it is vital to work on this approach, especially to test poor resource language as we have done in our work. To achieve better performance in terms of accuracy, precision, recall and F measure was performed in minimum possible computational time.

Methodology

In this section, the proposed method of deep learning models of faster RCNN, rule based and n gram methods on the data-set RUSA-19 has been implemented.

Faster Recurrent Convolutional Neural Network (Fast RCNN)

The faster RCNN was mainly introduced to overcome the limitations of the traditional neural network models like RNN, CNN, and RCNN model. The

RNN model examines the text token by token, extracts the text-based information from the given data, and stores them in the hidden layer. But the limitations of this model are that it only considers the most frequently used word of the sentence and generates a result based on these words, and it changes the overall sentiment of the text. To resolve that issue, CNN was presented with a max-pooling layer, and the RCNN model came into being as a hybrid of RNN and CNN model.

The polling layer of CNN works as a selector the select the important words from the text or sentence. But CNN uses a fix-sized search window which affects the learning speed of the CNN model. RCNN is considered as a computationally expensive model to train or test the data. To overcome the drawbacks of the RCNN model, the Fast RCNN model was introduced. But both (RCNN and Fast RCNN) use selective search techniques to find out region proposals, and as we know selective search is a time-consuming technique and we want better performance of the network.

A new model called faster RCNN was introduced to overcome the limitations of the previous models. This model eliminate the selective search algorithm and established a separate network for learning and prediction, and named it region proposal network (RPN), where regression and classification algorithms are embedded.

Proposed Model Infrastructure

The proposed model infrastructure consists of the following modules:

Input

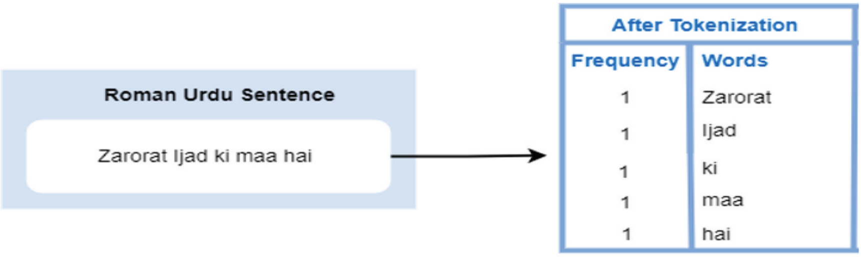
RUSA-19 corpus is used as an input data set. It consists of Roman Urdu sentences collected from five different genres like entertainment media, sports, foods, medicine and technology.

Pre-Processing

Input data is pre-processed using different techniques like:

Tokenization. Tokenization is a process of dividing the sentence into words or tokens with the count of their frequencies. As we can see in the Figure 1, we have a sentence “Zarorat ijad ki maa hai” and in the tokenization process whole sentence is divided into words with their frequency counts, here we have total of five words and every word have frequency count is 1, means no word is repeated.

Lower Casing. Lower casing is important because it helps to handle the data properly and remove the complexity. Here in the example we have “Z” and “I”



Frequency	Words
1	Zarorat
1	Ijad
1	ki
1	maa
1	hai

Figure 1. Tokenization example.

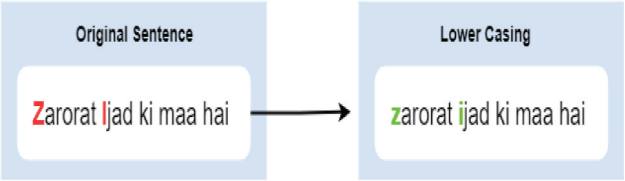


Figure 2. Lower casing example.

is in caps in the original sentence so after applying lower casing technique all words will be in small figure 2.

Stop Words. This method reduces the data size by removing the useless words from the text and only considering the meaningful words as we can see in the figure it removes "hui, k, to,e" from the sentence. It is important to note that the meaning of the sentence should not be changed after removing the useless words (Figure 3).

RNN (Recurrent Neural Network)

RNN is a bidirectional model and it is famous for of its forward and backward movement. Because of its bi-directional functionality it produce some result than take feedback and reproduce an improved result. We can say that RNN is a self-learning model and its accuracy increase with the increase in the number of recurrent steps (Figure 4).

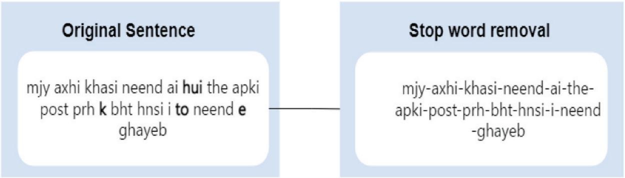

Original Sentence
mjy axhi khasi neend ai **hui** the apki post prh **k** bht hnsi i **to** neend **e** ghayeb
Stop word removal
mjy-axhi-khasi-neend-ai-the-apki-post-prh-bht-hnsi-i-neend-ghayeb

Figure 3. Stop word example.

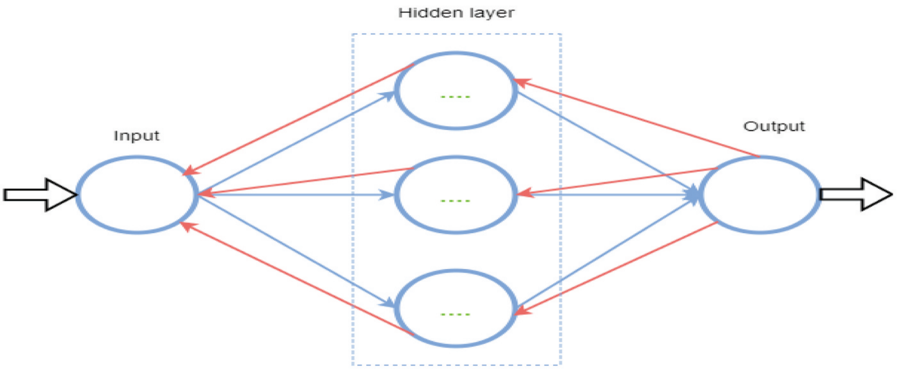


Figure 4. RNN (recurrent neural network) Model.

