Social Network Analysis and Mining (2024) 14:86
https://doi.org/10.1007/s13278-024-01245-6

ORIGINAL ARTICLE

Augmenting sentiment prediction capabilities for code-mixed tweets with multilingual transformers

Ehtesham Hashmi[1] · **Sule Yildirim Yayilgan**[1] · **Sarang Shaikh**[1]

Received: 1 March 2024 / Revised: 15 March 2024 / Accepted: 17 March 2024

Abstract
People in the modern digital era are increasingly embracing social media platforms to express their concerns and emotions in the form of reviews or comments. While positive interactions within diverse communities can considerably enhance confidence, it is critical to recognize that negative comments can hurt people's reputations and well-being. Currently, individuals tend to express their thoughts in their native languages on these platforms, which is quite challenging due to potential syntactic ambiguity in these languages. Most of the research has been conducted for resource-aware languages like English. However, low-resource languages such as Urdu, Arabic, and Hindi present challenges due to limited linguistic resources, making information extraction labor-intensive. This study concentrates on code-mixed languages, including three types of text: English, Roman Urdu, and their combination. This study introduces robust transformer-based algorithms to enhance sentiment prediction in code-mixed text, which is a combination of Roman Urdu and English in the same context. Unlike conventional deep learning-based models, transformers are adept at handling syntactic ambiguity, facilitating the interpretation of semantics across various languages. We used state-of-the-art transformer-based models like Electra, code-mixed BERT (cm-BERT), and Multilingual Bidirectional and Auto-Regressive Transformers (mBART) to address sentiment prediction challenges in code-mixed tweets. Furthermore, results reveal that mBART outperformed the Electra and cm-BERT models for sentiment prediction in code-mixed text with an overall F1-score of 0.73. In addition to this, we also perform topic modeling to uncover shared characteristics within the corpus and reveal patterns and commonalities across different classes.

Keywords Code-mixed · Transformers · Natural language processing · Topic modeling · Sentiment analysis

1 Introduction

The emergence of the internet has fuelled the growth of user-generated content, as users are given the opportunity to share their opinions (i.e., sentiments) or engage in conversations on a wide range of topics via blogs, online social networks, e-commerce websites, and forums (Ali et al. 2024). As a result, a massive amount of user-generated data has been generated. People, organizations, and governments need to identify and use key information from this data (Rahman and Islam 2021). As the volume of data grows, the challenge of collecting relevant information in a timely and effective way becomes increasingly important, which emphasizes the importance of using computational linguistic approaches (Xu et al. 2022). Sentiment Analysis (SA) is an automated process that extracts users' sentiments, feelings, and emotions from raw human text. It is one of the research areas studied under the umbrella of Natural Language Processing (NLP). One of the ways to perform SA is by using the approach of text classification SA is a type of text classification in which texts are categorized based on their sentiment orientation using a supervised Machine learning (ML) approach. SA has found application in diverse fields, such as healthcare, product reviews (Cao et al. 2022), politics (Valle-Cruz et al. 2022), and more key areas. These various

Sule Yildirim Yayilgan and Sarang Shaikh have contributed equally to this work.

✉ Ehtesham Hashmi
hashmi.ehtesham@ntnu.no

Sule Yildirim Yayilgan
sule.yildirim@ntnu.no

Sarang Shaikh
sarang.shaikh@ntnu.no

[1] Department of Information Security and Communication Technology (IIK), Norwegian University of Science and Technology (NTNU), Teknologivegen 22, 2815 Gjøvik, Innlandet, Norway

Published online: 15 April 2024

applications spanning different domains demonstrate the value of SA in obtaining valuable insights into public opinion regarding specific topics of interest. SA comprises three different levels: sentence-level, aspect-level, and document-level analysis. Recent advances in Artificial Intelligence (AI) and NLP have made SA more popular, which has prompted the creation of a number of innovative methods for study in the area. These techniques provide an improved understanding of SA and its applications, including social media surveillance and customer feedback analysis (Taherdoost and Madanchian 2023). Furthermore, the primary focus of these studies has been on well-resource languages such as English (Haque et al. 2018). Prioritizing languages with more resources has resulted in a significant difference in SA research, particularly for languages with fewer resources, like English, Urdu, Roman Urdu/Hindi, Arabic, and Persian. Over the recent years, few studies have been conducted to explore SA for languages with limited linguistic resources, employing classical ML methods (Hedderich et al. 2020) and advanced NLP-based approaches. Many individuals choose to express their opinions on social media in their local languages, and Roman Urdu (RU) is one of these languages, which is widely used on social media in Asia.

RU refers to the Urdu language written in the Latin script, which is also referred to as the Roman script.[1] Many people in South Asia prefer to use RU on social media over Urdu or English because it is easier to type. This leads to a lot of short messages on cyber platforms that are code-mixed, combining English and RU in an informal way (Younas et al. 2020; Shakeel et al. 2020). One approach for dealing with low-resource languages is to use translation method. However, these methods may not be appropriate for informal code-mixed text, because they may not adequately address syntactic ambiguity, potentially causing a loss of the original context. The aim of this paper is to extract sentiments from code-mixed Roman Urdu-English social media text by leveraging multilingual transformers. These models are particularly effective in handling syntactic ambiguity while preserving the original context and long-term dependency relationships (Zhao et al. 2023). Our experiments were conducted using the MultiSenti dataset Shakeel et al. (2020), which comprises a collection of tweets in both RU and English related to the general elections in Pakistan.

We employed a variety of innovative multilingual transformers in our implementation, including Electra (Tinn et al. 2021), multilingual Bidirectional and Auto-Regressive Transformers (mBART) (Dominic et al. 2023), and cm-BERT. Various hyperparameters were utilized throughout the model training and evaluation phases, including temperature, top-k, and top-p distribution. These parameters notably contributed to the enhancement of our f1-score. This study aimed to uncover insights by addressing the following Research Questions (RQs):

1. How effective are state-of-the-art transformer-based models, such as Electra, cm-BERT, and mBART, in enhancing sentiment prediction in code-mixed texts, specifically a combination of RU and English?
2. What role does topic modeling, particularly the Latent Dirichlet Allocation (LDA) algorithm, play in uncovering shared characteristics and patterns across different sentiment classes in code-mixed social media text, and how does it contribute to the refinement of sentiment prediction models?
3. To what extent can the adaptation of generative configuration parameters (such as temperature, top-k, and top-p sampling) in transformer-based models influence the outcome of SA in code-mixed text?

1.1 Work contributions

Our contributions to the proposed work are as follows,

1. Our main contribution is to enhance the SA in code-mixed text using state-of-the-art multilingual transformer-based models, including Electra, cm-BERT, and mBART. This focus addresses the significant gap in sentiment prediction capabilities for code-mixed languages, which are notably underrepresented in computational linguistic research.
2. After a thorough analysis, we conclude that many examples in both the neutral and positive classes represent similar sentiments. To address this, we manually reviewed the data and performed the topic modeling with the LDA algorithm to combine these similar examples offering a deeper understanding of the data and also unveiling the topics in our data.
3. We conducted binary classification using multilingual transformers with customized hyperparameters, regularization, and generative configurations. This customized approach was crucial in improving our model's performance, highlighting our dedication to enhancing sentiment analysis by using advanced NLP techniques and personalized computational methods.

2 Literature review

2.1 SA in resource-aware languages

In this section, we will discuss the latest advancements in transformer-based approaches for SA in a resource-aware context. U. Naseem et al. (2020) introduced a

[1] https://en.wikipedia.org/wiki/Roman_Urdu

transformer-based methodology for binary classification of English tweets, utilizing the Transformer Deep Intelligent Contextual Embedding (DICET) framework. This framework comprises three primary components: an intelligent preprocessor, a text representation layer, and a Bi-directional Long- Short-Term Memory (BiLSTM), along with various word embeddings such as GLOVE, contextual, POS, and lexicon embeddings, integrating with attention modeling. Their study involved the utilization of three distinct datasets, including data from US airlines, airline datasets, and Emirates airlines. S. Alaparthi and Mishra (2020) performed binary classification SA for the IMDb reviews dataset in their study. They performed their research on the IMDb reviews in the English language. In their study, they implemented several algorithms, including the unsupervised Sent WordNet, as well as two supervised methods, Logistic Regression (LR), LSTM and their study revealed that the transformer-based Bidirectional Long Short Term Memory (BERT) model delivered the highest f1-score and accuracy, both reaching 0.92.

