IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 June 2024, accepted 21 July 2024, date of publication 29 July 2024, date of current version 12 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3434664

RESEARCH ARTICLE

MLHS-CGCapNet: A Lightweight Model for Multilingual Hate Speech Detection

**ABIDA KOUSAR[1], JAMEEL AHMAD[2], KHALID IJAZ[1],
AMR YOUSEF[3,4], (Member, IEEE), ZAFFAR AHMED SHAIKH[5,6], (Member, IEEE),
IKRAMULLAH KHOSA[7], (Senior Member, IEEE),
DURGA CHAVALI[8], (Senior Member, IEEE),
AND MOHD ANJUM[9]**

[1]Department of Artificial Intelligence, School of Systems and Technology, University of Management and Technology, Lahore 54770, Pakistan
[2]Department of Computer Science, School of Systems and Technology, University of Management and Technology, Lahore 54770, Pakistan
[3]Electrical Engineering Department, University of Business and Technology, Jeddah 23435, Saudi Arabia
[4]Engineering Mathematics Department, Alexandria University, Alexandria 21544, Egypt
[5]Department of Computer Science and Information Technology, Benazir Bhutto Shaheed University Lyari, Karachi 75660, Pakistan
[6]School of Engineering Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland
[7]Department of Electrical and Computer Engineering, COMSATS University Islamabad, Lahore Campus, Lahore 54000, Pakistan
[8]Trinity Health, Trinity Information Services, Livonia, MI 48152, USA
[9]Department of Computer Engineering, Aligarh Muslim University, Aligarh 202002, India

Corresponding author: Jameel Ahmad (jameel.ahmad@umt.edu.pk)

This work was supported by the University of Management and Technology, Lahore, Pakistan.

ABSTRACT The rapid advancement of computer technology and the widespread adoption of online social media platforms have inadvertently provided fertile ground for individuals with antisocial inclinations to thrive, ushering in a range of security concerns, including the proliferation of fake profiles, hate speech, social bots, and the spread of unfounded rumors. Among these issues, a prominent concern is the prevalence of hate speech within online social networks (OSNs). However, the relevance of numerous studies on hate speech detection has been limited, as they primarily focus on a single language, often English. In response, our research embarks on an exhaustive exploration of multilingual hate speech across 12 distinct languages, offering a novel approach by adapting hate speech detection resources across linguistic boundaries. This study presents the development of a robust, lightweight and multilingual hate speech detection model, known as MLHS-CGCapNet, which combines convolutional and bidirectional gated recurrent units with a capsule network. With commendable accuracy, recall and f-score values of 0.89, 0.80, and 0.84, respectively, our proposed model exhibits strong performance, even when handling an imbalanced dataset. Notably, during the training and validation phases, the suggested model showcases exceptional effectiveness, achieving accuracy values of 0.93 and 0.90, respectively, particularly in the challenging context of imbalanced data. In comparison to both baseline and state-of-the-art techniques, our model offers superior performance.

INDEX TERMS Hate speech detection, deep learning, BiGRU, social networks, capsule network.

I. INTRODUCTION

One of the most significant inventions in human history, the Internet, serves as a tool to connect people from diverse racial, religious, and ethnic backgrounds. Social media platforms such as Twitter and Facebook have successfully linked billions of individuals, offering them a swift means to express their thoughts and opinions. However, these platforms also come with negative consequences, including troubling issues like online harassment, trolling, cyberbullying, and the spread of hate speech. Hate speech, as defined by the National Institute of Standards and Technology (NIST), involves “communication that degrades an individual or a group based on characteristics such as race, ethnicity, gender, sexual orientation, nationality, religion, or other distinctive attributes.” Numerous publications have explored

The associate editor coordinating the review of this manuscript and approving it for publication was Manuel Rosa-Zurera.

the detection of hate speech within online social networks. As hate crimes continue to rise in various regions, there is an urgent need for a deep understanding of the mechanisms driving the dissemination of offensive content across these digital platforms.

Around 50% of the world's population, which translates to more than 3.8 billion people who use the internet regularly, prefer using text as their primary way of communication due to the increasing usage of social platforms, including blogging platforms. However, a small fraction of these users use offensive language or engage in hate speech, which is directed at specific groups such as those with common racial, ethnic, national, religious, or gender affiliations, sexual orientation, caste, or those with significant illness or handicap [1].

Several studies in English have argued for automated hate speech detection. Most research made use of a machine learning algorithm with social media platform as a dataset source. Thousands of people, however, continue to violate the rules by using their Twitter accounts to spread hate speech and derogatory remarks. Because most datasets are only available in one language: English, identifying hate speech is a challenging task. It is challenging to research the effects of hate speech in other languages because most work is conducted only in English, which is a worrying situation. Even several social media platforms have algorithms operating in a few languages, this makes the size of available datasets very limited [2].

The surge in hate crimes in multiple states has underscored the pressing need for a more comprehensive understanding of the propagation of offensive language online. To address this objective, we have developed a model capable of identifying hate speech within social media posts. This study delves into the realm of multilingual hate speech, examining 12 different languages, including English, Turkish, Hindi, Italian, Spanish, Indonesian, German, Portuguese, Danish, Arabic, Malay, and French. Our investigation represents an extensive exploration of multilingual hate speech across these languages, specifically within tweets.

In this task, we approach it as a binary-class classification problem, utilizing a dataset of tweets, where each tweet falls into one of two categories: hate speech or non-hate speech. Researchers have increasingly turned to machine learning and deep learning methods in recent years to tackle this issue, with existing deep learning models showing promise in hate speech detection. However, despite the persistence of numerous hate speech-related tweets on Twitter, current models fall short in meeting the necessary criteria.

Academic and industry experts are employing statistical analysis, pattern mining, and deep learning techniques to combat the growing issue of hate content on social media platforms. Combining deep learning methods with a capsule network has the potential to enhance the effectiveness of hate speech identification further. This study introduces a novel deep neural network model to accurately distinguish between hateful and non-hate speech on Twitter. Our approach integrates a capsule network alongside convolutional and bi-directional gated recurrent units, simplifying the detection of hateful language in tweets. Unlike previous models, our proposed method leverages the capsule network to incorporate contextually relevant information from various perspectives.

II. RELATED-WORK

Developing a classifier that can accurately detect hate speech and subtle nuances in less-researched languages with limited data presents a formidable challenge. In their work [3], the authors introduce HateMAML, a model-agnostic meta-learning (MAML) framework designed to effectively address the issue of identifying hate speech in languages that suffer from resource constraints, aiming to bridge this research gap. To overcome the limitations imposed by scarce data, HateMAML employs a self-supervision technique, which results in improved initialization of language models. Essentially, this feature enables rapid adaptation to a foreign target language (cross-lingual transfer) or a wide array of datasets related to hate speech (domain generalization). The authors conduct a comprehensive set of experiments using five datasets and eight different low-resource languages. The results demonstrate that, in scenarios involving cross-domain multilingual transfer, HateMAML outperforms existing benchmarks by more than 3%. Ablation studies are also carried out to dissect the components of HateMAML.

In the prevalent utilization of online social media, a variety of Natural Language Processing (NLP) techniques, particularly those based on transformers, have been employed to detect instances of hate speech on the internet. These projects mostly focus on categorizing various forms of hate speech, including racism, sexism, and cyberbullying. However, the majority of these initiatives have been limited to specific languages, mostly English. Unlike more established NLP domains like sentiment analysis, which have benefited from cross-lingual learning strategies, the exploration of cross-lingual hate speech detection has remained comparatively limited (Pamungkas and Patti, [4]). Notably, Pamungkas and Patti have introduced Misogyny Detection in Twitter, a cross-lingual model that employs joint learning, capitalizing on the capabilities of multilingual HurtLex and MUSE embeddings. This model demonstrates notable superiority over alternatives that rely solely on monolingual embeddings [5]. Furthermore, diverse strategies embracing both multilingual and multi-aspect dimensions have been pursued to address hate speech. Additionally, cross-lingual contextual word embeddings are utilized for identifying offensive language, extending the analysis from English to encompass other languages [6].

The data for Warner and Hirschberg's study were drawn from Yahoo and the United States Jewish Congress [7]. Achieving an accuracy of 0.68, recall of 0.60, F1-score of 0.64, and a reliability rate of 95% in their best-performing instance, they obtained substantial results using SVM light classification. Kwok and Wang, in contrast, employed the Naive Bayes classifier and the Bag-of-Words feature selection technique to categorize tweets [8], [9], [10]. In their 10-fold cross-validation, their model's most favorable assessment yielded an accuracy of 76%.