RNN Feature Map

Feature map extract the meaningful features out of the text for which further become the input of RPN model.

RPN

RPN is an important module of faster RCNN model because it plays an important role in term of consuming less computational power and execution time. It takes input from the Feature map and performs classification and regression operations on it and generates another feature map named as RPN based feature map.

RPN Based Feature Map

This is the new feature map generated based on the RPN output, which further becomes the input of Max pooling layer.

Max Polling Layer

Max pooling layer pol out the maximum weighted information out of the feature ma and ignore the low weighted information.

FC (Full Connection Layer)

It takes input from the Max layer and processes it in the fully connected layer, which further passes through the classification algorithms; here we are using binary and tertiary classification algorithms.

Classifier (Binary and Tertiary Classifier)

In the end, two classifiers are used binary (binary deals with two classes positive and negative) and tertiary (binary deals with three classes positive, negative and neutral). Comparative analysis is also performed to analyze the performance of these two classifiers (Figure 5).

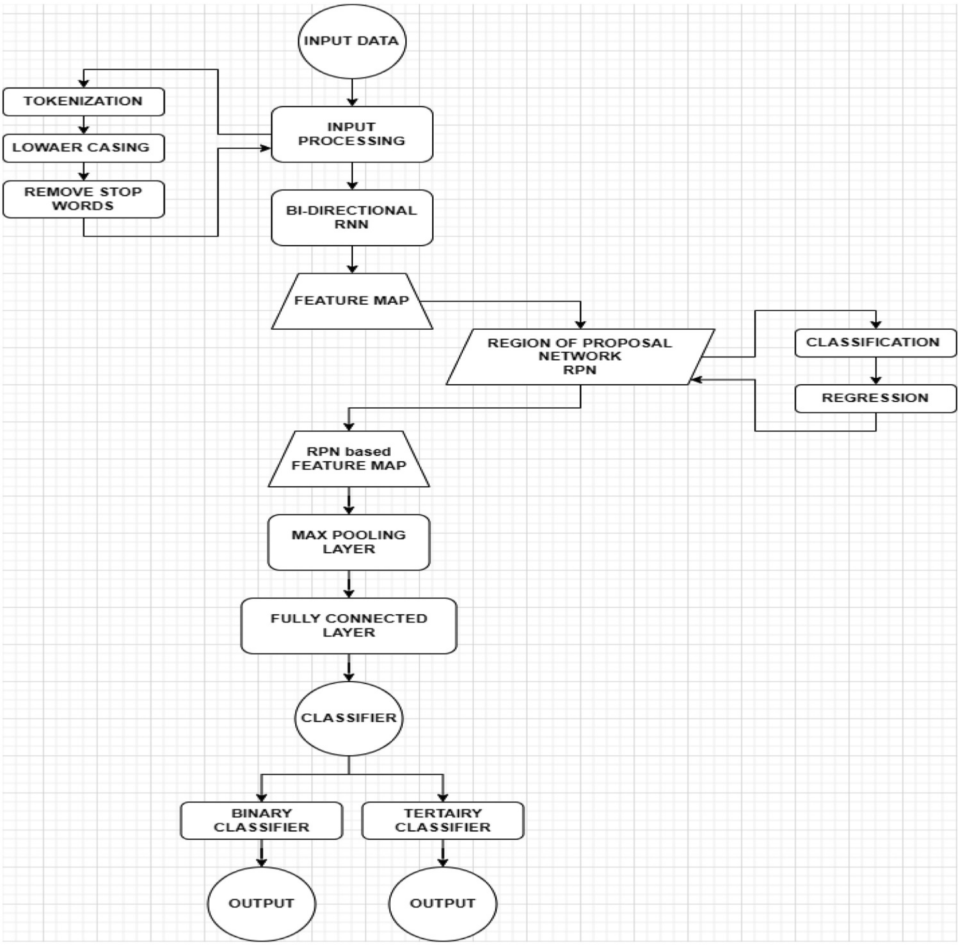


Figure 5. Architecture diagram of faster RCNN.

And generate a feature map out of it that extracts the feature and sends it to RPN (it plays an important role in the faster RCNN model, it is based on classification and regression algorithms) and it chooses selective data out of it or resized it. And its output becomes the input of the featured map, this map extracts features of the input data. Further Max pooling layer is applied to it, and it deals with mapping and pooling. Max pooling is used for utilizing a single feature map for all the proposals generated by RPN in a single pass.

$$c\_l(wi) = f_{act}\left(\left(W^{l}\right)c\_l(w_{i-1}) + \left(W^{sem\_l}\right)emb(w_{i-1})\right) \tag{1}$$

$$c\_r(wi) = f_{act}((W^{r})c\_r(w_{i-1}) + (W^{sem\_r})emb(w_{i-1})) \tag{2}$$

At training time, the data is labeled into three classes positive (1) negative (2) and neutral (0) based on the conditions that we discussed in chapter 3 (3.3 Annotation process).

Equations (1) and (2) are used to perform the left context c_l(wi) and right context c_r(wi) for the word Wi. And word embedding is calculated on wi represented as emb(w_{i-1}). W^{l} represents the matrix that transforms the hidden layer to the next hidden layer. W^{sem_l} It combines the current word I semantics with c_l and is a nonlinear activation function

$$xi = [C_{\mathbf{vl}}(Wi) : emb(Wi); C_{\mathbf{vr}}(Wi)] \tag{3}$$

In Equation (3), concatenation of context vector are performed for left contextC_{vl}, word embedding emb(Wi) and right context C_{vr}.

