An Ensemble-based Model for Sentiment Analysis of Kurdish Tweets

Sabat S. Muhamad[1], Abdulhady A. Abdullah[2†], Hakem Beitollahi[1], Shamal A. Abdullah[3], Rezhin S. Sleman[1] and Ashna D. Zrar[1]

[1]Department of Computer Science, Faculty of Science, Soran University,
Soran, Kurdistan Region – F.R. Iraq

[2]Artificial Intelligence and Innovation Centre, University of Kurdistan Hewlêr,
Erbil, Kurdistan Region – F.R. Iraq

[3]Department of English, Faculty of Arts, Soran University,
SoranKurdistan Region – F.R. Iraq

***Abstract*—Thousands of comments are generated daily on social media in the Kurdistan Region. Sentiment analysis (SA) of these comments is valuable for organizations. The Kurdish language has three main dialects: Sorani (Central), Northern, and Southern. This study focuses on Sorani SA, where existing methods have limited accuracy. The proposed ensemble combines diverse models to improve sentiment classification. Preprocessing and word embedding using Roberta is the first phase of the method. The second phase consists of four proposed models, namely K-nearest neighbor, support vector machine, multilayer perceptron long short-term memory (LSTM), and bidirectional-LSTM (Bi-LSTM), which are used as classifiers. Finally, the ensemble weighted averaging technique is utilized to generate the final classification. To evaluate the performance of the proposed model, a dataset including 24211 unbalanced Soran tweets is first used, and after balancing, the dataset is used. The Bi-LSTM model attained an accuracy of 89.87% on the balanced dataset, and the proposed ensemble method increased the accuracy to 91.76%, which is better than the established state-of-the-art methods of Kurdish SA.**

***Index Terms*—Deep learning, Ensemble method, Kurdish language, Machine learning, Roberta word embedding, Sentiment analysis.**

I. Introduction

Sentiment analysis (SA) is a widely utilized technique for evaluating user reviews and comments on digital platforms. Organizations utilize sales automation to market products, identify new customers, and uphold their image.

SA (opinion mining) is a method in the field of natural language processing (NLP) that supports determining the feeling conveyed in a text, whether it is positive, negative, or neutral (Esmaili et al., 2013). The approach plays a vital role in extracting subjective information about the material posted by users (Bordoloi and Biswas, 2023).

Depending on a particular level, SA could be document level (the evaluations of the overall sentiment of a whole document), sentence level (evaluation of a single sentence), or aspect level (assessment of opinions about a particular attribute) (Medhat, Hassan and Korashy, 2014).

Regarding the methodology, SA may rely either on lexicon-driven methods, machine learning (ML) methods, or a combination of both. Its uses are extensive, and it can be applied to business, where it can be used to monitor customer reaction, and politics, where it can be used to gauge the mood of the population (Medhat, Hassan and Korashy, 2014).

Deep learning (DL) has developed as a compelling domain within ML during the past decade. Initially, DL was popularized in image processing and was rapidly applied to voice, music, and NLP (Sarker, 2021).

Relying ultimately on dictionaries and traditional ML approaches, conventional text categorization techniques have been widely replaced by robust DL methods such as sequence-based recurrent neural networks (RNNs) (Yu et al., 2019), including long short-term memory (LSTM) (Pouyanfar, et al., 2018). RNNs possess an internal memory, rendering them effective for sequential data such as texts. LSTM employs a gating mechanism with input, forget, and output gates to tackle long-term knowledge retention and the vanishing gradient issue. Thus, the capacity of LSTMs to extract sophisticated textual information is vital in the classification of text, and they have been rapidly utilized to a higher extent (Sumit et al., 2018).

Tailored to individual languages, SA algorithms frequently encounter difficulties stemming from cultural disparities, linguistic complexities, word order, and contextual language factors. However, the majority of models work for English text (Shakeel, et al., 2020) analysis, similar studies have been suggested for Spanish (Paredes-Valverde, et al., 2017), Thai (Vateekul and Koomsubha, 2016), and Persian (Roshanfekr,

ARO-The Scientific Journal of Koya University
Vol. XIII, No.2(2025), Article ID: ARO.12255. 10 pages
DOI: 10.14500/aro.12255
Received: 07 May 2025; Accepted: 24 October 2025
Regular research paper; Published: 11 December 2025
†Corresponding author's e-mail: abdulhady.abas@ukh.edu.krd

Khadivi and Rahmati, 2017). Research has employed polarity-based sentiment DL models to examine tweets.

Kurdish is from the Indo-Iranian branch, which is an Indo-European language. Sorani, also called Central Kurdish, is spoken in the north of Iraq and some parts of western Iran. Several studies have concentrated on NLP for the Central Kurdish language, encompassing spell-checking, stemming (Mustafa and Rashid, 2018; Jaf and Ramsay, 2014), and the development of the Kurdish Language Processing Toolkit (Ahmadi, 2020). The toolbox comprises text preprocessing, tokenization, stemming, transliteration, lemmatization, and an n-gram-based document classifier (Mohammed, et al., 2012). Minor initiatives have been undertaken to develop a vocabulary and corpus for Kurdish (Walther and Sagot, 2010).

In this article, we first combine two Central Kurdish corpora to create a SA system. A hard voting ensemble model consisting of five machine/DL models is applied to both balanced and unbalanced. Previous studies have not utilized an ensemble model to analyze the sentiment of the Kurdish language. The Kurdish Roberta model is employed as a word embedding approach for this purpose. Subsequently, K-nearest neighbor (KNN), support vector machine (SVM), multilayer perceptron (MLP), and LSTM are used as classifiers for Kurdish SA. The primary contributions of this paper are as follows:

- Comprehensive comparison of text categorization methods: The research concentrates on text categorization in the Kurdish language, specifically employing ML and DL algorithms. It assesses conventional classifiers and contrasts them with the Bidirectional Encoder Representations from the Roberta model to identify the most effective model for categorizing both unbalanced and balanced Kurdish text.
- New ensemble-based SA model: We present a novel ensemble-based SA model for Sorani Kurdish. Using a weighted averaging approach, the model combines the strengths of RoBERTa word embeddings with different classifiers (KNN, SVM, MLP, LSTM, and bidirectional-LSTM [Bi-LSTM]). This approach is designed to address the challenges of SA in low-resource languages by leveraging pre-trained language models and diverse classification algorithms.