Pipalia et al. (2020) proposed a similar approach to SA using ML-based methods and the BERT transformer model. They performed their research on the same dataset as Alaparthi and Mishra (2020), which consists of IMDb reviews in English. Various transformer-based models, including (BiLSTM), BERT base, Distil-BERT, Roberta, T5, and XLNet, were implemented. The highest accuracy was attained by XLNet, with an accuracy score of 0.96. Javdan et al. (2020) introduced different ML-based and transformer-based methods for Aspect-Based Sentiment Analysis (ABSA) (Zhang et al. 2022), specifically focusing on sarcasm detection. They conducted their research using two separate datasets, one from Twitter and another from Reddit. Their study achieved the highest f1-score of 0.73 on both datasets by utilizing the BERT-base-cased model. Hashmi (2024) explored the detection of fake news in the English language using binary classification across three standard datasets. Their approach utilized both pretrained unsupervised and supervised FastText embeddings as inputs to various DL and ML-based models, enhanced with various regularization and optimization strategies. Additionally, they employed transformer-based models like BERT, XLNet, and RoBERTa, fine-tuning them with specific hyperparameters. Their innovative CNN-LSTM model, integrated with quantized FastText embeddings, surpassed existing benchmarks, achieving the highest performance metrics. In the final stage of their research, they applied Explainable AI (XAI) techniques, specifically LIME, to shed light on the decision-making process of the proposed CNN-LSTM model.

Enríquez et al. (2022) proposed a transformer-based SA approach for classifying reviews from Mexican tourists. The objective of their research was to determine the sentiment polarity expressed in tourists' opinions regarding key locations in Mexico. Their dataset comprised reviews written in both Spanish and English, with the majority of reviews written in Spanish because of the large amount of significant content available in that language. The task involved multi-class classification, with values ranging from 0 to 5, focusing on three primary topics: Hotel, Restaurant, and Attraction. Because the dataset was significantly imbalanced, data augmentation was performed to enhance the contribution of the more underrepresented classes. Initially, they employed a supervised ML algorithm, Support Vector Classifier (SVC), in combination with the FastText model and the Spanish Unannotated Corpora (SUC) vocabulary, as mentioned in Cañete (2019). Additionally, their study addressed some robust transformer-based models, including RoBERTa, which had been trained with a Spanish vocabulary known as RoBERTaESP, and GPT-2.

In the Spanish context, Jiménez-Zafra et al. (2023) presented transformer-based techniques for financial target detection and SA. The study was driven by two main goals, the first of which was to identify specific targets. During this phase, participants were tasked with identifying the main economic topic of discussion in newspaper headlines. Following that, teams were asked to categorize the sentiment polarity (positive, neutral, or negative) associated with that target within the processed text, assuming they had correctly identified the core economic issue in these financial news headlines. In their study, several advanced English and Spanish language transformer models were used, including MarIA, FinancialBERT, BETO, RoBERTuito, and mDeBERTa. Among these models, RoBERTuito achieved an 11-score of 0.7576, while MarIA exhibited f1-scores of 0.6050 for evaluating sentiments related to companies and 0.6968 for consumer SA.

2.2 SA in resource-limited languages

In their study, Ilyas et al. (2023) developed a two-level classification approach to detect emotion detection for Roman Urdu and English (RU-EN) text at the sentence level. They collected a dataset of 400,000 sentences from social media, manually selecting 20,000 code-mixed RU-EN sentences for annotation. In the annotation process, sentences were categorized into two levels: at the first level, sentences were classified as either "Neutral"n or "Emotion-sentences," and at the second level, emotions were classified into Anger, Fear, Happy, Sad, and Surprise. The researchers conducted several experiments using both traditional ML-based and Deep Learning (DL) techniques. Among these, Convolutional Neural Networks (CNNs) combined with GLOVE word embeddings proved to be the most effective.

Altaf e al. (2023) introduced a sentence-level SA method focused on classifying idioms and proverbs. They employed classical ML techniques and created an annotated dataset

consisting of 1800 Urdu script sentences, with half originating from the news domain. Linguistic characteristics were retrieved from this dataset using Part-of-Speech (POS) tagging, yielding an accuracy score of 0.90 when the J48 classifier was used. Muhammad and Burney (2023) created an Urdu SA system with two classes and a balanced dataset of 3737 negative and 2815 positive labels. They achieved an accuracy of 0.89 by combining Recurrent Neural Network with CNN. Hossain et al. (2023a) presented a novel text classification framework for Bengali, a low-resource language. It introduces a method that combines Average meta-embedding (AVG-M) with a CNN to enhance classification accuracy. The approach effectively addresses the challenges of inadequate standard corpora, hyper-parameter tuning, and language-specific embeddings. The framework achieves remarkable classification accuracies on four Bengali corpora: 0.96, for BARD, 0.93 for Prothom-Alo, 0.90 for a newly developed 11-category corpus, and 0.87 for IndicNLP, showcasing the method's efficacy in handling text classification in Bengali. In their study, Hashmi (2024) addressed the challenge of detecting multi-class hate speech in Norwegian texts. They employed a combination of supervised learning using the FastText framework and DL-based models, achieving significant success. Their innovative FAST-RNN model, which merges FastText with a BiLSTM-GRU architecture, surpassed existing benchmarks, delivering superior performance. Additionally, the study incorporated Explainable AI (XAI) through the use of Local Interpretable Model-Agnostic Explanations (LIME), enabling the researchers to understand the rationale behind specific predictions. For language processing, the team utilized Norwegian language models including Nor-BERT, scandiBERT, nb-BERT, and nor-T5, as well as multilingual transformer-based models like FLAN-T5, mBERT, ELECTRA, and mBART, further enhancing their analysis. Khan et al. (2022) presented a deep neural network architecture for sentiment categorization in code-mixed texts. CNN layers are utilized for feature selection, and Long Short-Term Memory (LSTM) layers are applied to capture long-term dependencies in textual input. They also used several word embedding techniques, such as Word2Vec Continuous Bag of Words (CBoW), GLOVE, and FastText. A similar approach was used by Nagra et al. (2022) where they conducted SA at the sentence level for RU using Faster Recurrent CNN (FR-CNN) on the RUSA-19 dataset. Their study encompassed two classification tasks: a binary classification involving positive and negative instances, and a tertiary classification that included neutral, positive, and negative instances.

Shakeel et al. (2020) conducted sentence-level sentiment prediction for online tweets during the 2019 general election in Pakistan. Their dataset included code-mixed text in RU, English, and a combination of both languages. They applied various transfer learning embeddings and attentional LSTM-based methods for a multi-classification task, which involved categorizing tweets as neutral, positive, or negative. The dataset was highly imbalanced and the same dataset was used by Younas et al. (2020) in their proposed work. In their suggested work they utilized fine-tuned transformer-based models such as mBERT and XLM-R. When compared to the baseline (Shakeel et al. 2020), these models performed better, making their approach more robust and accurate. M. Hossain et al. (2023b) explored a novel approach to identifying COVID-19-related texts in Bengali, addressing the challenge of misinformation and the scarcity of NLP-based tools for low-resource languages. The research introduces CovTiNet, a DL-based network that incorporates attention-based positional embedding and feature fusion to enhance the identification of COVID-19-related texts. The proposed model achieves a remarkable accuracy of 0.97 on the developed Bengali COVID-19 Text Corpus (BCovC), outperforming other baseline models such as BERT-M, IndicBERT, ELECTRA-Bengali, DistilBERT-M, BiLSTM, DCNN, CNN, LSTM, VDCNN, and ACNN. This advancement not only contributes to the computational linguistics field by providing a robust model for processing low-resource languages but also aids in mitigating the spread of misinformation related to COVID-19.