In the next step, tanh activation function is applied to generate a semantic vector that contains important semantics of the text. And this semantic vector becomes an input of the Region Proposal Network RPN. The goal of RPN is to generate a set score of its probability of being related to a particular class.

The output of RPN is based on a features map, and the projected featured map of RPN became the input of the max pooling layer. It takes two inputs one from RNN and the other from RPN as shown in Equation. (4).

$$\mathrm{y\_i} = \mathrm{Max}\left[\tanh\left(\mathrm{W}^1 \mathrm{x_i} + \mathrm{c}^1\right) + \mathrm{RPN}\left(\mathrm{W}^1 \mathrm{x_i} + \mathrm{c}^1\right)\right] \tag{4}$$

In Equation (5), helps to compute the output layer.

$$\mathrm{Y\_i} = \left(\mathrm{W}^2 \mathrm{x_i} + \mathrm{c}^1\right) \tag{5}$$

Equation 5 is further passed in Equation 6 through a softmax function which produce results in the form of probability which is further classified in classes.

$$\mathrm{P\_i} = \frac{\mathrm{Y_k}}{\sum_{\mathrm{k}=1}^{\mathrm{n}} \mathrm{Y_k}} \tag{6}$$

3.3. Annotation Process

Different rules were designed to annotate data and the collected data are annotated after defining rules of annotation. The whole corpus is classified into three different classes, positive, negative and neutral class. The descriptions of these classes are discussed below.

Positive Class

As we are dealing with the review, so we can call positive class as a positive_review and in our model we represented positive class as 1. There are certain rules that define a particular sentence as a positive class sentence and the rules are following:

- If the sentence shows positive emotion in all aspect (Martin 2013).
- Sentence is positive, in the existence of an agreement of approval

- If the given sentence expresses both positive and neutral emotions, then it is classified as positive. Positive dominates over negative sentiment.
- Sentences with words like admiration, thank you, greetings are considered as positive.

Figure 6 represents the positive class; here a sentence used “Mubarak ho islamabad ki shandar kamiyabi k leye.” In this sentence, we have total 8 words and according to the rules out of 8 words 3 words (Mubarak, shandar, kamiyabi) belongs to positive class, and 3 (ho, ki, k, leye) are the stop words. As we can see no negative word is used in this sentence, so we can predict that the review or sentence is positive.

Negative Class

A Negative class data is also called negative_review and in our model we represented it with 2. Following are the rules that define a review as negative class review.

- If the sentence shows negative emotions is all aspect (Pontiki et al. 2016).
- If negative sentiments are more than the other sentiment then it is labeled as negative sentiments.
- A sentence is considered as negative, if un-softened disagreements are used (Javed et al. 2014).
- Terms like boycott, punishing, and judging make the verdicts negative (Javed et al. 2014)
- Negation of positive adjective make the sentence negative (Ganapathibhotla and Liu 2008), because negation in a sentence is considered as a sentence with negative emotions.

Figure 7 exemplifies the negative class, “ye team kabhi nahi jeet sakti”; in this sentence, we have total 6 words and based on the rules, “nahi” is a negative word and the word “kabhi” increases the intensity of negativity. And collectively it makes the whole sentence negative.

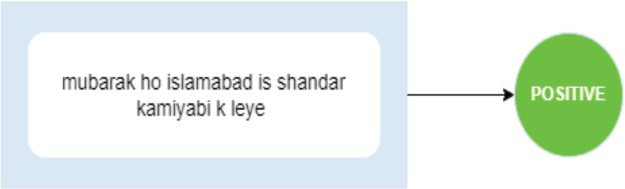


Figure 6. Positive class example.

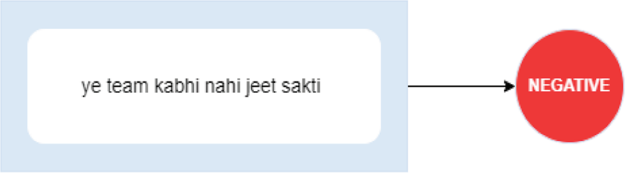


Figure 7. Negative class example.

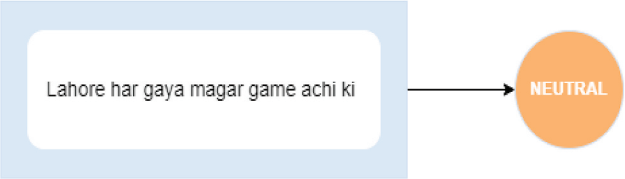


Figure 8. Neutral class example.

Neutral Class

Reviews of neutral class are also called neutral reviews, represented as 0. Following are the conditions:

- Information based on facts consider as neutral sentence (Boland, Wira-Alam and Messerschmidt 2013).
- Sentence is neutral, if it is grounded on thoughts (Turney 2002).
- A sentences with fuzziness or less surety are marked neutral sentences [3]
- Sentences with mix sentiment are considered as a neutral sentence (Lytos, Lagkas, Sarigiannidis and Bontcheva 2019).

Figure 8 is a neutral class example; here the sentence used is “Lahore har gaya magar game achi thi” this sentence consist of 7 words, “har gaya” shows negativity and “game achi” shows positivity, hence the number of positive words is equal to negative words, so they cancel out each other’s effect and make the sentence neutral.

These are the three classes, in which the whole corpus is divided. Different rules are defined to annotate the data and all the rules are taken from the literature.