The paper is organized as follows: Section 2 presents literature evaluations on word embedding and SA. Section 3 elucidates the context necessary to validate the methodologies employed in this paper. Section 4 elucidates our experimental findings. Section 5 presents the conclusions and future directions.

II. Literature Review

This section emphasizes some previous strategies that have demonstrated efficacy in addressing low-resource languages in certain sectors. A summary of these strategies is provided in Table I.

Travel evaluations represent a significant domain for SA since they assist individuals in making educated choices regarding accommodation. Automated review summarizations can improve the analysis of traveler reviews. Tsai, et al. (2020) proposed a systematic methodology employing Logistic Regression, Random Forest, Decision Tree, and SVM to extract pertinent text features and provide concise summaries.

Furthermore, few studies on SA have been conducted on Arabic. Abdullah and Shaikh (2018) employed Word2vec as a linguistic model. They used a dense network and an LSTM DL model, which achieved an accuracy of 77.7%. In another empirical study, CNN and LSTM as classifiers performed with an accuracy of 65% (Heikal, Torki and El-Makky, 2018). Al-Smadi, et al. (2018) employed an RNN classifier, resulting in 87% accuracy.

Abdulla and Hama. (2015) utilized a Naive Bayes classifier for SA among the Kurdish, categorizing sentiments into two classes: “Positive” and “Negative.” All papers and content were collected from social media platforms. After that, each individual received a score of one for positive emotions and a score of zero for negative emotions in a dataset. Later, a prototype Kurdish bag of words was developed for SA, which achieved an accuracy of 66%.

Authors in Chouikhi, Chniter and Jarray (2021) used a method that is based on the Arabic BERT tokenizer, to improve emotion classification accuracy on the Arabic dialect and modern standard Arabic. The model outperformed the existing methodologies, reaching the greatest accuracy on AJGT and ArSenTD-Lev datasets. The performance of hyperparameter adjustment is 96.11.

The largest sentiment-annotated Dialectal Arabic (DA) corpus present in the Gulf region, presented by Alowisheq, et al. (2021) contained 61,353 hand-tagged tweets in total, containing 840,000 tokens. To create a multi-domain collection, tweets were culled based on hot hashtags in four areas, namely, political, social, sports, and technology. This dataset was important to allow studying the domain-specific Arabic SA. In addition, the annotators also determined the terms of indicators to generate effect lexicons in every domain. They get information about the contextual polarity of some words based on the lexica.

In another study, Amin, Al-Rassam and Faeq (2022) emphasized the difficulties and challenges of Kurdish SA, as dialectal variation and limited linguistic resources. They suggested that constructing a comprehensive corpus would utilize precise transliteration mapping and employ a hybrid methodology. This integrates ML with lexicon-based techniques to enhance sentiment classification efficacy in Kurdish. Meanwhile, Awlla and Veisi (2022) developed a SA system using a Kurdish Word2Vec model and a diverse corpus that comprised 300 million tokens. They used LSTM as a classifier and collected 18,450 comments to enhance the Kurdish SA beyond expectations.

Badawi (2023) started to create the Kurdish Multilabel Emotional Dataset (KMD) for the Sorani dialect. This dataset includes emotional labels divided into four categories: Fear, sadness, joy, and surprise. The multilingual BERT model outperforms traditional approaches based on the experiments that implemented conventional ML classifiers and DL models.

Mahmud, Abdalla and Faraj (2023) worked to address the English-Kurdish linguistic disparity in SA of social media texts. They developed and annotated a new Kurdish SA dataset, evaluated many ML algorithms, and contrasted the outcomes using DL methodologies such as ANN, LSTM, and CNN. Naïve Bayes attained the highest performance, achieving an accuracy of 78%.

Ashraf, et al. (2023) designed a DL framework of the Urdu text SA (Urdu-BERT) based on bidirectional encoder representations of transformers and an Urdu dataset SA-23 (UDSA-23). It created BERT embeddings of every Urdu review. It uses the BERT embeddings to refine a DL classifier (BERT). The analysis of the results shows that USA-BERT significantly outperforms existing methodologies because it leads to an increase in the accuracy and f-measure by up to 26.09 and 25.87%, respectively.

The model created by Eyvazi-Abdoljabbar, et al. (2024) is an ensemble-based model that analyzes the Persian Instagram comments. The model was preprocessed and word-embedded with Word2Vec and used CNN, LSTM, CNN-LSTM, and LSTM-CNN as the classifiers. The Voting ensemble employed in the last step achieved a better result as it had an accuracy of 72.337% compared to the prior ensemble approaches, which had 67.439% accuracy.

Karim (2024) gathered 5,108 Central Kurdish comments from YouTube and Facebook to examine public attitudes toward Misyar marriage in Kurdistan. This dataset covers feelings and remarks, accompanied by preprocessing to enhance quality.

Badawi, Kazemi and Rezaie (2024) presented KurdiSent, the first manually annotated dataset for Kurdish SA. The dataset pertaining to the Sorani dialect contains more than 12,000 cases categorized as positive, negative, or neutral. XLM-R outperformed all ML and DL classifiers, with an accuracy of 85%.

III. Methodology

A. The Central Kurdish SA Dataset

The corpus we used in this paper combined two Kurdish corpora previously collected by these authors (Awlla and Veisi, 2022) and Kurdisent (Badawi, Kazemi and Rezaie, 2024). The merged dataset was unbalanced, exhibiting a significant bias toward Sentiment 1 (Positive: 11,374 occurrences), followed by Sentiment 2 (Neutral: 7,208 instances), with Sentiment 0 (Negative: 5,629 instances). This disparity might result in skewed model predictions, wherein the classifier may preferentially align with the predominant class (Sentiment 1) and exhibit subpar performance in the marginalized classes. Fig. 1 shows the combination of unbalanced datasets.