Javed and Saeed (2023) addressed the binary classification problem for the code-mixed text using ensemble learning techniques, LSTM, and the BERT model. After fine-tuning the model using various hyperparameters, the BERT model achieved a significant accuracy score of 0.90. This result demonstrates BERT's effectiveness in optimizing the model's performance. Ahmad and Singla (2022) proposed a DL-based approach for language recognition and sentiment prediction in code-mixed English-Urdu text. An Artificial Neural Network (ANN) was used in conjunction with character-based embeddings in the context of language identification, giving considerable results. This highlights the effectiveness of DL approaches for dealing with code-mixed text in language-related tasks. They worked on the tertiary classification problem, which included "negative," "positive," and "neutral" sentiment categories. They gathered data from popular Facebook (FB) sites, sports figures, and YouTube, giving a wide variety of sources for research.

Qureshi et al. (2023) introduced a sentiment prediction method for song reviews written in Urdu using Roman script. The Indo-Pak music sector is expanding, with a robust industry, and technology has been playing a key role in its development. Song reviews provide an excellent way to assess the content quality of these songs. They created a dataset using music comments to do SA. They used both unsupervised K Nearest Neighbours (KNN) and supervised ML-based approaches in their work. The Naïve Bayes (NB) technique showed the best accuracy score, reaching 0.82. Fuadi et al. (2023) proposed multilingual Text-2-Text

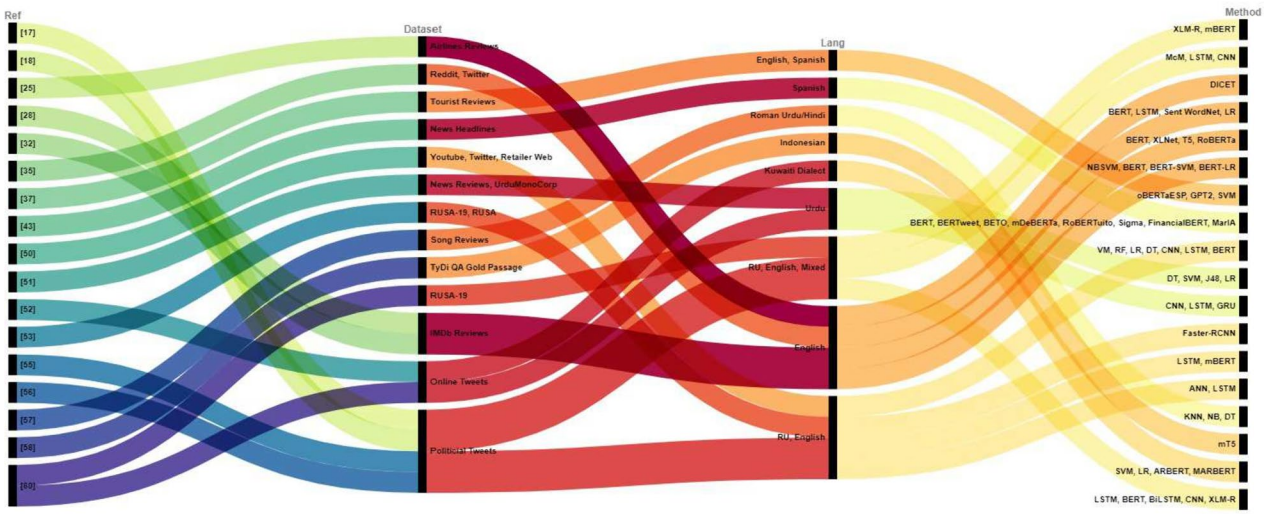

Fig. 1 Related SOTA studies

(mT5) for sentiment classification for the Indonesian language using TyDiQA-GoldP[2] dataset with only Indonesian instances. MT5 is a variant of T5 but this model had not been initially fine-tuned for any specific downstream tasks. Consequently, in their research, they performed fine-tuning on the SmSA dataset, as referenced in Wilie et al. (2020), to adapt the model to the sentiment classification task which resulted in an accuracy of 0.70.

Husain et al. (2022) addressed the tertiary classification problem in their research study for the Kuwaiti dialect. Over the course of a year, they collected the dataset, which included tweets about a wide range of issues, including murder with varied perspectives and employment-related material. Dataset has three labels neutral, positive, and negative. They utilized several ML-based techniques such as LR, SVM, bagging, and transformer-based models such as AraBERT, ARBERT, and MARBERT. The results of their study achieved an accuracy of 0.89 when the ARBERT model was applied to the testing dataset. In their study, Hossain et al. (2024) introduce AraCovTexFinder, an advanced system for identifying Arabic COVID-19-related texts, leveraging fine-tuned transformer models. Achieving an impressive accuracy of 0.99, AraCovTexFinder significantly outperforms conventional transformer-based and DL-based models. This development marks a substantial advancement in processing Arabic COVID-19 information, highlighting its potential to support informed decision-making in managing pandemic-related data.

The following Table 1 represents the comparative analysis of the current state-of-the-art methods along with a summarized version of the studies shown in Fig. 1.

[2] https://paperswithcode.com/dataset/tydiqa-goldp

After reviewing the existing literature, we conclude that many studies used transformers in both resource-aware and resource-limited scenarios. However, there is still a need to utilize the use of transformers primarily for analyzing code-mixed ENG-RU tweets. In this study, we will do this analysis using transformers in a different way that will provide us with a deep understanding of transformers for the code-mixed text and also produce robust results.

2.3 Problem statement

The increasing prevalence of code-mixed text on social media, where users blend languages within a single utterance, presents a significant challenge for SA. Traditional NLP-based models struggle with the syntactic ambiguity and the nuanced semantics of these texts, particularly in low-resource languages like Roman Urdu mixed with English. This study addresses the gap in effective SA tools for code-mixed languages, focusing on the development and evaluation of advanced transformer-based models. These models aim to enhance sentiment prediction accuracy by leveraging the complexity and diversity inherent in code-mixed texts, thus contributing to the broader understanding and processing capabilities for multilingual and code-mixed digital communications.

3 Methodology

The proposed research methodology using multilingual transformer-based models in this study involves a systematic approach to achieving promising results as shown in Fig. 2. Each of the steps from our research methodology is further elaborated in detail below.

Table 1 Comparative Analysis of Selected Studies

Refs.	Dataset	Lang	Feature Set	Method	Results
Younas et al. (2020)	Political Tweets	RU, English, Mixed	Contextual Embeddings	XLM-R, mBERT	F1-Score: 0.71
Shakeel et al. (2020)	Political Tweets	RU, English, Mixed	ELMO, ConvNet	McM, LSTM, CNN	Accuracy: 0.69
Naseem et al. (2020)	Airlines	English	Glove, Word2Vec, POS	DICET	Accuracy: 0.96
Alaparthi and Mishra (2020)	IMDb	English	TextBlob, VADER, AFINN	BERT, LSTM, Sent WordNet, LR	F1-score: 0.92
Pipalia et al. (2020)	IMDb	English	Contextual Embeddings	BERT, XLNet, T5, RoBERTa	Accuracy: 0.96
Javdan et al. (2020)	Reddit, Twitter	English	Glove, FastText	NBSVM, BERT, BERT-SVM, BERT-LR	F1-score: 0.73
Enríquez et al. (2022)	Tourist Reviews	English, Spanish	FastText	RoBERTaESP, GPT2, SVM	Accuracy: 0.99
Jiménez-Zafra et al. (2023)	News Headlines	Spanish	Contextual Embedding	BERT, BERTweet, BETO, mDeBERTa, RoBERTuito, Sigma, FinancialBERT, MarIA	F1-score: 0.75
Ilyas et al. (2023)	YouTube, Twitter, Retailer Web	RU, English	GLOVE, FastText, Word2Vec, Counter Vectorizer	SVM, RF, LR, DT, CNN, LSTM, BERT	F1-score: 0.88
Altaf e al. (2023)	News Reviews, UrMono-Corp	Urdu	POS Tagging	DT, SVM, J48, LR	Accuracy: 0.90
Muhammad and Burney (2023)	Online Tweets	Urdu	Manual Tagging	CNN, LSTM, GRU	Accuracy: 0.89
Khan et al. (2022)	RUSA-19, RUSA	RU, English	N-gram	Faster-RCNN	Accuracy: 0.92
Javed and Saeed (2023)	Political Tweets	RU, English	Contextual Embeddings	LSTM, mBERT	Accuracy: 0.90
Ahmad and Singla (2022)	Political Tweets	RU, English	Character Based Embeddings	ANN, LSTM	Accuracy: 0.72
Qureshi et al. (2023)	Song Reviews	Roman Urdu/Hindi	Rapid-Miner	KNN, NB, DT	Accuracy: 0.82
Fuadi et al. (2023)	TyDi QA gold passage	Indonesian	Contextual Embeddings	mT5	Accuracy: 0.77
Husain et al. (2022)	Online Tweets	Kuwaiti Dialect	Contextual Embeddings	SVM, LR, ARBERT, MARBERT	Accuracy: 0.89
Rizwan et al. (2020)	RUSA-19	RU, English, Mixed	ELMO, FastText, LASER	LSTM, BERT, BiLSTM, CNN, XLM-R	F1-Score: 0.89
Hashmi et al. (2024)	Online Tweets, Social Media Comments	FastText, Cotextual Embeddings	BiLSTM-GRU, CNN-LSTM, Nor-BERT, Nor-T5, FLAN-T5, ELECTRA, nb-BERT, scandiBERT, mBERT, mBART	F1-Score: 0.98	