Corpus Statistics

RUSA-19 corpus is divided into 9 different statistics positive, negative, neutral reviews, number of reviews, tokens, and types, reviews maximum, minimum and average length with their counts. This corpus contains of 10,021 Roman Urdu reviews from five different genres, of which positive reviews are 3778 which equals to 38% of the corpus; 2941 negative reviews equivalent to 29%; and 3302 neutral reviews equals to 33% of the corpus.

Table 1. Result obtained from previous related work.

Year	Reference	Model	Results
2022	(Onan 2022)	Bi-directional CRNN	Achieved best accuracies
2019	(Onan 2019)	Heterogeneous consensus clustering-based under sampling scheme	Achieved best predictive performance
2016	(Onan, Korukoğlu, and Bulut 2016)	Most-frequent keyword extraction and ensemble of Random Forest	94% accuracy
2019	(Onan 2019)	Two-stage topic extraction	Achieved best accuracies
2015	(Onan and Korukoğlu 2016)	Naïve Bayes and GA	95% accuracy
2017	(Onan, Korukoğlu, and Bulut 2017)	Hybrid ensemble pruning technique	Achieved best accuracies
2020	(Onan 2020)	TF-IDF with CNN-LSTM model	94% accuracy
2019	(Onan 2019)	Convolutional ML algorithms, DL algorithms, and ensemble learning methods	98% accuracy
2020	(Onan 2020)	for LSTM model with the conjunction of GolVe word embedding scheme	95% accuracy
2016	(Onan 2016)	Ensemble learning technique	94% accuracy
2021	(Onan and Tocoglu 2021)	Weighted word embedding technique using tri-grams,	95%accuracy
2019	(Onan 2022)	LDA2Vec	85% accuracy
2018	(Onan 2018)	SO-LDA	Achieved best accuracies
2016	(Huang et al. 2022)	Multi-model Joint Sentiment Topic (MJST)	Achieved best accuracies
2021	(Huang et al. 2021)	ELSTM	Achieved best accuracies

Table 2. Result obtained from previous work at the linguistic level.

Year	Reference Number	Model	Datasets	Results
2007	(Burger et al. 2011)	Polarity technique	Hindi dataset	80% accuracy
2013	(Amolik et al. 2016)	NN model	French CAC40 stock market dataset	80% accuracy
2015	(Gamal et al. 2019)	SVM	Arabic language ASTD,	81% accuracy
2016	(Nabil, Aly, and Atiya 2015)	SVM and NB model	Entertainment media mainly movie reviews	75% accuracy for SVM and 65% accuracy for NB model
2017	(Naseem and Hussain 2007)	ML	Thai language dataset	86% accuracy
2017	(Naseem and Hussain 2007)	Word2Vec technique	Bengali language	75% accuracy
2019	(Al-Amin, Islam, and Uzzal 2017)	ML	Arabic language	80% accuracy

Table 1 shows the results obtained from previous related work and Table 2 shows the results obtaied from previous work at the linguistic level. As shown in Table 3, the number of tokens in our corpus is 1,46,558 for 10,021 reviews and out of 1,46,558 tokens, we have 21,750 unique words named as “Types.” Minimum length of a review sentences is 1 word, Average length is 15 words and maximum length is 154 words.

Initial Conditions

For sentiment analysis, standard parameters were designed to setup this model and used as an initial parameter and these are following:

Table 3. Corpus Statistics.

Positive_reviews	3,778
Negative_reviews	2,941
Neutral_reviews	3,302
Number of reviews	10,021
Number of tokens	1,46,558
Number of types	21,750
Reviews minimum length	1 words
Reviews maximum length	154 words
Reviews average length	15 words
Positive_reviews	3,778

Table 4. Faster RCNN model initial parameters.

Word embedding	50
Size of hidden layer	1000
Context vector size	1000
Epochs	100
Gradient descent	.05

Evaluation Measure

In this research, two different set of experiments (tertiary and binary classification) were performed on the same data set with the help of our suggested model faster RCNN. We implemented rule-based technique for the comparative analysis of both the experiments. For this purpose, we divide the data set, binary classification method used 6720 Roman Urdu sentences from RUSA-19 corpus and Tertiary used 10,020 sentences.

Table 4 shows the initial parameters of Faster RCNN model. Further, *n*-gram mode is used to evaluate rule-based approach It simply tokenize the words into tokens, which act as our lexicons and used as input to perform polarity detection. And the output of this model is further used to evaluate accuracy, recall, F measure and precision. The following Table 5 describes all of the special symbols that we will use for the evaluation.

$$\text{accuracy} = \frac{\text{Tp} + \text{Tn}}{\text{Tp} + \text{Tn} + \text{Fp} + \text{Fn}} \tag{7}$$

Table 5. Special symbols description used for evaluation.

Symbols	Description
Tp	True positive
Tn	True negative
Fp0	False positive
Fn	False negative

$$\text{recall} = \frac{\text{Tp}}{\text{Tp} + \text{Fn}} \tag{8}$$

$$\text{precion} = \frac{\text{Tp}}{\text{Tp} + \text{Fp}} \tag{9}$$

$$\text{F measure} = 2* \frac{\text{precison} * \text{recall}}{\text{precison} + \text{recall}} \tag{10}$$

These are the formulas of accuracy, recall, precision and F measure, which are used for evaluation of the proposed model.

Results and Discussion

This section describes the performance and effectiveness of the faster RCNN model based on the results of the experiments. And comparative analysis is performed between faster RCNN, RCNN, rule-based approach and *N*-gram model on the base of their experimental results. Same data set is used in all models; binary and tertiary classification is performed in every model.