We employed undersampling to create a balanced dataset as a countermeasure against the class imbalance in our data and to prevent potential bias in our models. As indicated in Fig. 2, we reduced the instances in the dominant classes. Specifically, we reduced the instances of Sentiment 1 (positive) from 11,374 to 6,500 and Sentiment 2 (neutral) from 7,208 to 6,300. The instances of Sentiment 0 (negative) remained at 5,629. This undersampling operation resulted in a more balanced sentiment class representation.

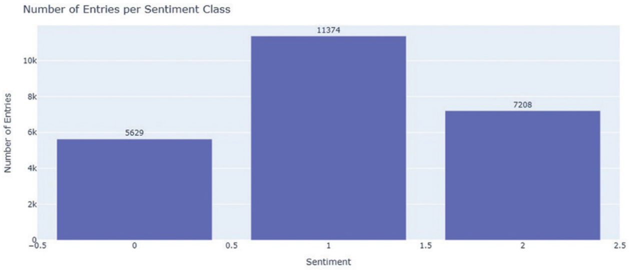


Fig. 1. Combining the two corpora.

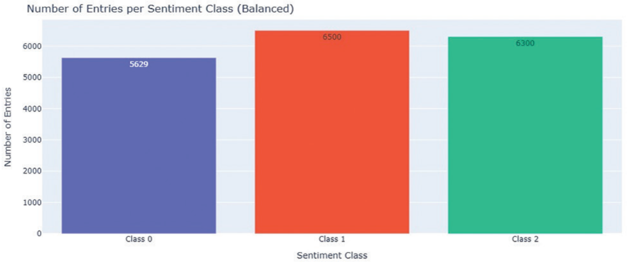


Fig. 2. Balancing the two corpora.

Corpus for the language model of central Kurdish

Acquiring data in the Kurdish language are particularly challenging due to a scarcity of resources, unlike most other languages. We invested significant work in sourcing enough text data for the Kurdish language so as to train the word vectors through Roberta. We gathered the texts from two sources to train Roberta and value their word vectors.

B. SA

Reberta model

Roberta necessitates training its tokenizer, the byte-pair encoding (BPE) tokenizer, which is developed using a substantial text corpus. Subsequently, five distinct Roberta models were trained, and finally, in the development of sentiment models utilizing all five classifiers: KNN, SVM, MLP, LSTM, and Bi-LSTM.

To mitigate the possible overfitting issue that is commonly linked with transformer-based approaches, especially during multi-epoch training, we have used a number of precautionary actions. On the one hand, we trained the RoBERTa model within a restricted number of epochs (10 in the case of word embeddings, and 5 in the case of sentiment models). Early termination was used, according to which the training stopped when the loss on valence ceased to decrease over two consecutive epochs. This was done to ensure that the model did not further train the patterns that fit the training data only.

Furthermore, the DL classifiers (LSTM and Bi-LSTM) were regularized with dropouts to minimize co-adaptation of the neurons. Another strategy we employed was 80/20 training/testing split and reported results only on the test

TABLE I
Present the Summary of Results from the Mentioned Study in the Written Text

References	Tokenizer	Dataset	Technique	Performance
Abdulla and Hama, 2015		15,000 Facebook, tweets, and Google	Naive Bayes classifier	66% accuracy
Abdullah and Shaikh, 2018	Word2vec	SemEval-2018 Task 1	LSTM	77.7% accuracy
Heikal, Torki and El-Makky, 2018	-	Arabic Sentiment Tweets Dataset (ASTD)	CNN and LSTM	65% F1
Al-Smadi, et al., 2018	-	Arabic Hotels Reviews Dataset (from SemEval-ABSA16), 24,028 ABSA annotated tuples	SVM and RNN	87% accuracy
Tsai, et al., 2020	-	TripAdvisor.com hotel reviews	Logistic Regression, Random Forest, Decision Tree, and SVM	70% accuracy and 80% AUC
Chouikhi, Chniter and Jarray, 2021	Arabic BERT	AJGT and ArSenTD-Lev	BERT	96.11% accuracy
Alowisheq, et al., 2021	-	MARSA (61,353 tweets)	-	-
Amin, Al-Rassam and Faeq, 2022		20000 websites, Facebook, and Twitter	SVM and Naïve Bayes	-
Awlla and Veisi, 2022	Word2Vec	18,450 Facebook comments	LSTM	71% accuracy
Badawi, 2023	Multilingual BERT	Kurdish Multilabel Emotional Dataset	Naive Bayes, SVM, BLSTM, and BERT	0.83% accuracy
Mahmud, Abdalla and Faraj, 2023	-	6,408 tweets	ANN, LSTM, CNN and Naïve Bayes	78% accuracy
Ashraf, et al., 2023	BERT	UDSA-23	Fine-tuning BERT	26.09% accuracy 25.87% F1
Karim, 2024	-	5,108 tweets	SVM	88% accuracy
Badawi, Kazemi and Rezaie, 2024	BERT	KurdiSent	Transformers (BERT)	85% accuracy
Eyvazi-Abdoljabbar, et al., 2024	Word2Vec	Insta.csv (Persian comments from Instagram)	CNN, LSTM, CNN-LSTM, LSTM-CNN. MLP and Voting Ensemble	72.337% accuracy

SVM: Support vector machine, MLP: Multilayer perceptron, LSTM: Long short-term memory

set. The fact that the accuracy and F1-score of the ensemble model are improved over the performance of the individual classifiers further indicates that the model did not overfit.

Central Kurdish Tokenizer

This study addresses the necessity of a tokenizer for the precise training of Roberta and sentiment models as presented by Abdullah, et al. (2024). Without a tokenizer, the model encounters unfamiliar words, causing out-of-vocabulary issues. For tokenizing sentences, BPE was utilized, and compound structures were properly, aligned with RoBERTa's methodology. Consequently, the annotation or labeling of the tokens was applied. The corpus is constructed by gathering and annotating 1,500 sentences from several sources. The corpus utilized an extensive array of tags for diverse named things, as delineated in Table II, while Table III presents a summary example of the annotated corpus, comprising two phrases with their respective tokens and tags. Moreover, Table IV presents the parameters of the Roberta model.