3.1 Dataset

In our study, we addressed the tertiary classification problem using the same dataset as the one used in the baseline study Shakeel et al. (2020). The dataset consists of three distinct classes: 1 for neutral, 0 for negative, and 2 for positive. It includes text written in both RU and English, as well as a combination of these two languages referred to as code-mixed text. The dataset consists of tweets originating from Pakistan's general election in 2019, where people expressed their opinions about the political parties and leaders they favored. Figure 3 shows the language distribution in the dataset.

Below Table 2 provides a summary of the dataset, showing the counts of languages and labels,

Based on the information presented in the table above, it is s clear that the dataset exhibits an imbalance, with a lower count of neutral tweets and instances in the English language compared to the other languages. This dataset's imbalance is a key aspect that our research will emphasize, particularly when evaluating the F1 score.

3.2 Data preprocessing

Data preprocessing plays an important part in any method before feeding the data to our model. It usually includes steps

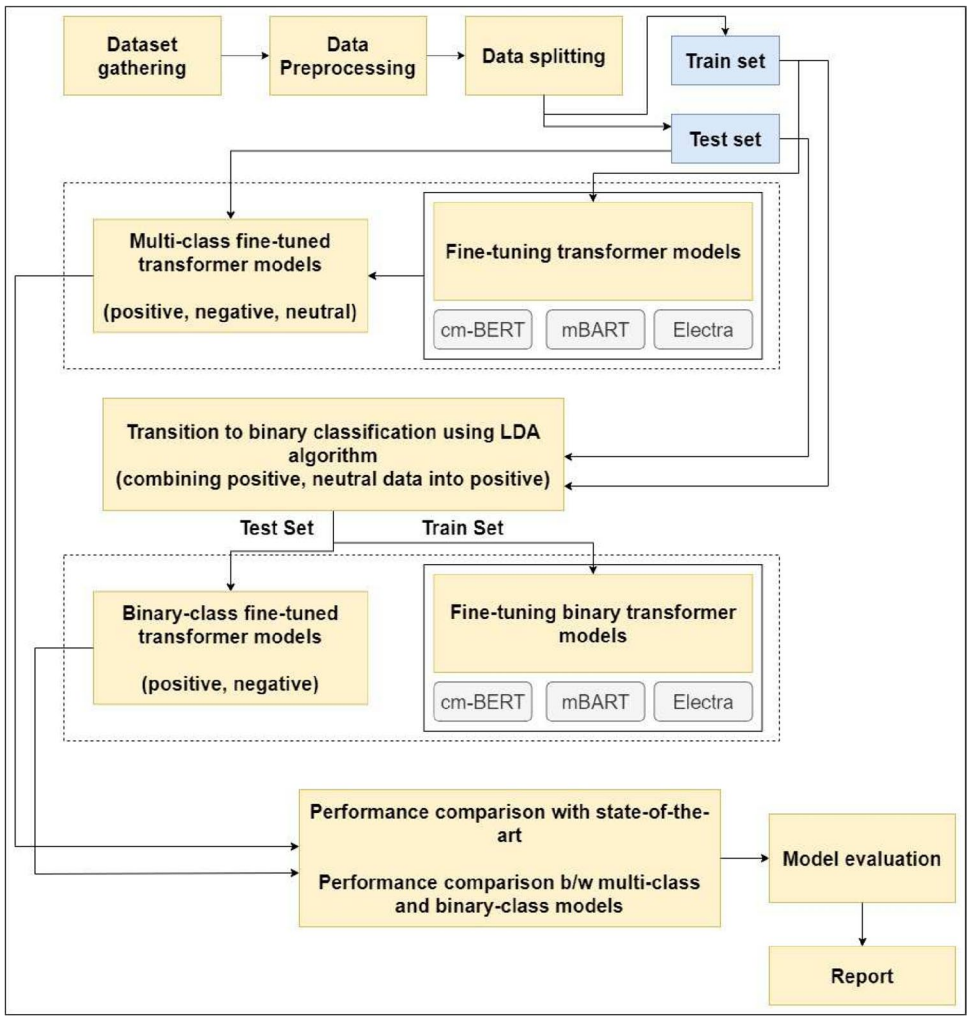

Fig. 2 Proposed methodology

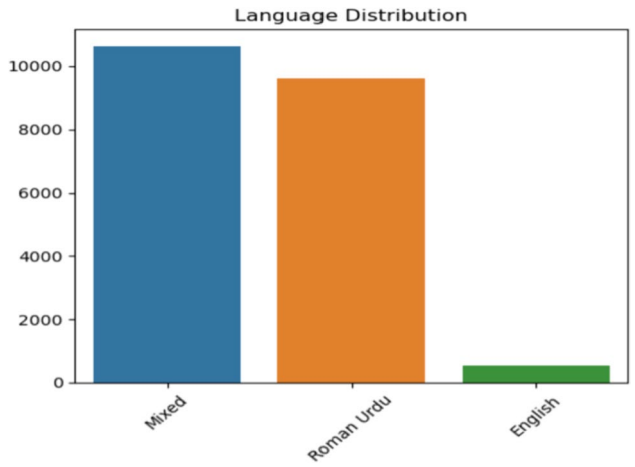

Fig. 3 fig: Language Distribution

like lemmatization or stemming, removing stop words, and tokenizing words and sentences to clean the data. Proper preparation of data enhances the quality and pertinence of

Table 2 Count of Labels and Dialects

Class	Class Counts	Lang	Counts
Negative	10008	Roman Urdu	9609
Neutral	3452	English	521
Positive	7275	Mixed	10605

the information utilized for training and analysis, which in turn substantially boosts the efficacy of ML-based models. Initially, we transformed all uppercase letters to lowercase to maintain consistency. Next, we eliminated stop words from both English and Roman Urdu text to reduce noise. Following that, we executed word tokenization to break down the text into individual words. Lastly, we purged irrelevant ASCII characters and filtered out unnecessary patterns using regular expressions. However, in the case of multilingual transformers, the approach to data preprocessing will be distinct because transformers need to have an understanding of the whole sentence even the whole document. In our scenario, we conducted a limited set of preprocessing steps, deliberately