Faster RCNN Results and Evaluation

According to Table 6, using faster RCNN we achieved 0.9173 accuracy, 0.9083 precision, 0.9124 recalls and 0.9175 F1 score for binary classification. And for tertiary classification, we achieved 0.8994 accuracy, 0.08873 precision, 0.08875 recall and 0.8987 F1 score. It is observed that, faster RCNN model outperforms the RCNN, rule-based approach and *N*-gram model. But if we compare the binary and tertiary classification results, it is observed that the performance of faster RCNN for binary classification is far better than tertiary classification. We observed that, tertiary performed poorly for all models, and there could be multiple reasons behind its poor performance. It could be because of more number of classes or it could be because of more number of sentences with longer length.

Table 6. Comparative analysis of faster RCNN model for binary and tertiary classification experiment.

Model	Data set	Experimental methods	Accuracy	Precision	Recall	F measure
Faster RCNN	RUSA-19 corpus	Binary classification	0.9173	0.9083	0.9124	0.9175
		Tertiary classification	0.8994	0.8873	0.8875	0.8987
RCNN	RUSA-19 corpus	Binary classification	0.9027	0.9002	0.9102	0.9031
		Tertiary classification	0.8971	0.8832	0.8874	0.8986
Rule-based approach	RUSA-19 corpus	Binary classification	0.5543	0.5432	0.5563	0.6281
		Tertiary classification	0.5974	0.5765	0.5981	0.6272
2-gram	RUSA-19 corpus	Binary classification	0.7231	0.7324	0.7021	0.6810
		Tertiary classification	0.7443	0.7456	0.7024	0.6962

Table 7. Comparative analysis of faster RCNN model with RCNN, rule-based approach and *N*-gram model for binary and tertiary classification.

Model	Data set	Experimental methods	Accuracy	Precision	Recall	F measure
Faster RCNN	RUSA-19 corpus	Binary classification	0.9173	0.9083	0.9124	0.9175
		Tertiary classification	0.8994	0.8873	0.8875	0.8987

Comparative Analysis

Deep comparative analysis is performed in this section, where we have compared our research model faster RCNN with previously implemented RCNN, rule-based approach and *N*-gram model on the same data set. The result of our model and remaining models are listed below in Table 7. According to Table 7, we can see that our model outperformed as compared to other models for both binary and tertiary classification.

In Table 7, we can observe that faster RCNN is performing far better than RCNN, rule-based approach and *N*-gram model, in term of accuracy, precision, recall and F1 measure.

Figure 9 shows the performance of faster RCNN, RCNN, rule-based approach and *N*-gram model for binary classification and Figure 4 shows the performance of these models for binary classification. And it is observed that the blue line which actually shows faster RCNN is greater than the remaining bars.

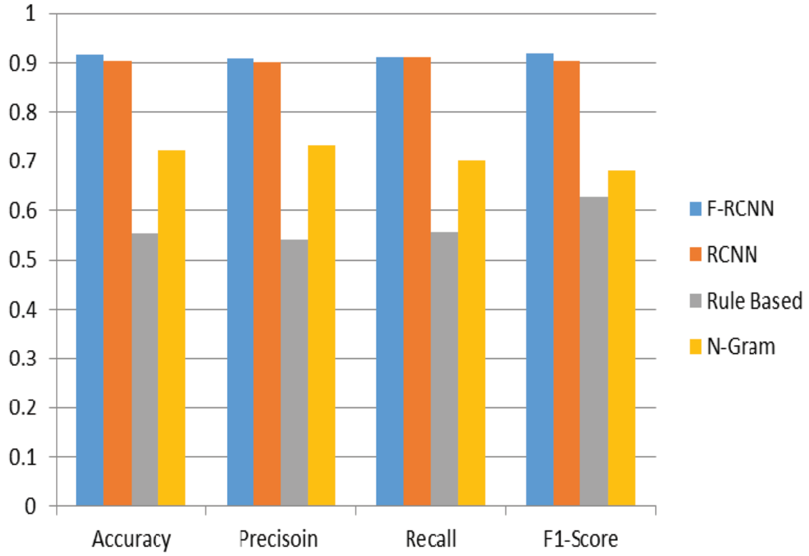


Figure 9. Assessment of Accuracy, Precision, Recall, and F1 score by binary classification for all models using RUSA-19 Corpus.

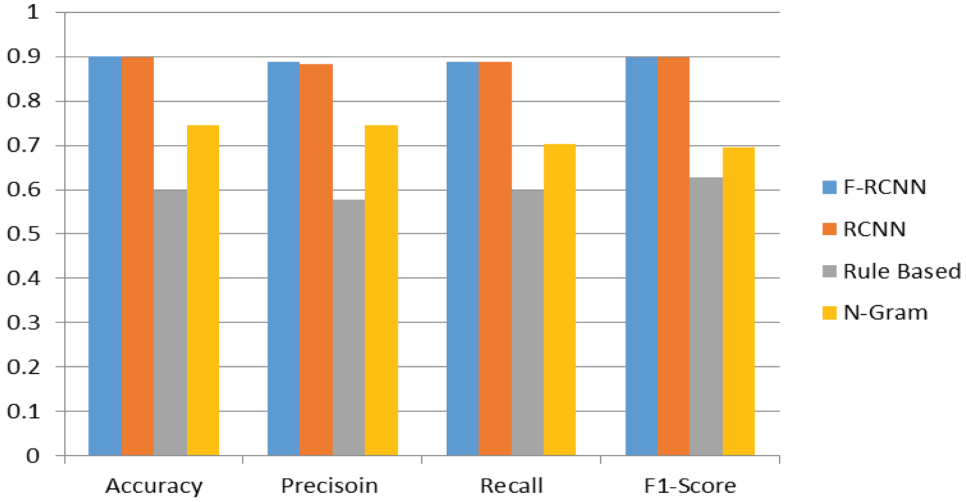


Figure 10. Assessment of accuracy, precision, recall, and F1 score by tertiary classification for all models using RUSA-19 Corpus.