Sentiment models by Roberta word embedding

The training of the language model does not necessitate labels; nevertheless, the development of a sentiment model requires labeled data, as shown in Fig. 3. The clean labeled data are partitioned into two segments: 80% for training and 20% for testing. The article constructs a sentiment model with five prevalent approaches or classifiers (KNN, SVM, MLP, LSTM, and Bi-LSTM), with the Roberta model employed for testing each methodology.

Training SA on unbalanced versus balanced classes

When training a SA model, the distribution of class labels significantly impacts the performance and generalization of the model. Initially, our dataset was unbalanced, with a heavy skew toward Sentiment 1 (Positive: 11,374 instances), followed by Sentiment 2 (Neutral: 7,208 instances), and the least represented class being Sentiment 0 (Negative: 5,629 instances). This imbalance can lead to biased model predictions, where the classifier may favor the dominant class (Sentiment 1) and

TABLE II
THE Enumeration of Identified Entities Together with their Respective Tags (Abdullah, et al., 2024)

Tag	Description
B-art	Begin Artifact/entity
B-ani	Begin Animal's name
B-bird	Begin Bird's name
B-dat	Begin Date
B-eve	Begin Event
B-fruit	Begin Fruit's name
B-geo	Begin Geographic entity
B-gpe	Begin Geopolitical entity
B-nat	Begin Natural phenomenon
B-org	Begin Organization name
B-per	Begin Person's name
B-tim	Begin Time expression
B-vine	Begin Vine's name
B-Money	Begin Monetary value
B-num	Begin Number
O	Non-Named Entity
I-gpe	Included Geopolitical entity
I-eve	Included Event
I-dat	Included Date
E-org	Ended Organization name
I-num	Included Number
I-per	Included Person name

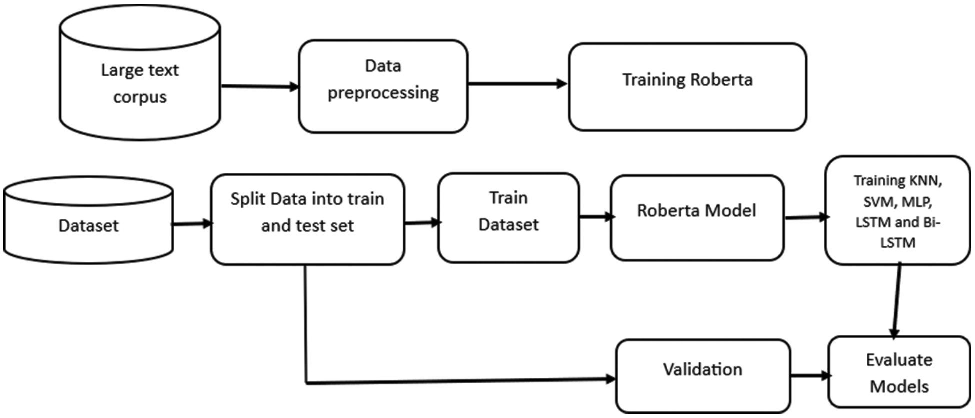


Fig. 3. Proposed structure of sentiment analysis for Central Kurdish using Roberta and the classifiers.

TABLE III
The Compilation of Annotated Tokens and their Corresponding Tags Inside Two Phrases (Abdullah, et al., 2024)

Sentence	Word	Tag
Sentence 1	تارا	B-per
Sentence 1	کتێبەکەی	O
Sentence 1	چاپ	O
Sentence 1	کرد	O
Sentence 2	مشک	B-ani
Sentence 2	لە	O
Sentence 2	پشیلە	B-ani
Sentence 2	دەترسێت	O

TABLE IV
Central Kurdish Roberta Model

Parameters	Values
epochs	10
Learning rate	3e-5
vocab_size	50,000
Loss	cross-entropy
batch_szie	16

perform poorly on the underrepresented classes.

To address this, we performed undersampling on the majority classes to create a more balanced dataset. Specifically, Sentiment 1 was reduced from 11,374 to 6,500 instances, and Sentiment 2 was reduced from 7,208 to 6,300 instances, while Sentiment 0 remained unchanged at 5,629 instances. This balancing ensures that the model does not learn from one sentiment category disproportionately.

KNN classifier

$$y_{pred} = mode\ (y_1, y_2, \ldots y_k)$$

KNNs is a non-parametric, simple algorithm. It works based on the majority class among k nearest neighbors. It does not require any deliberate training and uses distance metrics (e.g., Euclidean distance). For a given test point x, its predicted class is determined by the predominant class among the k nearest points in the training dataset.
Where:

- y_i is the label of the i-th nearest neighbor,
- k is the number of neighbors.

The distance between two points xi and xj is often calculated using Euclidean distance:

$$d(x_i, x_j) = \sqrt{\sum_{d=1}^{D} (x_{i,d} - x_{j,d})^2}$$

Where D is the dimensionality of the data.

In the current study, the k-NN model is evaluated using various k values. This evaluation achieved optimal performance at k=7 for both unbalanced and balanced datasets. The Kurdish text underwent preprocessing through normalization and tokenization, guaranteeing uniformity in feature extraction.

SVM classifier

The SVM works by determining the best hyperplane that can classify a set of data into different classes. The main idea is to determine a hyperplane that maximizes the distance between data that belong to different classes. In a binary classification problem, SVM aims at determining the optimal hyperplane that is defined by:

$$wx + \mathrm{b} = 0$$

Where:

- w is the weight vector (normal to the hyperplane),
- x is the input vector,
- b is the bias term.

SVM aims to optimize the margin M. The margin is defined as:

$$M = \frac{2}{w}$$

The optimization issue is then articulated as:

$$\min_{w,b} \frac{1}{2} \| w \|^2 \ \text{ subject to } y_i\,(\mathrm{w} \cdot x_i + b) \geq 1, \forall_i$$

Where y_i is the label for data point x_i.

In the current study, SVM is employed with a linear kernel due to its superior performance on textual datasets.