The researchers found the Bag-of-Words (BOW) model inadequate for accurately categorizing tweets related to hate speech (HS). Although they achieved an acceptable accuracy using uni-gram features, they proposed enhancing accuracy by incorporating bi-gram features and the tweet's sentiment score into the feature set [11]. Davidson et al. collected data from a crowd-sourced platform and categorized it into three groups—HS, Offensive, and Neither—to build an automated model for HS identification. Subsequently, they extracted features from the labeled dataset by considering uni-, bi-, tri-, and quad-grams. To reduce dimensionality, they employed logistic regression with L1 regularization. Performance was evaluated through a 5-fold cross-validation using classifiers like decision trees, random forests, linear SVM, and naive Bayes [7]. Meanwhile, Badjatiya et al. harnessed word representations learned via deep learning models to develop Gradient Boosted Decision Tree classifiers.

Researchers also investigated character-level representations, evaluating their efficacy in comparison to word-level representations [1]. For a number of NLP (natural language processing) applications, deep neural network models have been utilized to demonstrate how well pre-processing can be combined with CNN-GRU networks. These models contain a number of different elements, such as a word embedding layer, 1D CNN, 1D max-pooling, GRU, global max-pooling, and a softmax layer [12], [13].

Empirical evidence by Zhang et al. has affirmed the effectiveness of CNN in text classification, while RNNs (as demonstrated in [reference]) and Bi-LSTMs have similarly demonstrated enhanced performance in this realm [11]. A model for spotting hate speech (HS) in user comments was developed by Djuric et al. They employed continuous bag of words (CBOW) and paragraph2vec techniques to represent the comments in a two-dimensional space. These representations were then input into a binary classifier to discriminate between hateful and non-hateful comments, with paragraph2vec getting the greatest average area under the curve of 0.80. Meanwhile, Park and Fung combined deep learning models with standard machine learning classifiers to categorize tweets using a logistic regression technique and a CNN model. The outcome was improved performance that was superior to that of the separate models [14], [15].

Kamble and Joshi constructed their own embeddings for detecting hate speech in code-mixed tweets, which contained both Hindi and English, using a substantial text dataset. Their trial demonstrated that the generated code-mixed embeddings surpassed the performance of pre-trained word embeddings. The experiment encompassed various classification models, including SVM, Random Forest, CNN-1D, LSTM, and Bi-LSTM models [16]. Contemporary deep learning (DL) architectures for text processing generally encompass a word-embedding layer, aimed at capturing the semantic essence of words by converting each word into a low-dimensional vector within the input sentence. In addition to this, Zhang et al. introduced a DL classification architecture that combines both convolutional and recurrent processing layers, showcasing remarkable performance in hate speech classification [1], [17].

Recently, the implementation of the state-of-the-art large language models (LLMs) has shown dominance over the conventional DL models for hate speech detection. GPT-3 incorporated with few-shot learning delineates the remarkable ability to identify sexist and racist text. However, GPT-3 was unable to acquire the desired performance on the diversified hate speed data [18]. To combat the above problem, "hot" ChatGPT was introduced to classify obnoxious content, such as toxic, offensive and hateful speech [19]. However, the performance of the aforementioned model was not up to the mark as "hot" ChatGPT failed to achieve an F1-score above 0.640. Afterward, GPT-3, T5 and prompts-based hate speech detection models surpassed the above LLMs model by achieving a high F1-score of 0.643 compared to the above-discussed LLMs [20], [21]. However, the existing state-of-the-art LLMs consist of a higher number of parameters to capture the context of hate speech effectively by fully utilizing the knowledge base, which makes state-of-the-art LLMs difficult to use in real-world scenarios where resources are limited. For instance, the GPT-3.5-turbo is considered as one of the advanced LLMs that uses 175 billion parameters to train [22]. To eradicate the above limitation, the focus is to develop a new model which consumes a smaller number of trainable parameters without compromising the accuracy of hate speech detection.

III. CONTRIBUTIONS

This work's contributions may be summed up as follows:

1) Our study addresses the challenge of detecting hate speech in multiple languages through a comprehensive approach that includes diverse language training and innovative model architecture. We emphasize the significance of linguistic diversity by using datasets in twelve languages, fostering model relevance.
2) Our model training leverages this multilingual dataset, empowering it to effectively identify hate speech across languages, patterns, and contexts. The innovation lies in our model's lightweight architecture, integrating CNNs, BiGRUs, and capsule networks for advanced feature extraction and contextual comprehension, accommodating linguistic variations. The primary contribution of our current research is the exploration of the hybrid model's capability in detecting hate speech.
3) The features detected by our proposed model have been visualized, and we have established the correlation between these features and the resulting performance of the model as proof of the method's correctness.
4) We have thoroughly compared the performance of our proposed CNN-BiGRU-CapsNet-based hate speech detection approach to that of pre-trained algorithms.

IV. RESEARCH METHODOLOGY

The same research framework has been implemented to perform extensive experiments on all 12 languages for hate speech detection. The steps of the research approach presented in this work are as follows:

A. DATASETS

To empirically assess the proposed approach in this study, the authors collected a dataset of 9 languages, such as Arabic, English, German, Indonesian, Italian, Polish, Portuguese, Spanish and French, which is publicly available [23]. To increase the diversity of the dataset, the data of 3 other languages, such as Danish, Turkish and Hindi has been extracted from the Twitter developer account using the Tweepy library. To the best of the author's knowledge, the suggested study is the first one to include 12 languages for the hate speech detection of social media platforms.

Collection and annotation of data for the training of hate speech detection classifiers is a demanding task. The literature did not find any universal definition of hate speech. A literature review has been conducted to determine the reliability of the annotation system. The literature review advocates that most of the annotation systems are unreliable and vary accordingly with the understandings of the annotators [24].

Furthermore, social media platforms are a main source of providing data related to hate speech detection systems, yet many have very stringent distribution policies and data usage. Eventually, the accessed data is relatively small to study, with most coming from Twitter (which has a more lenient data usage policy). While the Twitter resources are valuable, their general applicability is restricted due to the unique genre of Twitter posts; the character limitation results in terse, short-form text. Instead, the posts from other sources are typically longer and can be part of a larger discussion on a specific topic. This provides additional context that can affect the meaning of the text. The authors also faced difficulties with accessing and creating the Twitter developer account due to platform privacy concerns.

To empirically assess the proposed approach in this study, we utilized 12 datasets collected from Twitter, as detailed in the Table 1. The research “Deep Learning Models for Multilingual Hate Speech Detection” was conducted by Sai Saketh Aluru1, Binny Mathew1, Punyajoy Saha1, and Animesh Mukherjee2 at the Indian Institute of Technology Kharagpur, India, in the year 2020 [14]. It made use of 9 different data sets in nine distinct languages. In order to accomplish the objective of hate speech detection, we utilized dedicated datasets. For the languages English, Turkish, and Hindi we conducted data collection by extracting tweets from Twitter using a Twitter developer account, facilitated by the Tweepy library. Within this process, each individual tweet is categorized as either containing hate speech or not.

TABLE 1. Language datasets for study.

Language	Total	Non-Hate Speech	Hate Speech
English	81943	67735	14208
Arabic	4117	3649	468
Danish	3275	2850	425
French	1028	821	207
German	6962	5526	1436
Hindi	15000	12188	2812
Indonesian	13882	8061	5821
Italian	12116	8507	3609
Malay	1113	700	413
Portuguese	5668	4440	1228
Spanish	12600	8294	4306
Turkish	34794	28035	6757

In our research, we have leveraged a diverse hate speech detection dataset curated meticulously to encompass multiple languages and cultural contexts. Recognizing the global nature of online hate speech, our dataset spans a wide range of languages, including but not limited to English, Spanish, French, German, Arabic, Chinese, and more. This multilingual approach is pivotal in addressing the pressing need for a comprehensive understanding of hate speech across linguistic boundaries, facilitating the development of robust and inclusive detection models. By embracing linguistic diversity, we aim to shed light on the nuances and variations in hate speech across different linguistic and cultural landscapes, ultimately contributing to the creation of more effective and inclusive hate speech detection systems.

It is worth noticing that despite Chinese being the 2nd largest speaking language globally, it has been much less investigated in HS detection community. One reason could be the lack of Chinese language in HS competition workshop such as SemEval2020 [192] and Hasoc2020 [82] where multilingual tasks included English, Danish, Greek, Arabic, Tamil, Malayalam, Hindi, German and Turkish, which encouraged many participants all over the world to work in these languages.

Figure 1 depicts the percentage of different HS categories in the identified records. We can see that publications related to ’general hate’ (36) are a dominant trend followed by ’abusive language’ (23) of total records. Cyberbullying and Radicalization categories share the same percentage of (15) each. While relatively a small percentage is assigned to religion (5), racial (3), and sexism (3) associated hate speech categories.