Table 3 Examples of Code-Mixed Tweets with English Translation

Class	Language	Tweet	English Translation
Negative	RU	mujhe wahan add kro sulah krwata hun sbki	add me there i will reconcile everyone
Positive	RU	sukkur na 207 say bhi pti jeti hai liken phir dubra jeet pppp ko dilwadi gei	sukkur pti won from na 207 as well but then they let ppp win again
Positive	Mixed	is tweet k bad kasirah or kadirah ne apne next serial me ali ko leading role dene ka faisla krlya	after this tweet kasirah and kadirah decided to cast ali in the leading role in their next serial
Neutral	Mixed	imrankhanpti congratulations pakistan pti amp imran khan anwar ul haque chakwal	imrankhanpti congratulates pakistan pti and imran Khan anwar ul haque chakwal
Neutral	English	jawabdeyh independents are joining pti	answerable independents are joining pti
Positive	Mixed	bander agar pmln ny 10 years ma police ko nonpolitics rakha hota tu aisa nahi hota jani	monkey if pmln had kept the police out of politics for 10 years things would not t have been like this
Positive	English	looks like i ll be moving back to pakistan	looks like i will be moving back to pakistan

excluding the removal of stop words, as it is not recommended under any circumstances. For these transformer-based models, preprocessing primarily involves converting uppercase letters to lowercase, eliminating irrelevant characters such as ASCII symbols, and tokenizing words and sentences. Another reason for limiting preprocessing is to address the issue of syntactic ambiguity, which has been a significant drawback in previous DL-based techniques and models. Syntactic ambiguity occurs when words within a sentence might have several interpretations depending on the context, making it a difficult problem to interpret. In the following Table 3 there are some examples of English, RU, and mixed instances,

The table above clearly illustrates that all the words and characters have been converted to lowercase. In the last example, you will notice the absence of an apostrophe sign between "i'll" which was a result of our preprocessing.

3.3 Transformed-based models

The Transformer is an NLP system designed to execute sequence-to-sequence processes following the self-attention mechanism with long-range dependencies comprising two main components encoder and decoder. Transformers were first introduced in 2017 by Vaswani et al. (2023); Hashmi (2024). The self-attention mechanism can be expressed mathematically as follows,

$$Attention(Q, K, V) = \text{softmax}(\frac{QK_i^\top}{\sqrt{d_k}})V_i \quad (1)$$

where:

Q: is the loss to minimize

K: is the key matrix

V: is the value matrix

d_k: is the dimension of the key vectors

N: is the length of the input sequence

i: is the index of the query vector

Our research is based on utilizing transformers and focuses on optimizing their hyperparameters and various related distributions such as top-k, top-p, and temperature, that play a crucial role in shaping the behavior of these models. Previous language models like RNNs, which were constrained by their computational and memory requirements for generative tasks, these transformers offer a significant advancement. The limitation of RNNs is that they heavily rely on predicting the next word based solely on the previous word, and even with increased scaling, they often struggle to provide accurate predictions in many cases. For a successful word prediction, a more comprehensive understanding of the entire sentence or even the entire document, and this is precisely why transformers have become the preferred choice.

In our study, we have multilingual text data and for the sake of this, we adopted multilingual transformers such as cm-BERT, mBART, and Electra.

3.3.1 cm-BERT

BERT is a transformers model that was self-trained on a large multilingual dataset. This implies it was trained utterly on raw text, with no human annotations, using publicly

available data and an automated approach to extract inputs and labels from the text. mBERT, on the other hand, is a variant of BERT that has been pre-trained exclusively on the most extensive Wikipedia content from the top 104 languages, using a Masked Language Modelling (MLM) target (Devlin et al. 2019). We fine-tuned this multilingual model on our code-mixed RU-ENG tweets, which led us to name it "code-mixed BERT," abbreviated as cm-BERT. This fine-tuning was necessary to achieve promising results, and we utilized the SequenceClassification class type along with the AutoTokenizer for this purpose.

3.3.2 mBART

Multilingual Bidirectional and Auto-Regressive Transformers (mBART) are a type of pre-trained language model that was trained on diverse monolingual datasets in multiple languages. This approach aimed to enhance the model's understanding of language across various linguistic contexts, which is crucial for its performance in multilingual tasks. It was one of the first methods for pretraining a comprehensive sequence-to-sequence model by denoising entire texts in several languages, whereas previous efforts had concentrated primarily on the encoder, decoder, or the reconstruction of text segments. In comparison, mBERT primarily focuses on cross-lingual understanding, making it suitable for tasks involving language transfer learning. In our study, we utilized mBART with the SequenceClassification class type which is mainly used for the classification tasks, and we employed the MBartTokenizer. This model was then trained to predict the original text, X, from these noisy inputs (Liu et al. 2020).

$$L_\theta = \sum_{D_i \in D} \sum_{X \in D_i} \log P(X|g(X);\theta) \tag{2}$$

where:

L_θ: is the loss to minimize

$\sum_{D_i \in D}$: sums over datasets

$\sum_{X \in D_i}$: sums over data points

$\log P(X|g(X);\theta)$: assesses prediction accuracy

3.3.3 Electra

In MLM pretraining approaches like BERT, the input data is altered by substituting certain tokens with the placeholder [MASK], and the model is subsequently trained to recover the original tokens from this modified input. Electra, as an alternative to traditional MLM pretraining methods, employs a more sample-efficient approach known as replaced token detection. In comparison to masking the input, Electra modifies it by substituting certain tokens with credible alternatives generated from a smaller generator network. Instead of training a model to predict the original tokens that were altered, Electra trains a discriminative model to determine whether each token in the modified input has been replaced by a generator-generated sample or remains unaltered. In this study, we used Electra with the SequenceClassification class type and employed the ElectraTokenizer. The generator produces a probability associated with generating a specific token x_t by employing a softmax layer. The following are the mathematical expressions for the generator and discriminator modules of the Electra algorithm (Clark et al. 2020).

$$P_G(x_t|x) = \frac{e(x_t)^T h_G(x)_t}{\sum_{x_0} \exp(e(x_0)^T h_G(x)_t)} \tag{3}$$

where:

$P_G(x_t|x)$: probability of predicting token x_t given context x

$e(x_t)$): embedding of the target token x_t

$h_G(x)_t$: hidden representation of context x at position t

$\exp(e(x_0)^T h_G(x)_t)$: Normalization term for probabilities

Following is the mathematical expression for the **discriminator** part of Electra,

$$\check{}_{\mathrm{D}} = -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[y \cdot \log(D(x)) + (1-y) \cdot \log(1-D(x))\right] \tag{4}$$

where:

$\check{}_{\mathrm{D}}$: discriminator's loss

$\mathbb{E}_{(x,y)\sim\mathcal{D}}$: expectation over data samples (x, y)

y: binary label indicating real or generated data

$\log(D(x))$: logarithm for real data

$\log(1 - D(x))$: logarithm the discriminator

3.4 Generative configuration

In our fine-tuning process for the proposed multilingual transformers, we made significant adjustments to the hyperparameters, leading to noticeable changes in our results. This involved experimenting with various batch sizes, learning rates, and epochs. Most importantly, we also employed generative configuration parameters, which are additional parameters that the model utilizes during training. These parameters are invoked during the inference phase, providing us with control over factors such as the maximum token count in the generated output and the level of creativity in the text. While many transformers typically rely on a greedy decoding approach, where the word with the highest predicted probability is chosen, we opted for

more natural text generation techniques in our study. These techniques include random sampling methods like **top-k** and **top-p**, which impose constraints on randomness and increase the likelihood of producing creative and diverse outputs (Hashmi et al. 2024)

Top-k sampling selects the top-k most probable words from the model's probability distribution for the next token. The formula is as follows:

$$P(w) = \begin{cases} \frac{e^{P(w)}}{\sum_{w'} e^{(P(w'))}} & \text{if } w \text{ is in the top-}k \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where:

w: is the word being sampled,

$P(w)$: is the probability of word,

V: is the vocabulary of possible words.

Top-p sampling selects the minimum number of words needed to have a cumulative probability exceeding a predefined threshold p. Following is the mathematical expression to calculate the top-p sampling (Hashmi et al. 2024),

$$P(w) = \frac{1}{\sum_{w' \in V : P(w') \geq p} P(w')} \quad (6)$$

where:

$\sum_{w' \in V : P(w') \geq p} P(w')$: sum of probabilities

Additionally, we incorporated another available set of configuration parameters, namely "temperature", into our approach. This parameter has a direct impact on the probability distribution that the model computes for predicting the next token. The temperature value acts as a scaling factor applied within the softmax layer of the transformers. A higher temperature setting increases the randomness in the generated output, while a lower temperature value reduces the range of possible words in the generated text (Hu et al. 2023). Following is the mathematical expression for random sampling with temperature,

$$P(w) = \frac{exp^{(P(w)/\tau)}}{\sum_{w'} exp^{(P(w')/\tau)}} \quad (7)$$

where:

τ: temperature parameter controlling distribution diversity

$\sum_{w'} exp^{(P(w')/\tau)}$: normalization factor

Table 4 provides an overview of the hyperparameters employed during the fine-tuning of our models, including details on class type and the tokenizer used.