MLP classifier

MLP is a type of feedforward artificial neural network. Each neuron in a layer is connected with all neurons in the next layer, and this forms a fully connected network. MLP is applied in regression and classification. The output in an MLP with one hidden layer can be expressed in the following way:

$$h=\sigma(w_1\ x+b_1)$$

$$h=\sigma(w_2\ x+b_2)$$

Where:

- x is the input vector,
- w_1 and w_2 are weight matrices,
- b_1 and b_2 are bias vectors,
- σ is the activation function (typically ReLU for hidden layers and softmax/sigmoid for output).

The network is trained by backpropagation to reduce the loss function, commonly cross-entropy for classification purposes:

$$L=-\sum_{i=1}^{N} y_i \log\left(\hat{y}_i\right)$$

Where:

- N is the number of training examples,
- y_i is the true label,
- $\hat{y}_i$ is the predicted label.

We used the MLP classifier to predict positive, neutral, and negative sentiment labels for Sorani Kurdish SA. We employ a compact MLP network with two hidden layers and one output layer. ReLU activation is applied to the hidden layers, while Softmax activation is used in the final layer. Table V explains further facts regarding the hyperparameters of the networks.

LSTM classifier

LSTM is a type of recurrent neural network (RNN). LSTM addresses the vanishing gradient problem through gated memory cells. The basic equations in the case of LSTM are the following:

Forget gate:

$$f_t = \sigma\ (w_f\ [h_{t-1},x_t\ ]\ +\ b_f)$$

Input gate:

$$i_t = \sigma(w_i\ [h_{t-1},x_t]\ +\ b_i)$$

Candidate memory:

$$\tilde{c}_t = \tanh\left(w_c\left[h_{t-1},x_t\right]+b_c\right)$$

Output gate:

$$o_t = \sigma\ (W_o\cdot[h_{t-1},x_t]+b_0)$$

Memory cell update:

$$C_t = f_t\cdot C_{t-1} +\ i_t\cdot\tilde{C}_t$$

Hidden state:

$$h_t = o_t\cdot tanh\ (C_t)$$

Where:

TABLE V
MLP Sentiment Analysis Model Hyperparameters for Roberta Models

Hyperparameter	Description	Value
Hidden_layer_sizes	A tuple representing the number of neurons in each hidden layer	(256, 128)
Max_iter	Maximum number of iterations for training	500
Random_state	Seed used by the random number generator (ensures reproducibility)	42
Dropout-rate		0.3

- x_t is the input at time step ttt,
- h_{t-1} is the previous hidden state,
- C_{t-1} is the previous cell state,
- W_f,W_i,W_C,W_o are weight matrices,
- b_f,b_i,b_C,b_o are bias vectors.

Training LSTM networks consists of considerable hyperparameter tuning to achieve optimal performance. Furthermore, as the optimal value fluctuates based on the task, the parameters must be determined empirically. This study estimates the values of these parameters. The training parameters are detailed in Table VI.

Bi-LSTM classifier

Bi-LSTM is an improved version of LSTM that allows the model to absorb context in both back and forward sequences and therefore is of particular benefit to tasks whose contextual information in both directions is important, such as SA and machine translation. Bi-LSTM equations are similar to the LSTM, but the network includes two LSTMs: one that processes the sequence forward and the other one processes the sequence backward. The final state at every time period is the combination of the forward and backward hidden states:

$$h_t^{forward} = LSTM(x_t, h_{t-1}^{forward})$$

$$h_t^{backward} = LSTM(x_t, h_{t+1}^{backward})$$

The final hidden state h_t is:

$$h_t = [h_t^{forward}, h_t^{backward}]$$

Where:

- $h_t^{forward}$ is the hidden state from the forward LSTM,
- $h_t^{backward}$ is the hidden state from the backward LSTM.

BiLSTM is an alternative classification method. The weights from the labeled dataset are extracted from the Roberta model and subsequently supplied to the classifier

TABLE VI
LSTM Sentiment Analysis Model Hyperparameters for Roberta Models

Parameters	Values
Epochs	5
Activation	Sigmoid
Optimizer	Adam
Loss	binary_crossentropy
Drop_out	0.2

TABLE VII
Bi-LSTM Sentiment Analysis Model Hyperparameters for Roberta Models

Parameters	Values
Epochs	5
batch_szie	128
Activation	Sigmoid
Optimizer	Adam
Loss	binary_crossentropy
DropOut_Rate	0.2

Bi-LSTM: Bidirectional-LSTM

along with a label. Table VII elucidates further facts regarding the network's hyperparameters.

Ensemble method: Voting

The voting mechanism integrates the results of all classifiers. We used a hard voting strategy, wherein each of the classifiers (KNN, SVM, MLP, LSTM, and Bi-LSTM) votes one time towards its prediction of the sentiment class. The last prediction is the one that is given most of the votes. The result of this process is that any error committed by one classifier is offset by the right predictions of other classifiers, and hence the variance is reduced and the robustness is increased.

Where there is a tie (equal number of classifiers), the system chooses one of the tied classes randomly. This is an effective though simple mechanism since it combines the synergistic capabilities of varying classifiers. An example of this is that the traditional models (SVM and KNN) are good at processing linearly separable or instance-based cases, whereas DL models (LSTM and Bi-LSTM) are good at processing contextual and sequential features. These capabilities become integrated in the voting layer, which leads to a better overall performance and stability than when using any one classifier.

We employed a hard voting ensemble method to combine the predictions of our individual classifiers. For a given tweet, each classifier provides a sentiment prediction. The final prediction of the ensemble is made by selecting the majority class from all the classifiers' predictions. We randomly choose one of the tied classes in case of a tie.

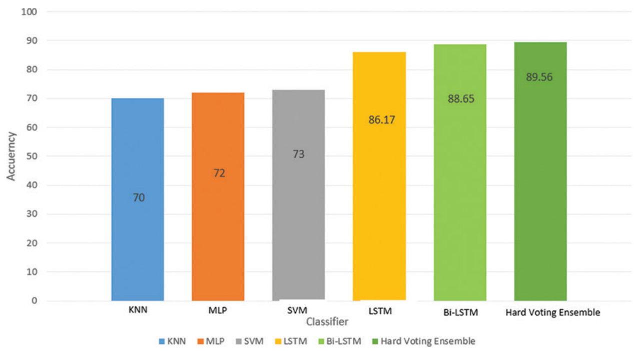


Fig. 4. Accuracy of diverse classifiers for the Roberta Model using the unbalanced dataset.