- Pre-Processing: The model presented initially undertakes preprocessing of the collected tweets to eliminate irrelevant information. Figure 3 provides a structured depiction of the preprocessing steps. During the preprocessing phase, the proposed model executes the subsequent actions.
  - The first step was converting all text to lowercase, which helps eliminate any discrepancies due to capitalization.
  - separate images from Twitter-related markers like as hashtags, URLs, phrases, and retweets.
  - Sifting of genuine numbers, stop words, ampersands, repetitive void areas, dabs, single and

hate
jew
people
xenophobia
https
twitter
african
muslim
fear
arab
racism
christian
hate crime
discrimination
think
know
israel
status
one
pic twitter
country
say
african american
islam
want
gender
anti semitic
racist
asian
black
hate speech
trump
islamophobia
antisemitism
south african
hatred
religion
white
love
hating

FIGURE 1. Wordcloud of the dataset.

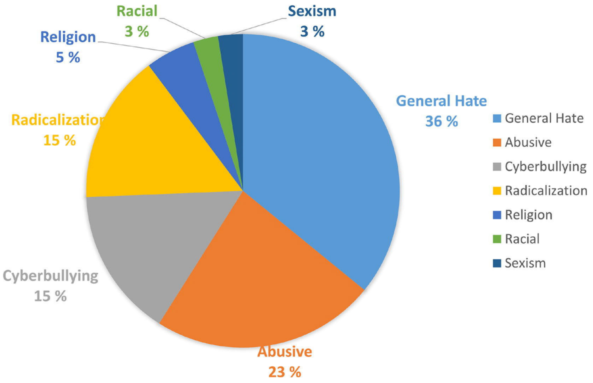

FIGURE 2. Bias distribution in the dataset.

twofold statements, non-ASCII characters, commas, emojis, interjection imprints, addition, and accentuation marks.

- Expulsion of every repetitive tweet.
- At long last, tweets are changed over into lower-case.

- During this phase, features are extracted using a deep-learning model that has already been trained. Keras and Tensorflow libraries provide us with all the essential tools for building a neural network for our system.

B. PROPOSED HATE SPEECH DETECTION ARCHITECTURE: MLHS-CGCAPNET

The same multi-layered architecture is used for the identification of hate speech detection for all 12 languages. The proposed MLHS-CGCapNet model's structure is shown in the MLHS-CGCapNet Figure 4. The multi-layered architecture of the model is examined in the following subsections.

1) INPUT LAYER

The input layer is responsible for generating unique token sequences for various colloquialisms, slang, and culturally specific terms of every language and text input to accommodate all linguistic variations through tokenization. Tokenization is the process of breaking down text into tokens, such as small words and characters. Then the input layer converts the token sequences into dense numerical representations (embeddings). The efficient tokenization process enhances the capability of the hate speech detection process.

2) EMBEDDING LAYER

The numerical representation of discrete token indices is transformed into dense numerical vectors by the embedding layer, which capture semantic nuances and contextual information pertinent to identifying hate speech. These vectors capture semantic meanings and token relationships in a continuous space. Text analysis and recommendation systems are only two examples of the many uses for this layer. We employed GloVe embeddings, a well-known method based on word co-occurrence statistics, for this investigation. GloVe generates word vector representations using co-occurrence matrices. We opt for Twitter-specific GloVe embeddings due to dataset origin. Each token in the input text that has been padded is converted into a GloVe embedding, forming an input matrix T with dimensions. The algorithm is presented in Algorithm. 1.

3) CONVOLUTIONAL LAYER

Three convolutional layers—conv1, conv2, and conv3—in a convolutional neural network (CNN) are used by the model to extract regional patterns and features from the input data. These layers use kernel sizes of 2, 3, and 4, sliding filters over token embeddings to create feature maps. These maps capture interactions among neighboring tokens, extracting spatial and temporal highlights associated with expressions of dislike. With 128 channels of varying sizes, the convolutional layers perform operations and derive a feature sequence fs from the input text. This sequence undergoes maximum pooling and concatenation, ultimately contributing to the subsequent layer's final feature vector.

$$fn = f(wtxt + b) \tag{1}$$

4) BIGRU LAYER

A bidirectional Gated Recurrent Unit (GRU), a specific kind of recurrent neural network (RNN) designed to capture sequential patterns and relationships within the data, is used in this layer. The GRU's bidirectional capability allows it to analyze input sequences both forward and backward, at the same time collecting data from the past and the future. In the BiGRU setup, a forward GRU processes succeeding sequences (f1 to f128), while a backward GRU handles preceding sequences (f128 to f1).

The combination of CNN and BiGRU in a proposed architecture is a key to boosting accuracy. The integration of CNN and BiGRU not only ensures the extraction of features but also finds unique trends and patterns in the contextual features of the various languages. Hence, the above-discussed

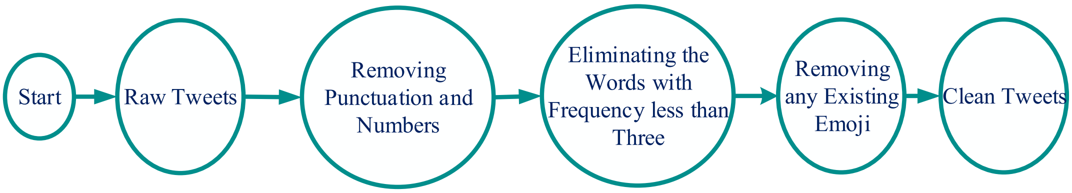
Start
Raw Tweets
Removing Punctuation and Numbers
Eliminating the Words with Frequency less than Three
Removing any Existing Emoji
Clean Tweets

FIGURE 3. Data Preprocessing Steps.

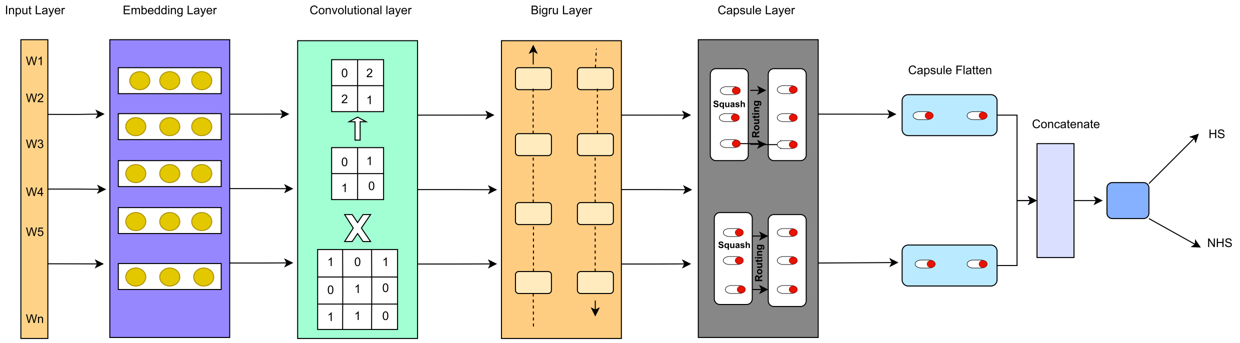

FIGURE 4. MLHS-CGCapNet model for multilingual hate speech detection.

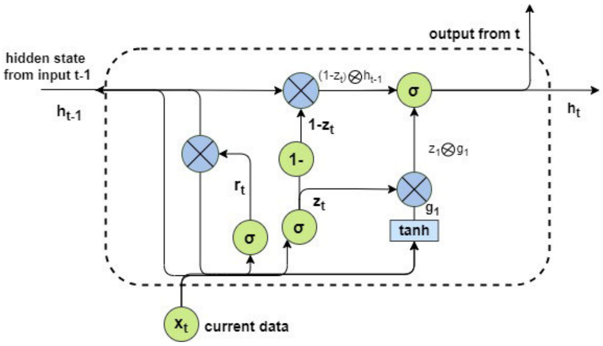

FIGURE 5. The internal structure of GRU Cell.

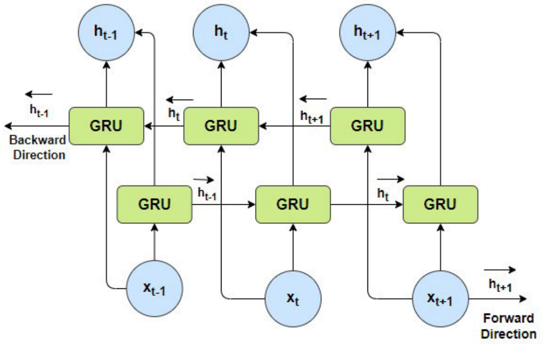

FIGURE 6. The internal structure of BiGRU Network.

combination provides extraordinary mapping between the contextual data and extracted features in the hate speech detection process.

$$ForwardHiddenState(h_f) = \mathrm{GRU}(L_{fs}), \quad n \in [1, 128] \tag{2}$$

$$BackwardHiddenState(h_b) = \mathrm{GRU}(L_{fs}), \quad n \in [128, 1] \tag{3}$$

$$TotalHiddenState(h_t) = [h_f, h_b] \tag{4}$$

C. GRU CELL

The basic model of GRU cell is shown in Figure 5 and BiGRU Network in Figure 6. The cell is equipped with three gates: the input gate (i_t), the forget gate (f_t), and the output gate (o_t). These gates serve the purpose of preserving and modifying information related to the data preceding time t. The cell states and updating the parameters is governed by Eq.5 to Eq.8.