Table 4 Hyperparameter and Configuration Details

Model	Temp	Top-k	Top-p	Tokenizer
mBART	1	5	0.3	mBart
cm-BERT	1	None	None	Auto
Electra	1	7	0.3	Electra

3.5 Transition to binary classification with LDA algorithm

In the initial phase of our experiments, we conducted multi-class classification with three distinct classes: negative, neutral, and positive. However, as our exploration progressed, we applied the LDA algorithm, which revealed the underlying topic distributions and the associations among the classes. This analysis unveiled the similarity between neutral and positive classes, motivating a transition in our approach. To gain more focused insights and simplify the classification task, we subsequently shifted to a binary classification framework, with neutral and positive classes. This transition allowed us to address the shared attributes and topic-related nuances between the two closely related classes, refining our experimental design for enhanced results. Table 5 represents tweets that share a similar sentiment, yet they have been categorized into different classes.

In Table 6, we performed the LDA algorithm to extract the top five most similar features within both classes neutral and positive, which provided us the similar words in both classes such as "pti", "ppp", and "khan". This analysis led to our decision to merge the initially distinct three classes into two, as it became evident that they share nearly identical characteristics. Furthermore, we also observed that the dataset consisted of political discussions during the 2019 elections in Pakistan. In this regard, we categorized the dataset into three distinct topics: National Politics Leadership, Regional Politics, and Elections (Table 7).

Following our experiments, we merged neutral and positive classes into one due to an abundance of positive instances. As a result, our focus now centers on the binary classification problem of distinguishing between negative and positive sentiments. To evaluate the LDA results, we implemented **coherence** and **perplexity** as key metrics. Perplexity measures model prediction accuracy, while coherence measures topic interpretability (Mifrah and Benlahmar 2020). These metrics were instrumental in assessing the quality of our LDA model and guiding our analysis (Hasan et al. 2021). Equations 8 and 9 compute the coherence Yang et al. (2023) and perplexity (Gan and Qi 2021).

$$\text{coherence}(V) = \sum_{(v_i, v_j) \in V} \text{score}(v_i, v_j) \quad (8)$$

Table 5 Examples of Binary Classes

Class	Tweet	English Translation
Neutral	Mere pyarey doston jeet mubarrik ho	happy victory my dear friends
Positive	mai khan ko jeet ki mubarak bad paish krta hun	i congratulate Khan on his victory
Neutral	pti ki jeet par jamaima ko Mubarakad	congratulations to jemima on pti's win
Positive	cmshehbaz tum jeeto ya haaro humain tum sy pyar hai	cmshehbaz either you win or lose we love you

Table 6 Similar Attributes in Neutral and Positive Class

Class Label	Attributes
Neutral	pakistan, khan, naya, pti, ppp
Positive	ppp, vote, pti, khan, election

Table 7 Topic Modeling with LDA

Topic	Attribute
National Politics Leadership	pakistan, imran, naya pakistan, nawaz
Regional Politics	punjab, karachi, pti, mqm, ppp
Political Elections	seats, votes, election, pti, pmln

$$\text{perplexity}(D_{\text{test}}) = \exp\left(-\sum_{d=1}^{M} \log p(w_d) \Big/ \sum_{d=1}^{M} N_d\right) \tag{9}$$

4 Results and discussion

In our evaluation, we employed the following standard metrics for assessment: accuracy, precision, recall, and f1-score. These metrics are computed using the following equations, which provide quantitative measures of the model's performance. Analyzing these metrics allows us to gain valuable insights into its classification accuracy and its ability to correctly identify and differentiate between class sentiments. These metrics serve as essential tools for assessing the model's effectiveness and refining its performance in SA-related tasks, contributing to the overall improvement of the model.

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{10}$$

$$\text{Precision} = \frac{T_P}{T_P + F_P} \tag{11}$$

$$\text{Recall} = \frac{T_P}{T_P + F_N} \tag{12}$$

Table 8 Multiclassificaiton SA Results

Classifier	P	R	A	F
cm-BERT	0.70	0.69	0.70	0.71
Electra	0.71	0.71	0.71	0.70
mBART	0.71	0.71	0.72	0.73

$$\text{F1-Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{13}$$

The transformer-based models intended for SA in RU are trained using the baseline dataset, as outlined in Table 8. Following training, the proposed model's performance is assessed using various metrics such as accuracy, precision, recall, and f1-score. The dataset is then split into distinct training and testing sets, with 80% of the data dedicated to training and 20% for testing in each corpus.

The multiclassification SA results demonstrate the performance of three different classifiers: cm-BERT, Electra, and mBART. Among these, mBART achieved the highest scores across precision 0.71, recall 0.71, accuracy 0.72, and F1-score 0.73, to classify sentiments into multiple categories accurately. Electra and cm-BERT showed comparable performance, with Electra slightly edging out in precision and accuracy but tying in recall and having a marginally lower F1-score. Following this analysis, we conducted LDA to uncover hidden thematic structures and similarities among the different sentiment classes, providing deeper insights into the data. This exploration further informed our decision to transition to binary classification to refine our understanding and treatment of sentiment analysis in complex textual datasets.

We performed a thorough analysis of the model predictions to obtain deeper aspects of the misclassified tweets by the models and to better understand the cause of these errors. The goal of this objective was to provide helpful insights into any potential difficulties with the dataset and the annotation process. To perform the error analysis, our focus was specifically on the mBART model, which demonstrated a higher F1-score and accuracy compared to the other transformer-based models(cm-BERT, Electra) in our study.

Table 9 Multiclassification Error Analysis

True	Predicted	Tweet	English Translation
Neutral	Negative	pakistan bhaar sa uturn ka sign ki jagha ik ki pic lga do	Put a picture of one in place of the U-turn sign from Pakistan
Neutral	Negative	bhaiya aaap bohat rotay hoo	brother you cry a lot
Negative	Neutral	mujhe wahan add kro sulah krwata hun sbki	Add me there i i will reconcile everyone
Negative	Neutral	pmln 4 pti 1 par lead thi result abhi 2 k aye hain	pmln was leading with 4 pti 1 results for 2 are still pending
Positive	Neutral	jail se chutkara panay k liye hi to pti ko vote dia tha	voted for pti to get rid of fools
Positive	Neutral	tabdali aie ray pti walo enjoy	change has come pti supporters enjoy
Neutral	Positive	hahaha kalli kagrri ik not shifting to pm house	hahaha kalli khagri ik is not shifting to the pm house
Neutral	Positive	i hope ik ne jo promises kiye hain wo unko pora karay	i hope ik fulfills the promises he made to them

4.1 Error analysis

4.1.1 Multiclassification error analysis

In Table 9, several instances were misclassified due to specific words that led the model to make incorrect predictions. The error analysis in the multiclassification of sentiments reveals a pattern where specific words or phrases significantly influence the model's prediction accuracy. Notably, the misclassification of sentiments from neutral to negative or positive and vice versa suggests a complex interplay between context and keyword recognition. For example, sentiments expressed in tweets 1 and 2 were inaccurately classified as negative due to possibly misinterpreting emotional expressions or culturally specific phrases without negative connotations. In **examples 4, 5** and **6**, when the model encountered the word "**pti**," it erroneously classified it as neutral when it was actually negative and positive respectively according to the true label. Notably, in these instances, there were other positive terms present, such as "**enjoy**" and "**tabdali**," which translates to "**change**" in English, and these terms provide us the positive sentiment in general. Besides above mentioned examples, there are many other examples in a dataset where this overlap between neutral and positive classes has been observed. The figures provided below illustrate the classification matrix and error analysis for multiclass SA in code-mixed tweets. Similarly, the model's difficulty in discerning the underlying sentiment in **examples 7 and 8**, where optimistic expressions were misclassified as positive from neutral, underscores the challenge of accurately capturing sentiment nuances in diverse linguistic contexts. These instances highlight the importance of enhancing the model's ability to understand context, idiomatic expressions, and cultural nuances to improve accuracy in sentiment classification (Fig. 4, Table 10).