IV. Result and Discussion

This section presents our evaluation setup, metrics, and findings.

A. Evaluation Metrics

To evaluate the SA task, we employed accuracy and F1 Score as measures. The outcome of analyzing an instance by a SA system can be either:

- True Positive (TP): the system's forecast is positive, and the actual result is also positive.
- False Positive (FP): the system predicts a positive outcome when the actual value is negative.
- False Negative (FN): the system predicts a negative outcome when the actual value is positive.
- True Negative (TN): the system's forecast is negative, corresponding with the actual result.

In the context of a binary classification problem:

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN)$$

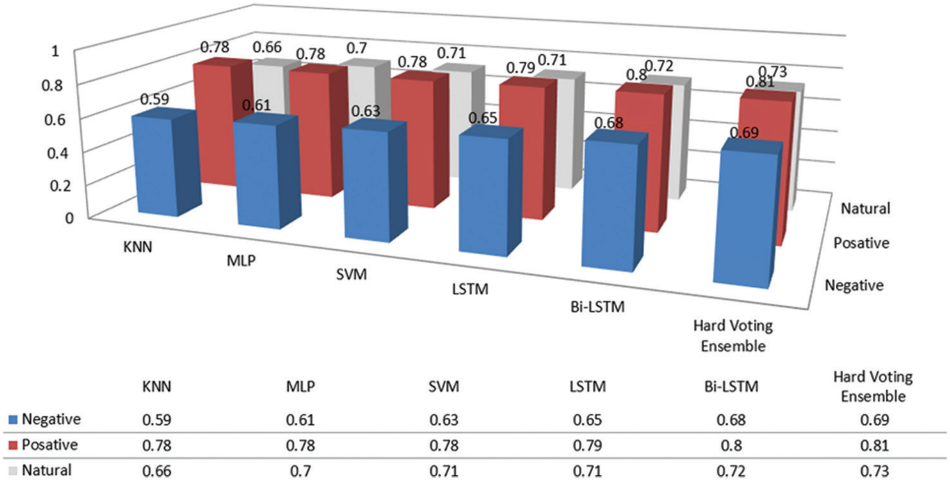
Accuracy is the predominant assessment metric for binary or multi-class classification problems, assessing the reliability of the solution by calculating the ratio of correct predictions to the total instances (Hossin, et al., 2011). For imbalanced data, the F-measure evaluates the impact of the unbalanced dataset on the models and computes the micro average.

$$\text{Precision } (p) = TP/(TP+FP)$$

$$\text{Recall } (r) = TP/(TP+FN)$$

$$\text{F1–score} = (2*p*r)/(p+r)$$

For multi-class classification problems, we report the *macro-averaged* F1-score. This gives equal weight to all classes, regardless of their frequency.



	KNN	MLP	SVM	LSTM	Bi-LSTM	Hard Voting Ensemble
■ Negative	0.59	0.61	0.63	0.65	0.68	0.69
■ Posative	0.78	0.78	0.78	0.79	0.8	0.81
■ Natural	0.66	0.7	0.71	0.71	0.72	0.73

Fig. 5. F1 measure of several classifiers for the Roberta Model across each class using an imbalanced dataset.

B. Implementation Environment

A high-performance computing setup for training models featuring two NVIDIA RTX 4090 GPUs with a combined memory capacity of 49 GB was utilized. Powerful GPU configuration can accelerate complex computations and facilitate efficient large dataset processing. This system was also equipped with 192 GB of RAM. These specifications ensure smooth multitasking and rapid data handling during intensive training sessions. Furthermore, a 4 TB hard drive provides ample storage for datasets, model checkpoints, and logs, which is crucial for maintaining quick access to data and supporting long training cycles.

C. Results of Roberta Word Embedding with Unbalanced Dataset

We first tested our models on the unbalanced dataset. The RoBERTa model, with training parameters are listed in Table III, was trained for about 2 days on our high-performance computing cluster, running 10 epochs and 1 million iterations. Fig. 4 shows the accuracy results for the five classifiers with RoBERTa embeddings as input features. The Bi-LSTM model achieved the best accuracy of 88.65%, with LSTM achieving an accuracy of 86.17%, SVM of 73%, MLP of 72%, and KNN of 70%. The ensemble model, applying the hard voting approach to average the five classifiers' outputs, had an accuracy of 89.56%, outperforming the individual classifiers.

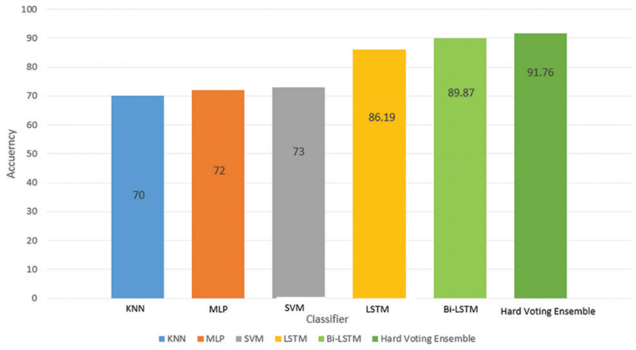


Fig. 6. Accuracy of diverse classifiers for the Roberta Model using a balanced dataset.

Fig. 5 shows the F1-scores of all classifiers on all the sentiment classes over the unbalanced dataset. Macro-averaged F1-scores were as follows: Bi-LSTM: 0.73, LSTM: 0.72, SVM: 0.71, MLP: 0.70, and KNN: 0.68. The macro-averaged F1-score was 0.74, even higher than the best among individual classifiers for the hard voting ensemble model.