The Gated Recurrent Unit (GRU) is a particular kind of recurrent neural network (RNN) cell created to detect long-range relationships in sequential input while minimising vanishing gradient issues. It is made up of a candidate activation ($\tilde{h}_t$) that computes the intermediary hidden state, an update gate (z_t), and a reset gate (r_t), which manage the movement of information, and the last hidden state to be computed (h_t), which acts as the cell's output.The equations for the GRU cell are as follows:

$$(updategate)z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{5}$$

$$(Resetgate)r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{6}$$

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \cdot h_{t-1}, x_t]) \tag{7}$$

$$(newhiddenstate)h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \tag{8}$$

Here, σ stands for the sigmoid activation function, tanhfor the hyperbolic tangent activation function, W_z, W_r, and W_h respectively, for weight matrices, h_{t-1} for the prior hidden state, and x_t for the input at time step t.The update gate z_t regulates how much of the previous hidden state and candidate activation are combined to create the new hidden state h_t, while the reset gate r_t regulates which portions of

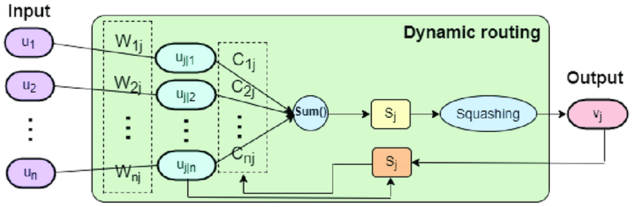

FIGURE 7. The internal structure of a Capsule Network with dynamic routing algorithm.

the previous hidden state are disregarded when calculating the candidate activation.

To capture context-rich sequences in both directions (forward as well as backward), the model applies a BiGRU layer to the result of the convolutional layer. The outputs of the BiGRU for forward and backward directions are represented by equations (2) and (3), respectively. The contextual representation of the input text is combined with information from both directions in the BiGRU output. This representation based on BiGRU combines the forward (hf) and backward (hb) hidden states, aligning them to reconstruct comprehensive, contextually-informed sequences. Equation (4) ultimately expresses the combined contextual sequence as the final hidden state ht, which is then passed to the capsule network layer.

1) CAPSULE LAYER

Furthermore, the precision of hate speech detection is ameliorated by using the capsule layer. Traditional CNNs lack distinct feature extraction and lose essential information due to pooling methods. Capsule networks, introduced by Hinton et al. [25], address this by capturing linguistically supplemented features that consider word order and local context. They excel in tasks like text classification and retrieval, outperforming CNNs. Capsules comprise smaller capsules and use vectors to convey classification likelihood and aspect orientation, resulting in more efficient representation. Unlike CNNs, capsules generate vectors instead of scalars, and the dynamic routing algorithm adjusts weights, enhancing feature extraction.

The Capsule Network (CapsNet) is a neural network architecture introduced to address the limitations of traditional Convolutional Neural Networks (CNNs) in handling hierarchical relationships and spatial hierarchies in data. Capsules are a fundamental building block in CapsNets, and they aim to represent specific patterns or entities in a more robust manner. The key equations for Capsules and routing-by-agreement are as follows:

Capsule Definition: A capsule C_i in layer l is defined as a vector s_i representing the instantiation parameters of an entity, such as the pose (position, orientation, etc.). In order to guarantee that the length of s_i is from 0 to 1, it also includes a squash function connected to it.

$$s_i = W_i \cdot x_i \tag{9}$$

$$v_i = squash(s_i) \tag{10}$$

Routing-by-Agreement (Dynamic Routing): Information is sent from lower-level capsules to higher-level capsules via dynamic routing to guarantee agreement among each capsule on the presence of entities.

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \text{ (Softmax to compute routing weights)} \tag{11}$$

$$s_j = \sum_i c_{ij} \cdot \mathrm{u}_{j|i} \text{ (Weighted sum of predictions)} \tag{12}$$

$$v_j = \mathrm{squash}(s_j) \text{ (Squash function)} \tag{13}$$

Here

C_i : represents a capsule in the lower layer,
v_i : is the output of that capsule,
W_i : represents the weight matrix,
x_i : is the input to the capsule,
c_{ij} : are the routing weights,
b_{ij} : are the log prior probabilities,
$\mathrm{u}_{j|i}$: is the prediction vector from capsule i to capsule j,
v_j : is the output of the higher-level capsule.

CapsNets are designed to capture hierarchical relationships by routing information between capsules and have shown promise in various tasks, especially in computer vision.

Our model utilizes capsules to improve hate speech classification. As a result, the final hidden state, denoting the output from the BiGRU layer, is transmitted to the capsule network layer. Dynamic routing is used for building coupling coefficients $c_i j$, which enables the model to ignore trivial hate-related terms in the input text. The importance of hate speech-related characteristics is correlated with the coupling coefficient $c_i j$; larger values denote more significant features, which capture crucial semantic representations of hateful speech in several directions.The coupling coefficient c_{ij} is computed using the Softmax function, and the output of the capsule, s_j, is obtained by adding up all of the prediction vectors. GRU-Caps algorithm is presented in Algorithm 2.

The extensive experiments reveal that the capability of detecting hate speech by accommodating linguistic variations and context, as discussed above, can be revamped by using the proposed combination of CNN, BiGRU, and the capsule layer.

2) DENSE AND OUTPUT LAYERS

The fully connected layer receives the output vector, vj, from the capsule network. Finally, using the sigmoid function on the dense layer output, the output layer classifies hateful speech. The proposed MLHS-CGCapNet model utilizes a binary cross-entropy loss function.

V. EXPERIMENTAL SETUP AND RESULTS

The suggested model's experimental details are presented in this section. The descriptions of the datasets, the experimental conditions, the hyperparameter settings, the evaluation

Algorithm 1 Tweet Preprocessing and Embedding Algorithm

```
Require: $T$ - Set of Tweets, $L$ - Labels, $lm$ - Maximum number of words in a tweet
Ensure: $E$ - Embedding matrix
for $t$ in $T$ do
    $Tt \leftarrow$ FilterTwitterMarkers($t$)
    $Ti \leftarrow$ FilterIrrelevant($Tt$)
    $Td \leftarrow$ FilterDuplicate($Ti$)
    $Tc \leftarrow$ ConvertCase($Td$)
end for
Encoding:
$tokenizer \leftarrow$ Tokenizer()
$tokenizer.fit\_on\_texts(Tc)$
$Tcs \leftarrow$ tokenizer.texts_to_sequences($Tc$)
$word\_index \leftarrow$ tokenizer.word_index
$word\_count \leftarrow$ len($word\_index$)
$label\_encoder \leftarrow$ LabelEncoder()
$encoded\_labels \leftarrow$ label_encoder.fit_transform($L$)
Padding:
$T\_padded \leftarrow$ pad_sequences($Tcs$, $lm$)
Glove Embedding Matrix:
$glove\_embedder \leftarrow$ GloVe()
for word, index in word_index.items() do
    $word\_embedding \leftarrow$ glove_embedder.get($word$)
    $E[index] \leftarrow word\_embedding$
end for
```

Algorithm 2 GRU-Caps Algorithm

```
Require: $E$ - Embedding-matrix, $W$ - Word Vectors
Ensure: Classified-tweets
Construct the Model:
$model \leftarrow$ Sequential()
$model$.add(Embedding(wc,embed,tweetlength, E))
$model$.add(Conv1D(128, 3, activation = 'relu'))
$model$.add(Bidirectional GRU(128))
$model$.add(Dropout(0.4))
$model$.add(Capsule(3, 5, 4))
$model$.add(Flatten())
$model$.add(Dense(1, activation = 'sigmoid'))
Compile the Model:
$model$.compile('binary_crossentropy', 'Adam', 'accuracy')
Train and Evaluate:
$history \leftarrow model$.fit($padded\_tweets$, $batch\_size$ = 128, $epochs$ = 50, $validation\_split$ = 0.20)
```

metrics, the evaluation results, and the comparative analyses are all presented.

A. EXPERIMENTAL SETTINGS

The presented model is developed using the Python programming language. All the models were trained on a high-performance computing cluster with NVIDIA Tesla V100 GPUs. For the purpose of crawling tweets from Twitter, we employed Tweepy, a built-in library. The implementation of the proposed model was carried out using Keras, Tensorflow, a widely used neural network library.