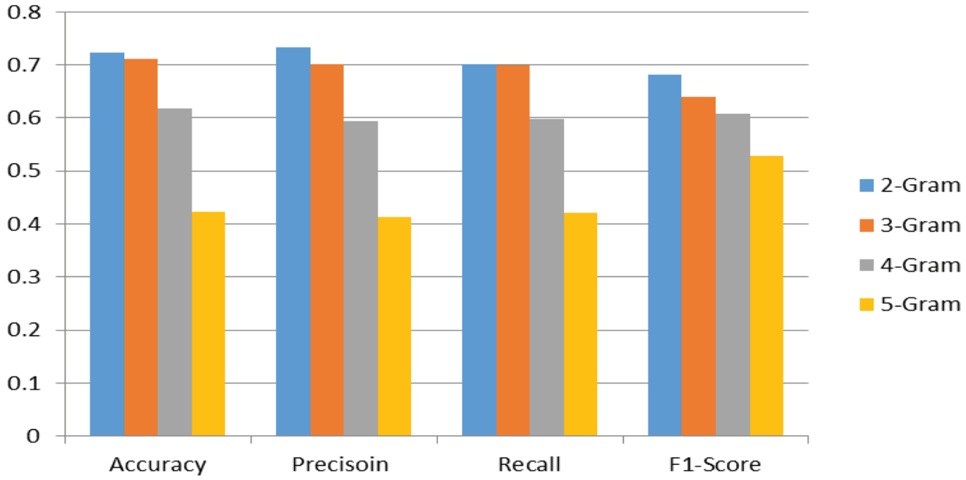


Figure 11. Comparison of precision, recall, and F1 score of binary classification for 2 to 5-gram.

Figure 10 shows the performance of faster RCNN, RCNN, rule-based approach and *N*-gram model for binary classification and Figure 4 shows the performance of these models for tertiary classification.

Figures 11 and 12 show the performance of *N*-gram model of different grams (like: 2-gram, 3-gram, 4-gram and 5-gram) for binary and binary classification.

Figure 13 shows the performance of faster RCNN model for two different classification techniques binary and tertiary classification, training accuracy between 0.4 and 0.9 is represented in Y-axis and epochs between 0 and 100 are represented on X-axis.

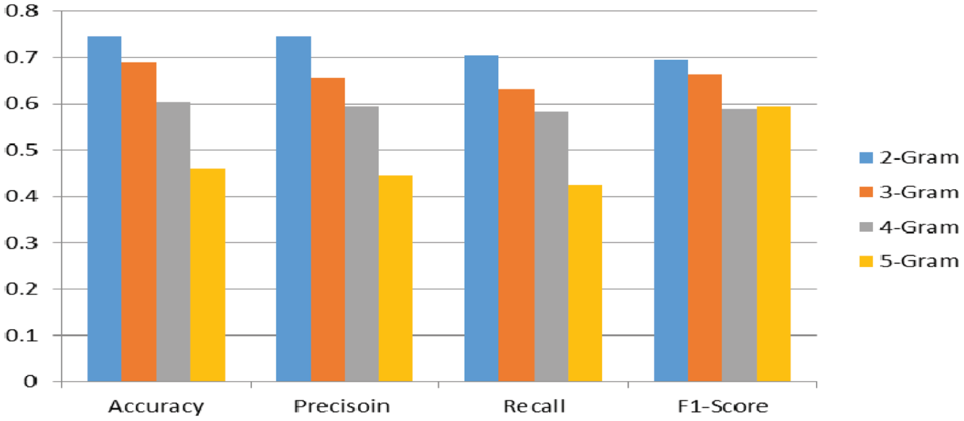


Figure 12. Comparison of Precision, recall, and F1 score of tertiary classification for 2 to 5-gram.

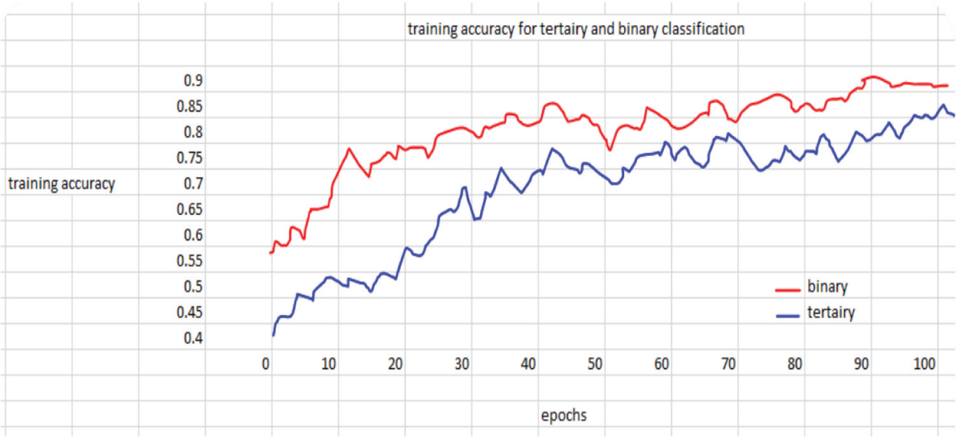


Figure 13. Faster RCNN training accuracy for binary and tertiary classification.

If we observe the graph, we can see that the training increases gradually and become stable after passing some epochs. The faster RCNN model can perform very efficiently even for a low resource language, and Roman Urdu is also a low-resource language. So because of this fact, it has the capability of avoiding noise and can extract more contextual information.

Table 8. Comparative analysis of faster RCNN model in term of speed.