D. Results of Roberta Word Embedding with a Balanced Dataset

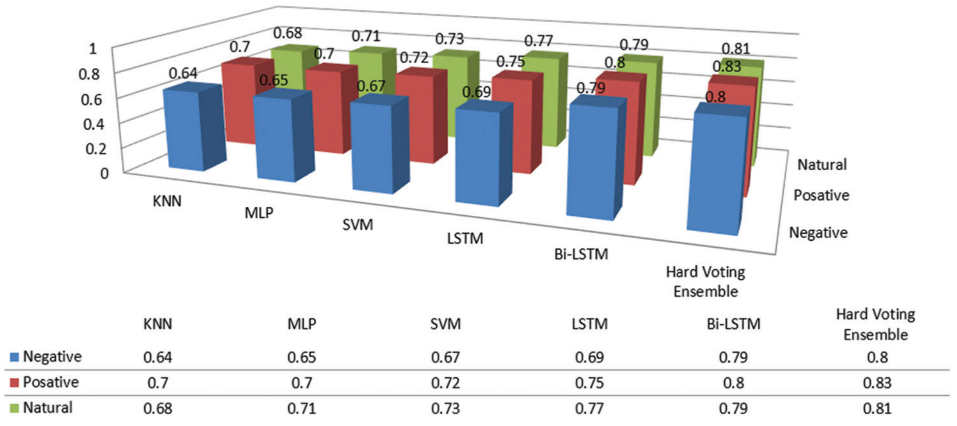
We examined model performance on the balanced dataset obtained through undersampling with the same training parameters (Table III). Fig. 6 shows the accuracy achieved by each classifier. Similar to performance on the unbalanced dataset, the Bi-LSTM model achieved the best accuracy at 89.87%. It was closely followed by LSTM at 86.19%, SVM at 73%, MLP at 72%, and KNN at 70%. The hard voting ensemble model achieved 91.76% accuracy on the balanced dataset.

Fig. 7 shows the F1-scores of each classifier for all sentiment classes over the balanced dataset. The macro-averaged F1-scores were Bi-LSTM: 0.79, LSTM: 0.74, SVM: 0.71, MLP: 0.68, and KNN: 0.68. The hard voting ensemble model achieved a macro-averaged F1-score of 0.81, which was greater than that of all individual classifiers.

E. Comparison between Results of Roberta Word Embedding with Balanced and Unbalanced Dataset

Our experiments show the effect of balancing on model accuracy. As shown in Figs. 5 and 7, the Bi-LSTM model had consistently high accuracy in both the unbalanced (88.65%) and balanced (89.87%) datasets. Dataset balancing improved the accuracy of the Bi-LSTM by 1.22%. The LSTM model showed a smaller increase, from 86.17% on the unbalanced dataset to 86.19% on the balanced dataset (an increase of 0.02%).

The hard voting ensemble model was even better than the performance of the best single model, with an accuracy of 89.56% on the unbalanced dataset and 91.76% on the balanced dataset. This is an improvement of 0.91% compared to the Bi-LSTM on the unbalanced dataset and an improvement of 1.89% on the balanced dataset.



	KNN	MLP	SVM	LSTM	Bi-LSTM	Hard Voting Ensemble
■ Negative	0.64	0.65	0.67	0.69	0.79	0.8
■ Posative	0.7	0.7	0.72	0.75	0.8	0.83
■ Natural	0.68	0.71	0.73	0.77	0.79	0.81

Fig. 7. F1 measure of several classifiers for the Roberta Model across each class using a balanced dataset.

TABLE VIII
Performance Comparison with Other Models Such As KuBERT

Model	Epochs	Batch	FP16	Total time (hh: mm)	Samples/s	Accuracy	F1-macro	ROC-AUC (micro)
SVM (embeds)	—	—	—	00:03	—	0.80	0.78	0.87
BiLSTM (embeds)	5	128	—	00:27	880	0.85	0.83	0.91
KuBERT (baseline)	5	32	✓	00:29	1,240	0.89	0.88	0.946
Proposed (Ours)	5	32	✓	00:27	1,230	**0.93**	**0.92**	**0.96**

F. KurdBERT (KuBERT) Model as a Baseline for Comparison

As indicated in Table VIII, experimenting with the proposal of an ensemble approach was done as a baseline by having the KurdBERT (KuBERT) model compare the performance of proposed ensemble approach on the Kurdish Sorani Twitter dataset by Wady et al. (2024). KuBERT showed high accuracy of 0.89, F1-macro of 0.88, and ROC-AUC (micro) of 0.946, which indicates high level of performance in Kurdish language perception. Nevertheless, the proposed ensemble model performed better when trained and evaluated on the identical data, where accuracy, F1-macro, and ROC-AUC (micro) were 0.93, 0.92 and 0.96, respectively. These findings affirm that the ensemble method not only generalize well across datasets but was also more accurate and robust than current Kurdish transformer models.

V. Conclusion and Future Work

This paper proposed a hard-voting ensemble using KNN, SVM, MLP, LSTM, and Bi-LSTM for Central Kurdish SA. RoBERTa was used for tokenization and word embedding. We received an unbalanced dataset and then generated a balanced dataset using undersampling. Bi-LSTM achieved 89.87% accuracy on the balanced dataset. The hard-voting ensemble improved accuracy to 91.76%. Future work will explore lemmatization-based data augmentation, which has improved results in other languages. In future research, we will use methods of data augmentation like lemmatization and stemming to increase the linguistic variety of input data. By adding bigger and more varied Kurdish corpora, the model generalization and the entire SA performance of low-resource languages, such as Kurdish, will be improved.

References

Abdulla, S., and Hama, M.H., 2015. Sentiment analyses for Kurdish social network texts using naive bayes classifier. *Journal of University of Human Development*, 1(4), pp.393-397.

Abdullah, A.A., Abdulla, S.H., Toufiq, D.M., Maghdid, H.S., Rashid, T.A., Farho, P.F., Sabr, S.S., Taher, A.H., Hamad, D.S., Veisi, H., and Asaad, A.T., 2024. *NER- RoBERTa: Fine-Tuning RoBERTa for Named Entity Recognition (NER) within Low-Resource Languages*. [arXiv Preprint].

Abdullah, M., and Shaikh, S., 2018. Teamuncc at Semeval-2018 Task 1: Emotion Detection in English and Arabic Tweets Using Deep Learning. In: *Proceedings of the 12th International Workshop on Semantic Evaluation*. pp.350-357.

Ahmadi, S., 2020. KLPT - Kurdish Language Processing Toolkit. In: *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*. pp.72-84.