TABLE 2. Hyperparameters and their values in the proposed model.

Hyperparameter	Values
Padding	25
Glove Embedding Dimension	300
Filter Size (CNN-layer)	2, 3, 4
Number of CNN Filters	1
Number of neurons (BiGRU-layers)	128
Dropout	0.4
Routing	3
Numbers of Capsule	3
Dimension of Capsule	100
Optimizer	Adam

B. HYPERPARAMTER SETTINGS

The crucial hyper-parameters used to train the proposed model are dropout, batch size, verbose, epochs, and optimizer in order, 0, 4, 128, 2, and 50, and Adam respectively. The learning rate is set to 1×10^{-5}. Filter width, filter count, and pool size are set to 3, 128, and 2, respectively, in the CNN layer. To process information, the BiGRU layer makes use of 128 neurons. With the capsule network layer, there are 3, 3, and 100 capsules, respectively, along with the corresponding numbers of route iterations and capsules. Hyperparameters are listed in Table 2.

C. COMPARISON OF OUR MODEL WITH THE BASELINES

The authors compared the proposed model with the state-of-the-art architectures from recent papers, which observed better precision, recall and F1-score on the benchmark dataset of 9 languages for hate speech detection. In Table 3, In the domain of identifying hate speech, the effectiveness of our approach is contrasted with HateDetectNet, DeepHateModel, and HateBERT. The models were tested on both balanced as well as unbalanced sets of data after receiving training on unbalanced datasets. On the imbalanced dataset, our model's precision, recall, and score for F1 were, respectively, 0.85, 0.78, and 0.82; on the balanced dataset, they were, respectively, 0.78, 0.75, and 0.80. Similar performance metrics are presented for HateDetectNet, DeepHateModel, and HateBERT.

In the proposed study, the non-hate speech class is considered the positive class. As already discussed in Table 1, the samples of the positive class are higher than the samples of the negative class. Therefore, the precision, recall, and F1 score of the implemented models are higher when experimenting on an imbalanced dataset as compared to balanced datasets. The models are heavily biased towards

TABLE 3. Comparison of model performance on balanced and unbalanced datasets.

Model	Dataset	Precision	Recall	F1 Score
MLHS-CGCapNet	Unbalanced Balanced	**0.85** **0.78**	**0.78** **0.75**	**0.82** **0.80**
HateDetectNet [26]	Unbalanced Balanced	0.72 0.70	0.80 0.78	0.76 0.74
DeepHateModel [27]	Unbalanced Balanced	0.68 0.66	0.82 0.80	0.74 0.72
HateBERT [28]	Unbalanced Balanced	0.76 0.74	0.70 0.68	0.73 0.71
mBERT with Muse [14]	Unbalanced Balanced	0.70 0.80	0.75 0.75	0.765 0.70

TABLE 4. Comparison of model accuracy on balanced and unbalanced datasets.

Model	Balanced Data	Unbalanced Data
MLHS-CGCapNet	**87%**	**90.71%**
HateDetectNet [26]	74%	76%
DeepHateModel [27]	72%	74%
HateBERT [28]	71%	73%
mBERT with Muse [14]	80%	76.5%
HateDetect [14]	82%	85%
CrossLing [29]	66.7%	62%

the majority class. The above fact can also be confirmed in Table 3.

In Figure 8, we present a comparison of model performance on a balanced dataset. The figure displays the F1 Score, recall, and precision for five distinct models: MLHS-CGCapNet, HateBERT, DeepHate-Model, HateBERT, and mBERT with Muse. Each curve represents one of these models, with different colors indicating the model identity. Model names are listed on the x-axis, while the related metric scores are shown on the y-axis. The continuous curves for precision, recall, and F1 Score show how each model's performance varies across these metrics. As each model is trained on balanced data, this visualization offers insightful information about its advantages and disadvantages. Figure 9 displays a similar comparison of model performance, but this time on an unbalanced dataset. Like the previous plot, it includes precision, recall, and F1 Score curves for the same five models. By comparing this plot to the one for balanced data, we gain insights into how these models adapt to imbalanced datasets and whether their relative strengths and weaknesses change. This analysis is crucial for selecting the most suitable model for a given dataset scenario. The results demonstrate that our model performs better than the competition in terms of accuracy, recall, and F-1 scores. On the data set that is unbalanced, HateDetectNet displays a greater recall, but DeepHateModel shows a strong recall at the price of precision. HateBERT shows balanced trade-offs between precision and recall on both dataset distributions. The comparison sheds light on how these algorithms perform in various hate speech detection circumstances.In a groundbreaking endeavor, we expanded the capabilities of the mBERT transformer model beyond its traditional training in English to encompass a diverse range of languages. By meticulously training the mBERT transformer on our curated dataset of 12 distinct languages, we transcended the limitations of monolingual training and pioneered a truly multilingual approach. This transformative adaptation empowers our model to harness its contextual understanding and linguistic prowess across various languages, enabling it to effectively detect hate speech nuances, cultural idiosyncrasies, and linguistic intricacies. Our extension of the mBERT transformer's training paradigm marks a pivotal step towards creating a globally relevant and culturally sensitive solution for hate speech detection in an increasingly interconnected digital landscape.

A detailed breakdown of model accuracies in the context of hate speech detection, with an emphasis on both balanced and unbalanced datasets, is provided in Table 4. With an accuracy of 87 on the balanced dataset and 90.71 on the unbalanced dataset, the benchmark model, MLHS-CGCapNet, performs admirably, demonstrating its robustness against different data distributions. In comparison, HateDetectNet [26] yields accuracies of 74 on the balanced dataset and 76 on the unbalanced dataset, while DeepHateModel [27] and HateBERT [28] achieve accuracies of 72 and 71, and 74 and 73, respectively. Additionally, HateDetect [14] and CrossLing [29] exhibit accuracies of 82 and 85, and 66.7 and 62, respectively, highlighting their varying degrees of performance in hate speech detection tasks. Collectively, these findings provide insightful information for the field of hate speech identification by highlighting the advantages and disadvantages of various methods in addressing data imbalances.

D. EVALUATION METRICS

The evaluation metrics, such as the F1-score, precision, and recall of the proposed hate speech detection model, are determined from the mapping between predicted and actual classes using the output of the capsule layer. The improved evaluation metrics reveal an efficient tokenization process, a powerful and enriched contextual extractor, and a remarkable pattern tracker for which the input and embedding layers, capsule layer, and BiGRU layer are used, respectively. Therefore, the efficacy of the proposed model can easily be reflected by the above-discussed evaluation metrics, as the evaluation metrics also provide an image of the efficiency of the captured features in a quantitative manner.

E. PERFORMANCE COMPARISON

The main challenge faced during experimentations is achieving the generalization of the proposed model so that the hate detection algorithm precisely detects hate speech in multilingual datasets. Moreover, processing text in multiple languages involves significant computational power and storage, especially for deep learning models. The assessment

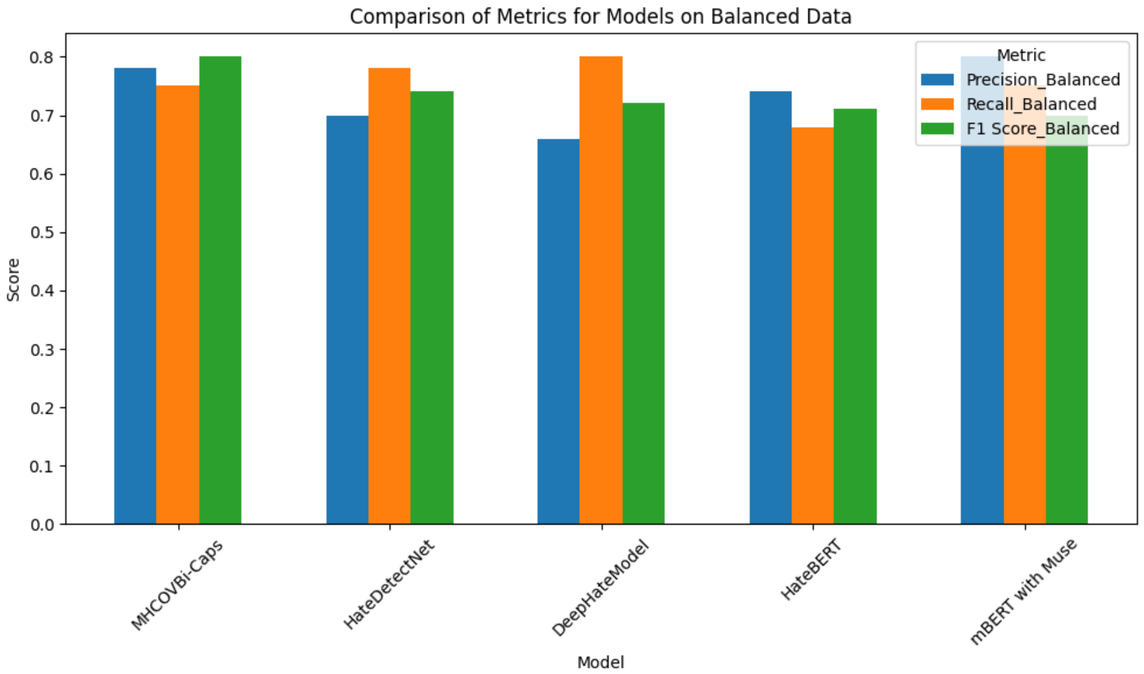

FIGURE 8. Comparison of various approaches and our suggested model on balanced data.