In the context of SA for RU, our LDA model achieved a coherence score of 0.3056, showing that it has a good understanding of the more detailed and subtle aspects of the data. The model's perplexity value of −8.9494 suggests strong predictive performance for unseen data.

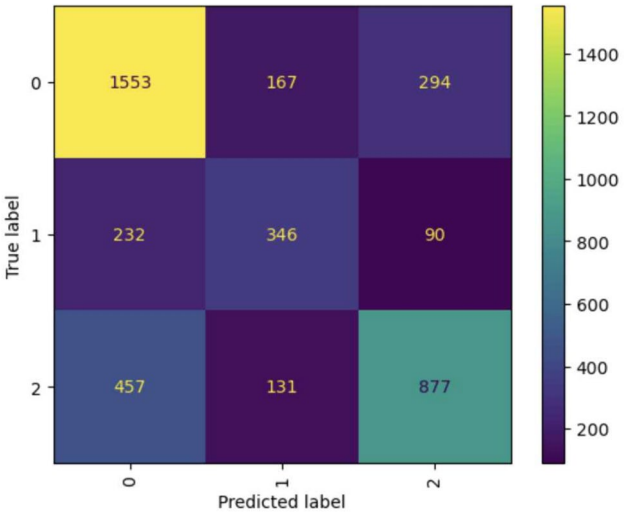

Fig. 4 Confusion matrix for multiclass SA

Table 10 Topic modeling evaluation results

Evaluation Measures	Scores
Coherence	0.305635035
Perplexity	−8.94938796

Table 11 Binary classification SA Results generative configurations

Classifier	P	R	A	F
cm-BERT	0.73	0.72	0.74	0.74
Electra	0.72	0.72	0.71	0.71
mBART	0.74	0.73	0.74	0.75

Lower perplexity values are generally better, as they indicate that the model is better at predicting the given context. These findings are particularly significant given the absence of established benchmarks for this language, highlighting the model's potential in this unique context.

Table 12 Binary Classification SA Results with Generative Configurations

Classifier	P	R	A	F
cm-BERT	0.74	0.73	0.75	0.75
Electra	0.72	0.73	0.73	0.73
mBART	0.75	0.75	0.77	0.77

4.1.2 Binary classification error analysis

For the error analysis in binary classification, we focus on the mBART model in Table 13 This allows us to understand how the model performs in this specific scenario.

In the table above, it is evident that the models made incorrect predictions in various instances. For instance, in **example 2**, the model labeled a tweet as negative due to the presence of the word "**kutti** which has the meaning of "**bitch**" in English," which generally conveys a negative sentiment but still the instance has been annotated as Positive. Similarly, in another **example 4**, the model predicted a positive sentiment because of the phrases "**nayapakistan**" and "**mubark**" with English translations of "**new Pakistan**" and "**congratulations**," which convey a positive meaning, contrary to the manual annotation, which was likely to be influenced by an inaccurate label due to the tweet's apparent positive nature.

Table 13 Binary Classification Error Analysis

True	Predicted	Tweet	English Translation
Negative	Positive	lol yaa to hona hi tha	this had to happen lol
Positive	Negative	ye ik kutti cheez hai	it is a bitch thing
Negative	Positive	waseembadami mqm ko sath milna hoga ussay karachi ki halat b achi hojay gi	waseem badami will have to join mqm it will improve karachi's situation
Negative	Positive	nayapakistan imrankhan mubark ho	congratulations on the new pakistan imran khan

Based on this information, Tables 11 and 12 display the results of binary classification using the same transformer-based models.

The comparative analysis of the SA results from two different setups without and with generative configurations reveals interesting insights into the performance of various classifiers on code-mixed text. In the first Table 11, without generative configurations, the classifiers cm-BERT, Electra, and mBART exhibited a balanced performance with cm-BERT and mBART showing slightly better accuracy and F1-score values of 0.74 and 0.74, 0.74 and 0.75 respectively, compared to Electra's 0.71 for both metrics. Notably, precision and recall were closely matched across all three classifiers, indicating a consistent level of performance in identifying positive and negative sentiments without generative configurations. When generative configurations were employed in Table 12, an overall improvement in performance was observed for all classifiers. The cm-BERT and mBART models, in particular, demonstrated a noteworthy enhancement with cm-BERT's accuracy and F1-score increasing to 0.75 and 0.75, and mBART achieving the highest scores across all metrics with an accuracy and F1-score of 0.77. Electra also showed improvement, particularly in recall and accuracy, which rose to 0.73 and 0.73, reflecting a better balance in identifying the full range of sentiments within the code-mixed text. This comparison underscores the potential benefits of integrating generative configurations into the sentiment prediction of code-mixed text, with mBART emerging as the most effective classifier in leveraging these configurations to enhance its predictive capabilities.

5 Comparison of the results with the State-of-the-Art

In this section, we evaluate our multi-classification results in comparison to the baseline methods, as cited in Shakeel et al. (2020) and Younas et al. (2020). After incorporating our proposed language models with a generative configuration that includes fine-tuning hyperparameters like top-k, top-p, temperature, and various other specific adjustments, our approach not only outperforms both baseline approaches but also excels in terms of accuracy, precision, recall, and notably, in f1-score. The dataset was highly imbalanced and our model's effectiveness in dealing with this challenge resulted in a notable f1-score. This significant performance improvement demonstrates the effectiveness of our work in delivering better results in the field of code-mixed tweets. Table 14 shows the comparison between our results and the baseline.

Table 14 Comparison of the Results with SOTA

Ref	P	R	A	F
(Shakeel et al. 2020)	0.71	0.62	0.69	0.64
(Younas et al. 2020)	-	-	0.71	0.71
Proposed Work	0.71	**0.71**	**0.72**	**0.73**

The evaluation scores in bold indicate our higher results compared to the baseline

6 Future work and conclusion

This study represents a transformer-based approach to enhance sentiment prediction in code-mixed informal text. We implemented multilingual transformers like cm-BERT, mBART, and Electra. Fine-tuning these models with a focus on parameters like top-k, top-p, and temperature aimed to improve SA accuracy and f1-score. With the help of fine-tuning, our proposed research study outperformed the baseline. We also extended our analysis to binary classification where we used LDA topic modeling to gain a deeper insight into our dataset. LDA not only helped us to unveil the hidden details in our data but also improved our models' ability to differentiate between positive and negative sentiments. In the future, we are motivated to deal with various informal texts in different languages, including those with limited resources. We will use advanced transformers such as mT5, GPT models, and others. Our goal will be to solve problems related to multi-label and multi-classification issues across diverse languages, including Urdu, RU, Norwegian, Danish, and more.[3]

Acknowledgements This research work has been acknowledged by SOCYTI.[3] The SOCYTI project has received funding from the Research Council of Norway as a Researcher Project for Technological Convergence related to Enabling Technologies under grant agreement no 331736.

Author Contributions Ehtesham Hashmi: Conceptualization, Data Analysis, Formal Analysis, Research Execution, Design of Methods, Resources, Software, Writing Original Draft, Investigation. Sule Yildirim Yayilgan: Visualization, Supervision, Project Management, Funding Acquisition, Research Conduct, Validation. Sarang Shaikh: Visualization, Research Conduct, Validation.

Funding Open access funding provided by NTNU Norwegian University of Science and Technology (incl St. Olavs Hospital - Trondheim University Hospital).

Data Availibility Statement The datasets employed in this research are publicly accessible, and their respective references are provided in the manuscript.

[3] https://www.bigdata.vestforsk.no/ongoing/socyti

Declarations

Conflict of interest The authors declare that they have no Conflict of interest.

Informed Consent Not Applicable.