Model	1 epoch	10 epoch
RCNN	51 min	510 min
Faster RCNN	10.2 min	102 min

Comparative Analysis

According to Tables 7 and 8, we can say that our proposed model outperform the previous proposed models in term of accuracy and speed. We have implemented both RCNN and faster RCNN and both the models performed very well in term of accuracy and F1 score but the processing time for both the models are different. For example, for one epoch RCNN took 51 min it means for 10 epochs this model took 51 × 10 = 510 mins and its huge time consuming. Because of this processing time consumption, we wanted to use faster RCNN, and the proposed model completed one epoch in 10.2 min and for 10 epochs it only took 10.2x10 = 102 min and it's a huge difference. So we can say that faster RCNN outperformed than RCNN in terms of speed.

Conclusion

In recent times, a number of studies have been carried out in the domain of sentiment analysis and English language is enriched with the new vocabulary. However, for Roman Urdu, the research work is very limited. It is reported that a good accuracy for a linguistic model can be achieved but it requires massive data processing techniques such as LDA, TF-IDF. These techniques also require manual crafting to extract features from the dataset. In contrast, the popular deep learning models can efficiently perform feature extraction with end to end processing and only the input data with a few pre-processing steps is required. This study opens up a new domain for further research as it builds a deep learning model which will be the language independent and work efficiently for low resource languages as well. Our research work has also highlighted the essential insights in deep learning model to process complex languages and it efficiently processes Roman Urdu as well. The only limitation is that the dataset is very complex as Roman Urdu has variety of words and different spellings for the same word (like for "what" we can use all these words; kia, kiya, kya, etc.). We have performed our analysis on a corpus of 10,021 sentences. It is a small dataset to develop a generalized model for performing sentiment analysis on a language. In future, we will extend our dataset to three or four times and employ a modified version of FRCNN that will further enhance the accuracy levels.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

Al-Amin, M., M. S. Islam, and S. D. Uzzal. 2017. Sentiment analysis of Bengali comments with word2vec and sentiment information of words. *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 186–90. IEEE.

Al-Ayyoub, M., A. A. Khamaiseh, Y. Jararweh, and M. N. Al-Kabi. 2019. A comprehensive survey of Arabic sentiment analysis. *Information Processing & Management* 56 (2):320–42. doi:10.1016/j.ipm.2018.07.006.

Al-Smadi, M., M. Al-Ayyoub, Y. Jararweh, and O. Qawasmeh. 2019. Enhancing aspect-based sentiment analysis of Arabic hotels' reviews using morphological, syntactic and semantic features. *Information Processing & Management* 56 (2):308–19. doi:10.1016/j.ipm.2018.01.006.

Amolik, A., N. Jivane, M. Bhandari, and M. Venkatesan. 2016. Twitter sentiment analysis of movie reviews using machine learning techniques. *International Journal of Engineering and Technology* 7:1–7.

Ananiadou, S., P. Thompson, and R. Nawaz 2013. Enhancing search: Events and their discourse context. *International Conference on Intelligent Text Processing and Computational Linguistics, Budapest, Hungary* 318–34. Springer.

Araque, O., G. Zhu, and C. A. Iglesias. 2019. A semantic similarity-based perspective of affect lexicons for sentiment analysis. *Knowledge-Based Systems* 165:346–59. doi:10.1016/j.knosys.2018.12.005.

Attardi, G., and D. Sartiano 2016. UniPI at SemEval-2016 task 4: Convolutional neural networks for sentiment classification. *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)* 220–24 , San Diego, California.

Ayata, D., M. Saraclar, and A. Ozgur 2017. BUSEM at SemEval-2017 task 4A sentiment analysis with word embedding and long short term memory RNN approaches. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* 777–83 , Turkey.

Boland, K., A. Wira-Alam, and R. Messerschmidt (2013). Creating an annotated corpus for sentiment analysis of German product reviews. *GESIS-Technical Reports*.

Burger, J. D., J. Henderson, G. Kim, and G. Zarrella. 2011. Discriminating gender on Twitter. Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK, 1301–09.

Chen, L. S., C. H. Liu, and H. J. Chiu. 2011. A neural network based approach for sentiment classification in the blogosphere. *Journal of Informetrics* 5 (2):313–22. doi:10.1016/j.joi.2011.01.003.

El-Beltagy, S. R., M. E. Kalamawy, and A. B. Soliman. 2017. Niletmrg at semeval-2017 task 4: Arabic sentiment analysis. *Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017)*, Arabic, 790–95.

Gamal, D., M. Alfonse, E.-S.-M. El-Horbaty, and A.-B.-M. Salem. 2019. Twitter benchmark dataset for Arabic sentiment analysis. *International Journal of Modern Education and Computer Science* 11 (1):33. doi:10.5815/ijmecs.2019.01.04.

Ganapathibhotla, M., and B. Liu 2008. Mining opinions in comparative sentences. *Proceedings of the 22nd International Conference on Computational Linguistics. 1. Proceedings of the 22nd International Conference on Computational Linguistics* 241–48. Association for Computational Linguistics.

Huang, F., X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao. 2021. Attention-Emotion-Enhanced convolutional LSTM for sentiment analysis. *IEEE Transactions on Neural Networks and Learning Systems*,Waikoloa, HI, USA

Huang, F., S. Zhang, J. Zhang, and G. Yu. 2022. Multimodal learning for topic sentiment analysis in microblogging.

Jahangir, M., H. Afzal, M. Ahmed, K. Khurshid, and R. Nawaz (2017). An expert system for diabetes prediction using auto tuned multi-layer perceptron. *2017 Intelligent Systems Conference (IntelliSys)*, London, United Kingdom, 722–28. IEEE.