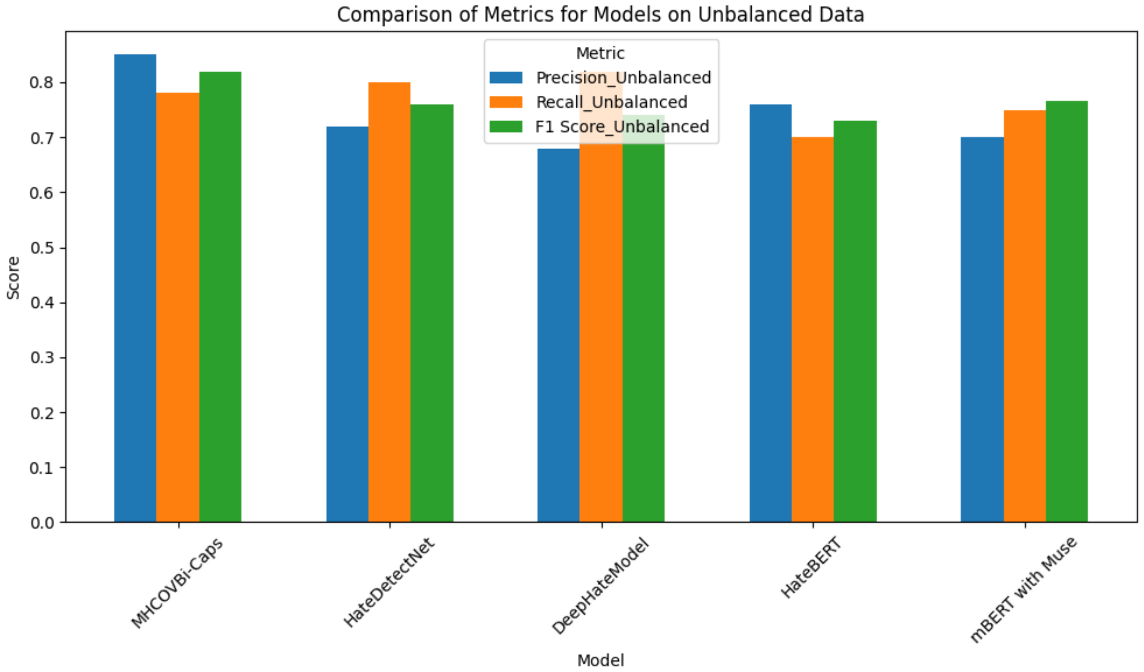

FIGURE 9. Comparison of various approaches and our suggested model on unbalanced data.

findings for the suggested approach for detecting hate speech across the 12 datasets are presented in this section. Additionally, we contrast the suggested model with two state-of-the-artÂ deep learning-based methods and six standard approaches. It's interesting to observe that the suggested approach performs better for the data set with imbalances than the balanced dataset.

Precision is a statistic that measures how well a classifier predicts the future in the affirmative. It computes the proportion of accurate positive predictions to all the positive predictions made by the classifier. In other words, precision reveals the proportion of correctly anticipated positive things. Recall, sometimes referred to by the term sensitivity or the actual positive rate, measures the ratio of accurate positive

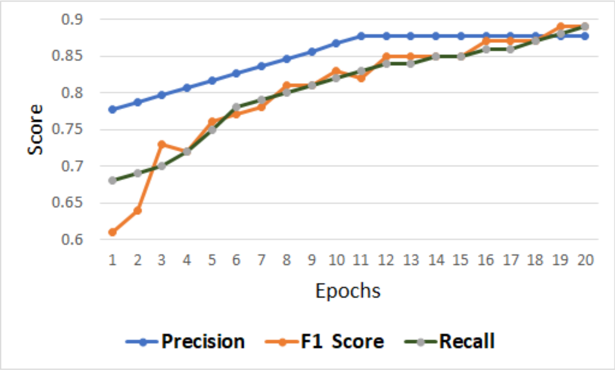

FIGURE 10. Precision, recall, and F1 score curves.

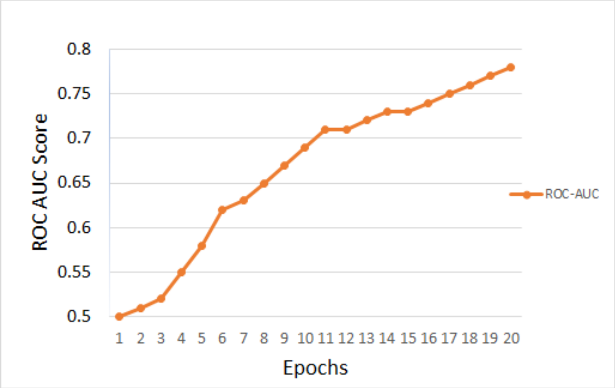

FIGURE 11. Validation ROC AUC scores.

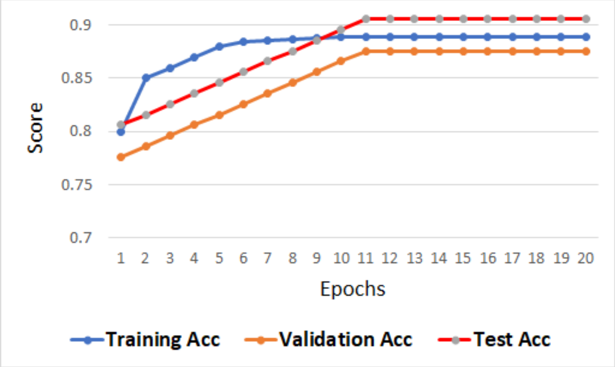

FIGURE 12. Accuracy trends across epochs for training, validation, and test sets.

predictions to all the actual positive cases that exist in the dataset. Recall reveals how effectively the classifier can identify all the favorable cases. Precision and recall are balanced by the harmonic average of the F1 Score, which provides this balance. When there is an imbalance between the classes in the dataset, it is very helpful. The F1 score offers a single number that combines accuracy and recall while taking into account both positives that are false and false negatives. Figure 10 displays Precision, Recall, and F1 score curves.

In Figure 11, We provide the ROC AUC scores for the validation set across epochs. A popular statistic for assessing the effectiveness of binary classification models is the ROC AUC score. What is being assessed is the area under the Receiver Operating Characteristic (ROC) curve, which compares the True Positive Rate (Sensitivity) with the False Positive Rate (1-Specificity) for different threshold values. ROC and (AUC) area under the curve values of 0.5 and 1 respectively indicate random guessing, while the former indicates flawless categorization. From the plot, we observe that the ROC AUC score gradually increases over the epochs. This shows that the model becomes better at differentiating between positive and negative classes as training goes on. The upward trend signifies that the model's predictions are becoming more reliable, yielding better separability between the classes.

In the presented accuracy analysis, Figure 12 illustrates the progression of accuracy scores across epochs for the training, validation, and test datasets. The training accuracy curve initially starts at around 0.80 and gradually increases, stabilizing near 0.89. The validation accuracy curve follows a similar pattern, starting at 0.77 and plateauing around 0.88. The test accuracy curve, starting at approximately 0.80, also levels off at around 0.90. The convergence of the training and validation curves indicates effective learning without overfitting, while the stability of the test curve suggests consistent generalization performance. These trends collectively reflect the model's learning progression and its capacity to generalize accurately to new data, highlighting the point of diminishing returns with continued training.

The computational cost of the suggested model heavily relies on the quantity of the datasets, the languages enriched in vocabulary, trainable parameters and complexity of the architecture. For instance, the suggested model takes 31 minutes to train for the proposed multilingual hate speech detection classifier on a same machine used for the experimentations. However, the state-of-the-art architecture i.e. mBERT with Muse spends 42 minutes to train. Moreover, the recommended model consumes only 0.45 million parameters. The number of parameters consumed by the proposed model is relatively smaller than the parameters consumed by the mBERT with Muse, which uses 110 million parameters to train.