References

Ahmad GI, Singla J (2022) (lisacmt) language identification and sentiment analysis of english-urdu 'code-mixed' text using lstm. In: 2022 international conference on inventive computation technologies (ICICT), IEEE, pp 430–435

Alaparthi S, Mishra M (2020) Bidirectional encoder representations from transformers (bert): a sentiment analysis odyssey. eprint2007.01127

Ali H, Hashmi E, Yayilgan Yildirim S et al (2024) Analyzing amazon products sentiment: a comparative study of machine and deep learning, and transformer-based techniques. Electronics 13(7):1305

Altaf A, Anwar MW, Jamal MH, et al (2023) Exploiting linguistic features for effective sentence-level sentiment analysis in urdu language. Multimedia Tools and Applications pp 1–27

Cañete J (2019) Compilation of large spanish unannotated corpora. Zenodo, mayo de

Cao Y, Sun Z, Li L et al (2022) A study of sentiment analysis algorithms for agricultural product reviews based on improved bert model. Symmetry 14(8):1604

Clark K, Luong MT, Le QV, et al (2020) Electra: Pre-training text encoders as discriminators rather than generators. eprint2003.10555

Devlin J, Chang MW, Lee K, et al (2019) Bert: Pre-training of deep bidirectional transformers for language understanding. eprint1810.04805

Dominic P, Purushothaman N, Kumar ASA, et al (2023) Multilingual sentiment analysis using deep-learning architectures. In: 2023 5th international conference on smart systems and inventive technology (ICSSIT), IEEE, pp 1077–1083

Enríquez MP, Mencía JA, Segura-Bedmar I (2022) Transformers approach for sentiment analysis: classification of mexican tourists reviews from tripadvisor

Fuadi M, Wibawa AD, Sumpeno S (2023) idt5: indonesian version of multilingual t5 transformer. eprint2302.00856

Gan J, Qi Y (2021) Selection of the optimal number of topics for lda topic model-taking patent policy analysis as an example. Entropy 23(10):1301

Haque TU, Saber NN, Shah FM (2018) Sentiment analysis on large scale amazon product reviews. In: 2018 IEEE international conference on innovative research and development (ICIRD), IEEE, pp 1–6

Hasan M, Rahman A, Karim MR, et al (2021) Normalized approach to find optimal number of topics in latent dirichlet allocation (lda). In: Proceedings of International Conference on Trends in Computational and Cognitive Engineering: Proceedings of TCCE 2020, Springer, pp 341–354

Hashmi E, Yayilgan SY (2024) Multi-class hate speech detection in the norwegian language using fast-rnn and multilingual fine-tuned transformers. Complex & Intelligent Systems pp 1–22

Hashmi E, Yayilgan SY, Yamin MM, et al (2024) Advancing fake news detection: Hybrid deep learning with fasttext and explainable ai. IEEE Access

Hedderich MA, Lange L, Adel H, et al (2020) A survey on recent approaches for natural language processing in low-resource scenarios. arXiv preprint arXiv:2010.12309

Hossain MR, Hoque MM, Siddique N (2023) Leveraging the meta-embedding for text classification in a resource-constrained language. Eng Appl Artifl Intell 124:106586

Hossain MR, Hoque MM, Siddique N et al (2023) Covtinet: covid text identification network using attention-based positional embedding feature fusion. Neural Comput Appl 35(18):13503–13527

Hossain MR, Hoque MM, Siddique N et al (2024) Aracovtexfinder: leveraging the transformer-based language model for arabic covid-19 text identification. Eng Appl Artif Intell 133:107987

Hu J, Zhang Q, Yin H (2023) Augmenting greybox fuzzing with generative ai. arXiv preprint arXiv:2306.06782

Husain F, Al-Ostad H, Omar H (2022) A weak supervised transfer learning approach for sentiment analysis to the kuwaiti dialect. In: Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP), pp 161–173

Ilyas A, Shahzad K, Kamran Malik M (2023) Emotion detection in code-mixed roman urdu-english text. ACM Trans Asian Low-Resour Langu Inform Process 22(2):1–28

Javdan S, Minaei-Bidgoli B, et al (2020) Applying transformers and aspect-based sentiment analysis approaches on sarcasm detection. In: Proceedings of the second workshop on figurative language processing, pp 67–71

Javed I, Saeed H (2023) Opinion analysis of bi-lingual event data from social networks. 2023 5th international congress on human-computer interaction. Optimization and robotic applications (HORA), IEEE, pp 1–6

Jiménez-Zafra SM, Garcıa-Baena D, Garcıa-Cumbreras MA, et al (2023) Sinai at financesiberlef2023: Evaluating popular tools and transformers models for financial target detection and sentiment analysis. In: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023), co-located with the 39th Conference of the Spanish Society for Natural Language Processing (SEPLN 2023), CEUR-WS. org

Khan L, Amjad A, Afaq KM et al (2022) Deep sentiment analysis using cnn-lstm architecture of English and roman urdu text shared in social media. Appl Sci 12(5):2694

Liu Y, Gu J, Goyal N, et al (2020) Multilingual denoising pre-training for neural machine translation. eprint2001.08210

Mifrah S, Benlahmar E (2020) Topic modeling coherence: a comparative study between lda and nmf models using covid'19 corpus. Int J Adv Trends Comput Sci Eng 15:5756–5761

Muhammad KB, Burney SA (2023) Innovations in urdu sentiment analysis using machine and deep learning techniques for two-class classification of symmetric datasets. Symmetry 15(5):1027

Nagra AA, Alissa K, Ghazal TM et al (2022) Deep sentiments analysis for roman urdu dataset using faster recurrent convolutional neural network model. Appl Artif Intell 36(1):2123094

Naseem U, Razzak I, Musial K et al (2020) Transformer based deep intelligent contextual embedding for twitter sentiment analysis. Future Gener Comput Syst 113:58–69

Pipalia K, Bhadja R, Shukla M (2020) Comparative analysis of different transformer based architectures used in sentiment analysis. In: 2020 9th international conference system modeling and advancement in research trends (SMART), IEEE, pp 411–415

Qureshi MA, Asif M, Khan MF, et al (2023) Roman urdu sentiment analysis of songs 'reviews

Rahman MM, Islam MN (2021) Exploring the performance of ensemble machine learning classifiers for sentiment analysis of covid-19 tweets. In: sentimental analysis and deep learning: proceedings of ICSADL 2021. Springer, p 383–396

Rizwan H, Shakeel MH, Karim A (2020) Hate-speech and offensive language detection in roman urdu. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP), pp 2512–2522

Shakeel MH, Karim A (2020) Adapting deep learning for sentiment classification of code-switched informal short text. In: Proceedings of the 35th annual ACM symposium on applied computing, pp 903–906

Taherdoost H, Madanchian M (2023) Artificial intelligence and sentiment analysis: a review in competitive research. Computers 12(2):37

Tinn R, Cheng H, Gu Y, et al (2021) Fine-tuning large neural language models for biomedical natural language processing. eprint2112.07869

Valle-Cruz D, López-Chau A, Sandoval-Almazán R (2022) Review on the application of lexicon-based political sentiment analysis in social media. In: handbook of research on opinion mining and text analytics on literary works and social media. IGI Global, p 1–21

Vaswani A, Shazeer N, Parmar N, et al (2023) Attention is all you need. eprint1706.03762

Wilie B, Vincentio K, Winata GI, et al (2020) Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. eprint2009.05387

Xu QA, Chang V, Jayne C (2022) A systematic review of social media-based sentiment analysis: emerging trends and challenges. Decis Analyt J 3:100073

Yang H, Li J, Chen S (2023) Topicrefiner: coherence-guided steerable lda for visual topic enhancement. IEEE Trans Visuali Comput Graph 13:203

Younas A, Nasim R, Ali S, et al (2020) Sentiment analysis of code-mixed roman urdu-english social media text using deep learning approaches. In: 2020 IEEE 23rd international conference on computational science and engineering (CSE), IEEE, pp 66–71

Zhang W, Li X, Deng Y, et al (2022) A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. IEEE Transactions on Knowledge and Data Engineering

Zhao WX, Zhou K, Li J, et al (2023) A survey of large language models. eprint2303.18223