Javed, I., H. Afzal, A. Majeed, and B. Khan 2014. Towards the creation of linguistic resources for bilingual sentiment analysis of twitter data. *International Conference on Applications of Natural Language to Data Bases/Information Systems* 232–36, València, Spain Springer.

Kiritchenko, S., S. Mohammad, and M. Salameh 2016. Semeval-2016 task 7: Determining the sentiment intensity of English and Arabic phrases. *Proceedings of the 10thinternational workshop on semantic evaluation (SEMEVAL-2016)*, San Diego, California, USA 42–51.

Luo, Z., S. Huang, and K. Q. Zhu. 2019. Knowledge empowered prominent aspect extraction from product reviews. *Information Processing & Management* 56 (3):408–23. doi:10.1016/j.ipm.2018.11.006.

Lytos, A., T. Lagkas, P. Sarigiannidis, and K. Bontcheva. 2019. The evolution of argumentation mining: From models to social media and emerging tools. *Information Processing & Management* 56 (6):102055. doi:10.1016/j.ipm.2019.102055.

Martin, V. 2013. Predicting the French stock market using social media analysis. Proceedings - 8th International Workshop on Semantic and Social Media Adaptation and Personalization, SMAP 2013, 3–7. doi:10.1109/SMAP.2013.22.

Nabil, M., M. Aly, and A. Atiya. 2015. Astd: Arabic sentiment tweets dataset. Proceedings of the 2015 conference on empirical methods in natural language processing, 2515–19.10.1109/SMAP.2013.22.

Naseem, T., and S. Hussain. 2007. A novel approach for ranking spelling error corrections for Urdu. *Language Resources and Evaluation* 41 (2):117–28. doi:10.1007/s10579-007-9028-6.

Nawaz, R., P. Thompson, and S. Ananiadou. 2013. Negated bio-events: Analysis and identification. *BMC Bioinformatics* 14 (1). doi:10.1186/1471-2105-.

Onan, A. 2016. An ensemble scheme based on language function analysis and feature engineering for text genre classification. *Journal of Information Science* 44 (1):28–47. doi:10.1177/0165551516677911.

Onan, A. 2018. Biomedical text categorization based on ensemble pruning and optimized topic modelling. *Computational and Mathematical Methods in Medicine* 1–22. doi:10.1155/2018/2497471.

Onan, A. 2019. Consensus clustering-based undersampling approach to imbalanced learning. *Scientific Programming* 1–14. doi:10.1155/2019/5901087.

Onan, A. 2019. Mining opinions from instructor evaluation reviews: A deep learning approach. *Computer Applications in Engineering Education* 28 (1):117–38. doi:10.1002/cae.22179.

Onan, A. 2020. Sentiment analysis on massive open online course evaluations: A text mining and deep learning approach. *Computer Applications in Engineering Education* 29 (3):572–89. doi:10.1002/cae.22253.

Onan, A. 2020. Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks. *Concurrency and Computation: Practice and Experience* 33 (23). doi:10.1002/cpe.5909.

Onan, A. 2022. Bidirectional convolutional recurrent neural network architecture with group-wise enhancement mechanism for text sentiment classification. *Journal of King Saud University - Computer and Information Sciences* 34 (5):2098–117. doi:10.1016/j.jksuci.2022.02.025.

Onan, A. 2022. Topic-Enriched word embeddings for sarcasm identification.

Onan, A., and S. Korukoğlu. 2016. A feature selection model based on genetic rank aggregation for text sentiment classification. *Journal of Information Science* 43 (1):25–38. doi:10.1177/0165551515613226.

Onan, A., S. Korukoğlu, and H. Bulut. 2016. Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications* 57:232–47. doi:10.1016/j.eswa.2016.03.045.

Onan, A., S. Korukoğlu, and H. Bulut. 2017. A hybrid ensemble pruning approach based on consensus clustering and multi-objective evolutionary algorithm for sentiment classification. *Information Processing & Management* 53 (4):814–33. doi:10.1016/j.ipm.2017.02.008.

Onan, A., and M. Tocoglu. 2021. A term weighted neural language model and stacked bidirectional LSTM based framework for Sarcasm Identification. *IEEE Access* 9:7701–22. doi:10.1109/ACCESS.2021.3049734.

Pontiki, M., D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, and A. S. Mohammad. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, San Diego, California 19–30.

Safder, I., and S. U. Hassan. 2019. Bibliometric-Enhanced information retrieval: A novel deep feature engineering approach for algorithm searching from full-text. *Scientometrics* 119 (1):257–77. doi:10.1007/s11192-019-03025-y.

Sailunaz, K., and R. Alhajj. 2019. Emotion and sentiment analysis from Twitter text. *Journal of Computational Science* 36:101003. doi:10.1016/j.jocs.2019.05.009.

Shamsi, A., A. Arwa, R. Bayari, and S. Salloum. 2021. Sentiment analysis in English texts. *Advances in Science Technology and Engineering Systems Journal* 5:1683–89. doi:10.25046/aj0506200.

Turney, P. D. 2002. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. *Proceedings of the 40th annual meeting on National Research Council of Canada*

Vuong, T., M. Saastamoinen, G. Jacucci, and T. Ruotsalo. 2019. Understanding user behavior in naturalistic information search tasks. *Journal of the Association for Information Science and Technology* 70 (11):1248–61. doi:10.1002/asi.24201.

Xing, F. Z., F. Pallucchini, and E. Cambria. 2019. Cognitive-Inspired domain adaptation of sentiment lexicons. *Information Processing & Management* 56 (3):554–64. doi:10.1016/j.ipm.2018.11.002.

Zhang, L., S. Wang, and B. Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery* 8 (4):e1253.