VI. DISCUSSION

In the presented research work, the authors concluded that a diverse and complex hate speech dataset illustrates significant improvement in the performance of the hate speech detection algorithm. The variability between the multiple languages and volatile cultural contexts creates rich contextual training features with strong patterns, which maps the textual data to the contextual features to improve the generalization capability across different languages. This can be particularly beneficial for languages with limited training data.

The proposed study broadens the scope of toxic speech classification by introducing 12 different languages, which enhance the semantic features and patterns. Eventually, the practicability of the proposed model can be extended globally for various cultural languages for the real-world toxic speech detection with reliability and consistent performance. Moreover, the dataset exhibits a wide range of HS categories, such as general hate, abusive and Cyberbullying, Radicalization, religion, racial, and sexism. Therefore, the proposed trained model can be deployed in multi-faceted real-world scenario with small hyper-parameter tuning. The recent work did not take benefit from the additional information extracted from the multilinguistic diversity classifier.

The constructed dataset has both short and long contexts. The extensive experiments delineate that the deployment of the capsule layer emanates substantial improvement in model performance by using both the long and short context. This might coincide with the stochastic nature of human behavior —that some people comment based solely on headlines, while others offer diverse responses after thoroughly reading the content.

The proposed model is trained on a diversified dataset; however, the proposed model also has limitations. The problem is considered as supervised learning where human annotators labeled the data as a binary classification. The multi-class classification can also be deployed on the under-considered dataset to assess the severity of the hated comments. Secondly, more comments based on the toxicity of the speech can be accommodated to develop a more practical and robust algorithm.

VII. CONCLUSION AND FUTURE WORK

This research has developed a lightweight deep neural network model that combines the convolutional, BiGRU, and capsule network models in order to recognize utterances of hatred. The proposed framework achieved the better hate speech detection accuracy using the following design concepts of deep learning:

1) The suggested approach employs a capsule network to combine contextual data from different orientations.
2) The CNN layer was used to capture the contextual and semantic nuances.
3) The BiLSTM hidden layer was used in the proposed BiLSTM model to extract useful patterns from the contextual features.
4) The proposed model was tested against 12 benchmark datasets from Twitter that were balanced and unbalanced in order to differentiate statements of hatred from other types of text. When accuracy, recall, and f-score are all taken into account, the suggested model scores best on the entire (unbalanced) dataset, with scores of 0.89, 0.80, and 0.84, respectively.
5) The recommended model performs best when taking into account the imbalanced dataset, reaching values of 0.93 and 0.90, respectively, for training and validation accuracy. The proposed model significantly improves performance compared to baseline and state-of-the-art approaches.
6) Combinations of different hyperparameters were used and observed the effect of different hyperparameters thoroughly on both neural and capsule networks. This investigation aimed to evaluate how well our suggested method performed.

The authors are keen to extend the presented work by incorporating new techniques in the future, which are as follows:

1) Dataset Enhancement: The authors plan to evaluate the performance of the proposed model on additional datasets sourced from social media sites like Facebook in the future. Moreover, the authors will also work on other benchmark datasets, such as Hatebase Twitter etc., to make the model more robust and reliable.
2) Data Augmentation: The authors aim to create synthetic data for the minority classes for the hate speech detection system using the Synthetic Minority Oversampling Technique (SMOTE) and random oversampling and see the impact of the above-mentioned techniques on the performance of the hate speech detection model.
3) Model Improvement: The presented model attains 87% accuracy on the balanced dataset and 90.71% accuracy on the imbalanced dataset. Hence, there is a gap in improving the model accuracy on a balanced dataset. The authors intend to improve the model architecture so that a new modified model achieves accuracy up to 90%.

DECLARATIONS

COMPETING INTEREST

The authors declare that they have no competing interest to report regarding the present study.

FUNDING

The authors received no specific funding for this study.

REFERENCES

[1] T. Wullach, A. Adler, and E. Minkov, "Towards hate speech detection at large via deep generative modeling," *IEEE Internet Comput.*, vol. 25, no. 2, pp. 48–57, Mar. 2021.

[2] M. Ptaszynski, A. Pieciukiewicz, and P. Dybała, "Results of the PolEval 2019 shared task 6: First dataset and open shared task for automatic cyberbullying detection in polish Twitter," in *Proc. PolEval 2019 Workshop. Inst. Comput. Sci., Polish Acad. Sci.*, 2019, pp. 89–110.

[3] M. R. Awal, R. K-W. Lee, E. Tanwar, T. Garg, and T. Chakraborty, "Model-agnostic meta-learning for multilingual hate speech detection," *IEEE Trans. Computat. Social Syst.*, vol. 11, no. 1, pp. 1086–1095, 2023.

[4] A. Jiang and A. Zubiaga, "Cross-lingual capsule network for hate speech detection in social media," in *Proc. 32st ACM Conf. Hypertext Social Media*, Aug. 2021, pp. 217–223.

[5] V. Basile, C. Bosco, E. Fersini, D. Nozza, V. Patti, F. M. R. Pardo, P. Rosso, and M. Sanguinetti, "SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter," in *Proc. 13th Int. Workshop Semantic Eval.*, 2019, pp. 54–63.

[6] M. Zampieri, P. Nakov, S. Rosenthal, P. Atanasova, G. Karadzhov, H. Mubarak, L. Derczynski, Z. Pitenis, and Ç. Çöltekin, "SemEval-2020 task 12: Multilingual offensive language identification in social media (OffensEval 2020)," 2020, *arXiv:2006.07235*.

[7] P. K. Roy, A. K. Tripathy, T. K. Das, and X.-Z. Gao, "A framework for hate speech detection using deep convolutional neural network," *IEEE Access*, vol. 8, pp. 204951–204962, 2020.

[8] S. T. Luu, H. P. Nguyen, K. Van Nguyen, and N. Luu-Thuy Nguyen, "Comparison between traditional machine learning models and neural network models for Vietnamese hate speech detection," in *Proc. RIVF Int. Conf. Comput. Commun. Technol. (RIVF)*, Oct. 2020, pp. 1–6.

[9] Y. Lee, S. Yoon, and K. Jung, "Comparative studies of detecting abusive language on Twitter," 2018, *arXiv:1808.10245*.

[10] B. Evkoski, A. Pelicon, I. Mozetič, N. Ljubešić, and P. K. Novak, "Retweet communities reveal the main sources of hate speech," *PLoS ONE*, vol. 17, no. 3, Mar. 2022, Art. no. e0265602.

[11] H. T.-T. Do, H. D. Huynh, K. Van Nguyen, N. L.-T. Nguyen, and A. G.-T. Nguyen, "Hate speech detection on Vietnamese social media text using the bidirectional-LSTM model," 2019, *arXiv:1911.03648*.

[12] J. Melton, A. Bagavathi, and S. Krishnan, "Del-hate: A deep learning tunable ensemble for hate speech detection," in *Proc. 19th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2020, pp. 1015–1022.

[13] T. Wullach, A. Adler, and E. Minkov, "Character-level HyperNetworks for hate speech detection," *Expert Syst. Appl.*, vol. 205, Nov. 2022, Art. no. 117571.

[14] S. S. Aluru, B. Mathew, P. Saha, and A. Mukherjee, "Deep learning models for multilingual hate speech detection," 2020, *arXiv:2004.06465*.

[15] S. Zimmerman, U. Kruschwitz, and C. Fox, "Improving hate speech detection with deep learning ensembles," in *Proc. 11th Int. Conf. Lang. Resour. Eval. (LREC)*, 2018.

[16] S. Kamble and A. Joshi, "Hate speech detection from code-mixed Hindi-English tweets using deep learning models," 2018, *arXiv:1811.05145*.

[17] R. Velioglu and J. Rose, "Detecting hate speech in memes using multimodal deep learning approaches: Prize-winning solution to hateful memes challenge," 2020, *arXiv:2012.12975*.

[18] K.-L. Chiu, A. Collins, and R. Alexander, "Detecting hate speech with GPT-3," 2021, *arXiv:2103.12407*.

[19] L. Li, L. Fan, S. Atreja, and L. Hemphill, "'HOT' ChatGPT: The promise of ChatGPT in detecting and discriminating hateful, offensive, and toxic comments on social media," *ACM Trans. Web*, vol. 18, no. 2, pp. 1–36, 2023.

[20] X. He, S. Zannettou, Y. Shen, and Y. Zhang, "You only prompt once: On the capabilities of prompt learning on large language models to tackle toxic content," 2023, *arXiv:2308.05596*.

[21] Q. Zhong, L. Ding, J. Liu, B. Du, and D. Tao, "Can ChatGPT understand too? A comparative study on ChatGPT and fine-tuned BERT," 2023, *arXiv:2302.10198*.