Alowisheq, A., Al-Twairesh, N., Altuwaijri, M., Almoammar, A., Alsuwailem, A., Albuhairi, T., Alahaideb, W., and Alhumoud, S., 2021. MARSA: Multi-domain Arabic resources for sentiment analysis. *IEEE Access*, 9, pp.142718-142728.

Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., and Gupta, B., 2018. Deep recurrent neural network vs. Support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. *Journal of Computational Science*, 27, pp.386-393.

Amin, M.H.S.M., Al-Rassam, O., and Faeq, Z.S., 2022. Kurdish language sentiment analysis: Problems and challenges. *Mathematical Statistician and Engineering Applications*, 71(4), pp.3282-3293.

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

Awlla, K., and Veisi, H., 2022. Central Kurdish sentiment analysis using deep learning. *Journal of University of Anbar for Pure science*, 16(2), 119-130.

Badawi, S., 2023. KMD: A New Kurdish Multilabel Emotional Dataset for the Kurdish Sorani Dialect. In: *Proceedings of the 6th International Conference on Natural Language and Speech Processing (ICNLSP 2023)*. pp.308-315.

Badawi, S., Kazemi, A., and Rezaie, V., 2024. KurdiSent: A corpus for Kurdish sentiment analysis. *Language Resources and Evaluation*, pp.1-20.

Bordoloi, M., and Biswas, S.K., 2023. Sentiment analysis: A survey on design framework, applications and future scopes. *Artificial Intelligence Review*, 56, pp.12505-12560.

Chouikhi, H., Chniter, H., and Jarray, F., 2021. Arabic Sentiment Analysis Using BERT Model. In: *International Conference on Computational Collective Intelligence.* Springer International Publishing, Cham, pp.621-632.

Esmaili, K.S., Eliassi, D., Salavati, S., Aliabadi, P., Mohammadi, A., Yosefi, S., and Hakimi, S., 2013. Building a Test Collection for Sorani Kurdish. In: *2013 ACS International Conference on Computer Systems and Applications (AICCSA)*. IEEE, pp.1-7.

Eyvazi-Abdoljabbar, S., Kim, S., Feizi-Derakhshi, M.R., Farhadi, Z., and Mohammed, D.A., 2024. *An Ensemble-based Model for Sentiment Analysis of Persian Comments on Instagram Using Deep Learning Algorithms*. IEEE Access.

Heikal, M., Torki, M., and El-Makky, N., 2018. Sentiment analysis of Arabic tweets using deep learning. *Procedia Computer Science*, 142, pp.114-122.

Hossin, M., Sulaiman, M.N., Mustapha, A., Mustapha, N., and Rahmat, R.W., 2011. A Hybrid Evaluation Metric for Optimizing Classifier. In: *2011 3rd Conference on Data Mining and Optimization (DMO)*. IEEE, pp.165-170.

Jaf, S., and Ramsay, A., 2014. Stemmer and a POS Tagger for Sorani Kurdish. In: *6th International Conference on Corpus Linguistics*.

Karim, S.H.T., 2024. Kurdish social media sentiment corpus: Misyar marriage perspectives. *Data in Brief*, 57, p.110989.

Mahmud, D., Abdalla, B.A., and Faraj, A., 2023. Twitter sentiment analysis for Kurdish language. *Qalaai Zanist Journal*, 8(4), pp.1132-1144.

Medhat, W., Hassan, A., and Korashy, H., 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4), pp.1093-1113.

Mohammed, F.S., Zakaria, L., Omar, N., and Albared, M.Y., 2012. Automatic Kurdish SORANi Text Categorization using N-Gram based Model. In: *2012 International Conference on Computer and Information Science (ICCIS)*. Vol. 1, IEEE, pp.392-395.

Mustafa, A.M., and Rashid, T.A., 2018. Kurdish stemmer preprocessing steps for improving information retrieval. *Journal of Information Science*, 44(1), pp.15-27.

Paredes-Valverde, M.A., Colomo-Palacios, R., Salas-Zárate, M.D.P., and Valencia-García, R., 2017. Sentiment analysis in Spanish for improvement of products and services: A deep learning approach. *Scientific Programming*, 2017(1), p.1329281.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M.P., Shyu, M.L., Chen, S.C., and Iyengar, S.S., 2018. A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys*, 51(5), pp.1-36.

Roshanfekr, B., Khadivi, S., and Rahmati, M., 2017. Sentiment Analysis Using Deep Learning on Persian Texts. In: *2017 Iranian Conference on Electrical Engineering (ICEE)*. IEEE, pp.1503-1508.

Sarker, I.H., 2021. Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, 2(6), p.420.

Shakeel, M.H., Faizullah, S., Alghamidi, T., and Khan, I., 2020. Language Independent Sentiment Analysis. In: *2019 International Conference on Advances in the Emerging Computing Technologies (AECT)*. IEEE, pp.1-5.

Sumit, S.H., Hossan, M.Z., Al Muntasir, T., and Sourov, T., 2018. Exploring Word Embedding for Bangla Sentiment Analysis. In: *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*. IEEE, pp.1-5.

Tsai, C.F., Chen, K., Hu, Y.H., and Chen, W.K., 2020. Improving text summarization of online hotel reviews with review helpfulness and sentiment. *Tourism Management*, 80, p.104122.

Vateekul, P., and Koomsubha, T., 2016. A Study of Sentiment Analysis Using Deep Learning Techniques on Thai Twitter Data. In: *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. IEEE, pp.1-6.

Wady, S.H., Badawi, S., and Kurt, F., 2024. A Kurdish Sorani twitter dataset for language modelling. *Data in Brief*, 57, 110967.

Walther, G., and Sagot, B., 2010. Developing a Large-Scale Lexicon for a Less-Resourced Language: General Methodology and Preliminary Experiments on Sorani Kurdish. In: *Proceedings of the 7th SaLTMiL Workshop on Creation and Use of basic Lexical Resources for Less-Resourced Languages (LREC 2010 Workshop)*.

Yu, Y., Si, X., Hu, C., and Zhang, J., 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, 31(7), pp.1235-1270.