[22] *Models*. Accessed: Jun. 26, 2024. [Online]. Available: https://platform.openai.com/docs/models/gpt-3-5

[23] Hate Alert. (2022). *DE-LIMIT: A Dataset for Low-Resource Languages for Hate Speech Detection*. [Online]. Available: https://github.com/hate-alert/DE-LIMIT

[24] B. Ross, M. Rist, G. Carbonell, B. Cabrera, N. Kurowsky, and M. Wojatzki, "Measuring the reliability of hate speech annotations: The case of the European refugee crisis," 2017, *arXiv:1701.08118*.

[25] S. Sabour, N. Frosst, and G. Hinton, "Matrix capsules with EM routing," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–15.

[26] P. Kapil and A. Ekbal, "A deep neural network based multi-task learning approach to hate speech detection," *Knowl.-Based Syst.*, vol. 210, Dec. 2020, Art. no. 106458.

[27] Z. Zhang, D. Robinson, and J. Tepper, "Detecting hate speech on Twitter using a convolution-GRU based deep neural network," in *Proc. 15th Int. Conf. Semantic Web (ESWC)*, Crete, Greece. Heidelberg, Germany: Springer, Jun. 2018, pp. 745–760.

[28] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A BERT-based transfer learning approach for hate speech detection in online social media," in *Proc. 8th Int. Conf. Complex Netw. Their Appl.*, vol. 1. Stroudsburg, PA, USA: Springer, 2019, pp. 928–940.

[29] D. Nozza, "Exposing the limits of zero-shot cross-lingual hate speech detection," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 907–914.

ABIDA KOUSAR received the B.S. degree in software engineering from COMSATS University Islamabad, Pakistan. She is currently pursuing the M.S. degree in data science with the School of Systems and Technology, University of Management and Technology, Lahore, Pakistan. She is a Software Engineer. Her research interests include machine learning, computer vision, deep learning, pattern recognition, and cybercrimes on social media.

JAMEEL AHMAD received the M.Sc. degree in electrical engineering from the University of Southern California, Los Angeles, USA, and the Ph.D. degree in electrical engineering from UET Lahore. He was with Qualcomm and Broadcom, San Diego, CA, USA, on 3G mobile communication systems. Currently, he is an Assistant Professor with the University of Management and Technology, Lahore, Pakistan. His research interests include parallel and distributed computing, deep learning, the cybersecurity of autonomous vehicles, and the optimization of smart microgrids for energy internet. He is the reviewer of numerous journals and conferences.

KHALID IJAZ received the M.Sc. degree in electrical engineering from UMT Lahore. He is currently pursuing the Ph.D. degree with COMSATS University Islamabad, Lahore Campus, Pakistan. Since August 2018, he has been a Lecturer with the Electrical Engineering Department, School of Engineering, UMT Lahore. He was a Lab Engineer with UMT, from April 2013 to July 2018. He was associated with the Telecom industries and worked in PTCL and WorldCall, Pakistan, for five years. He has in-depth knowledge and hands-on experience in mobile switching centre (MSC) and CDMA technology. His research interests include machine learning, deep learning, and reinforcement learning applied to time-series and imaging data.

AMR YOUSEF (Member, IEEE) received the B.Sc. degree from the Electrical Engineering Department, Alexandria University, in 2001, the M.Sc. degree from the Engineering Mathematics Department, Alexandria University, in 2006, and the Ph.D. degree in electrical and computer engineering from Old Dominion University (ODU), in May 2012. He was a Postdoctoral Research Associate with the Old Dominion Vision Laboratory, USA. He is currently an Assistant Professor with the Electrical Engineering Department, University of Business and Technology, Saudi Arabia, and the Engineering Mathematics Department, Alexandria University, Egypt. His research interests include optimization approaches, image processing, computer vision, and machine learning algorithms. He is a member of SPIE and OSA.

ZAFFAR AHMED SHAIKH (Member, IEEE) was born in Khairpur, Sindh, Pakistan, in 1977. He received the Ph.D. degree from the Institute of Business Administration, Karachi, Pakistan, in 2017, under the supervision of Prof. Shakil Ahmed Khoja. The title of his dissertation was Guided Personal Learning Environment Model: Concept, Theory, and Practice. He is currently an Associate Professor of computer science and information technology with Benazir Bhutto Shaheed University Lyari, Karachi. His teaching portfolio includes a range of subjects, such as assembly language, business intelligence and analytics, compiler construction, computer architecture, digital logic design, human–computer interaction, semantic analysis, semantic web, statistical inference, and automata theory. He is an Alumnus of the REACT Research Group, École Polytechnique Fédérale de Lausanne, Switzerland, where he spent six months as a Visiting Doctoral Fellow due to his contributions to personalized recommender systems, technology-enhanced learning, and semantic analysis-based personalized recommender systems, in 2014. His academic career spans more than 24 years, during which he received many prestigious scholarships and travel grants from national and international organizations, including the M.S. leading to the Ph.D.

scholarship, for five years, and the International Research Support Initiative Program (IRSIP) Scholarship from HEC Pakistan, for six months; and several international travel grants for presenting research: four grants from HEC Pakistan, two grants from IBA-Karachi, and two grants from the Ministry of Education, Saudi Arabia, for the third and fourth eLi conferences. He has published more than 75 peer-reviewed articles in high-ranked journals, many of which are indexed in SSCI, SCIE, and Scopus. His current research interests include artificial intelligence, blockchain, business intelligence, cybersecurity, educational technologies, energy economics, expert systems, fault detection and diagnosis, green computing, healthcare systems, the Internet of Things, large language models, learning environments, machine learning methods, medical image processing, metaverse, pharmacy informatics, recommender systems, and fintech. He has presented his work at leading international conferences, such as ACM SIGITE, IEEE iCALT, IEEE IANA, and the PLE Conference. He is a Senior Editorial Board Member and a Reviewer of many prestigious journals, such as *Australasian Journal of Educational Technology*, *British Journal of Educational Technology*, *Behavior and Information Technology*, *BMJ Open*, *Complexity*, *Computers in Human Behavior*, *Computers and Education*, *Cogent Education*, *Cybernetics and Systems*, *Human-centric Computing and Information Sciences*, IEEE ACCESS, IEEE SENSORS JOURNAL, *Interactive Learning Environments*, *Multimedia Tools and Applications*, *PLOS One*, *System*, *Wireless Communications and Mobile Computing*, and many MDPI journals.

IKRAMULLAH KHOSA (Senior Member, IEEE) received the Ph.D. degree in electronics and telecommunications from Politecnico di Torino, Italy, in 2015. He is currently an Assistant Professor with the Electrical and Computer Engineering Department, COMSATS University Islamabad, Lahore Campus. His research interests include artificial intelligence, data analysis, machine learning, and pattern recognition.

DURGA CHAVALI (Senior Member, IEEE) is currently a Distinguished Researcher and a Leader in the field of technology and healthcare, has a robust background in cloud computing, AI technologies, IoMT, deep learning, blockchain, machine learning, and big data analytics. As an accomplished author, she has contributed significantly to the literature, having edited and authored best-selling books on behavioral health and AI remedies and AI virtual assistance for care management. With a strong presence in research publications, she has showcased expertise by publishing work on machine learning techniques with wide-reaching impact. Notably, her contributions to ResearchGate have engaged millions of researchers, and her participation in judging research papers at Springer journals, ACM journals, and international seminars and conferences attest to her commitment to advancing knowledge. In research laboratory management, she has demonstrated leadership by overseeing a dynamic team of Ph.D. graduates, engineers, and business executives focusing on cutting-edge technologies, such as federated learning (FL) and blockchain. In professional engagements, she serves on prestigious committees and boards, such as the SIG Advisory Board in HDAA; and the Vice Chair for IEEE EMBS South East Michigan. As a Full Member of the Sigma XI, she serves as an editorial board reviewer for the Springer, IGI-Global, and ACM Scopus-indexed AI journals. Notably, at Trinity Health, she played a pivotal role in leading a team and implementing data and AI/ML strategies and contributing significantly to the development of Trinity Health's medical economics claims analytics system and claims system for Medigold and Medicare Health Care. Recognized for impactful research during the COVID era, she has received nominations for the Best Innovator Award and the Women in Engineering Award, showcasing an ongoing commitment to excellence. With numerous ongoing activities, including pending publications, awards, and fellowship considerations, she continues to make profound contributions to the field, leaving an indelible mark on the intersection of technology and healthcare.

MOHD ANJUM received the Master of Technology degree in computer science and engineering (software engineering) and the Ph.D. degree in computer engineering from Aligarh Muslim University, India. He was an Assistant Professor with Aligarh Muslim University, from 2012 to 2015. His current research interests include waste management, the Internet of Things, and machine learning. He has published and presented numerous research papers in reputed journals and international conferences in his area of interest.