Iraqi Journal for Computer Science and Mathematics

Journal Homepage: http://journal.esj.edu.iq/index.php/IJCM
e-ISSN: 2788-7421 p-ISSN: 2958-0544

Addressing Challenges in Hate Speech Detection Using BERT-Based Models: A Review

Jinan Ali Aljawazeri[1]*, Mahdi Nsaif Jasim[2]

[1] Department of Software, University of Babylon, Hillah, 51001, IRAQ
[2] Department of Business Informatics, University of Information Technology and Communications, Baghdad, 10001, IRAQ

*Corresponding Author: Jinan Ali Aljawazeri

DOI: https://doi.org/10.52866/ijcsm.2024.05.02.001
Received October 2023; Accepted January 2024; Available online March 2024

ABSTRACT: The rapid growth of social media platforms has led to an increase in hate speech. This has prompted the development of effective detection mechanisms that aim to mitigate the potential hazards and threats it poses to society. BERT (Bidirectional Encoder Representations from Transformers) has produced cutting-edge results in this field. This review paper aims to identify and analyze the whole process of using the BERT model to tackle the challenges associated with the hate speech detection problem. This academic discussion will begin by addressing the training datasets and the preprocessing methods involved. Subsequently, the use of the BERT model will be explored, followed by an examination of the contributions made to address the issues encountered. Finally, we will discuss the evaluation phase. The use of BERT included the application of two primary approaches. In the feature-based approach, BERT accepts textual input and generates its corresponding representation as output. The resulting output is then used as input for any classification model. The second approach involves the process of fine-tuning BERT using labeled datasets and then employing it directly for classification purposes. The controversial issues and open challenges that appeared at each stage were discussed. The results indicate that in both approaches, BERT has shown its efficacy relative to other models under contention. However, there is a need for greater attention and advancement to effectively solve the existing issues and constraints in the future.

Keywords: Hate Speech Detection, BERT, Feature-Based, Fine-Tuning

1. INTRODUCTION

In our modern era, social media platforms play an important role in the lives of most people. Despite the many benefits of this trend, its disadvantages cannot be ignored. It facilitates the spread of hate speech, transcending all geographical borders [1]. According to this fact, the automatic detection of hate speech on social media platforms has become the subject of extensive research. Hate speech detection is considered and treated as a text classification task [1], [2]. The input usually consists of a string of words, and the output is the resulting class.

Many researchers have made efforts to contribute to solving the problem. Methods used are mainly classified into two popular groups: classical machine learning methods and deep learning methods. Classical machine learning applies classification algorithms such as Random Forests, Naïve Bayes, Logistic Regression, Decision Trees, and Support Vector Machines (SVM) [1], [3]. On the other hand, deep learning methods employ multiple layers of neural networks to automatically select useful features and uncover hidden patterns from the input raw data [4]. The most commonly used neural networks are convolutional neural networks (CNNs), recurrent neural networks (RNNs), long short-term memory networks (LSTMs), and bidirectional long short-term memory networks (Bi-LSTMs) [5]. Recent studies have shown that deep learning models have outperformed classical ones [6], [7]. One of the reasons is that classical machine learning techniques usually rely on manually crafted characteristics that fail to capture the conceptual connection between terms [8].

Transformers, a special type of deep learning model that relies completely on the attention mechanism, are achieving outstanding results in hate speech classification specifically and in natural language processing (NLP) in general [9], [10]. BERT (Bidirectional Encoder Representations from Transformers) is the most popular transformer model used in the task of hate speech detection [9]. It produces superior results and outperforms any other text representation model. Many researchers have highlighted the significance of this model as a crucial cutting-edge approach in the area [11]. BERT plays a vital role in both the textual representation and classification processes [12].

Despite the wide availability of research, hate speech is still problematic and challenging [1], [13], [14], [15]. Although many researchers have contributed to addressing these challenges and others have conducted many reviews

*Corresponding author: author@organization.edu.co on hate speech detection, the literature lacks a comprehensive review specifically focused on BERT-based models and the challenges they have tackled. This paper aims to bridge this gap by analyzing BERT-based models and advancing progress in the field of hate speech detection. It does so by consolidating and synthesizing related datasets, preprocessing steps, current methodologies, approaches used, contributions applied to address challenges, and the findings obtained. It aims not only to fill the current void in specialized reviews but also to highlight possible paths for future research that can strengthen their applicability and impact. The remaining sections introduce the following topics: an introduction to transformers and the BERT model, popular datasets for hate speech classification, preprocessing steps in previous work, the role of BERT-based models and their challenges, discussion, conclusion, and future work. These topics are covered in sections 2, 3, 4, and 5.

2. BACKGROUND

2.1 TRANSFORMERS

Transformers are a superior deep learning architecture proposed in 2017. Nowadays, transformers are commonly exploited for NLP problems. They are competing with many existing networks, such as RNNs and CNNs. They have an encoder-decoder architecture and depend completely on the attention mechanism instead of combining it with recurrent and convolutional layers. The attention mechanism used is called self-attention [10], [16].

The transformer consists of a set of encoders, followed by a set of decoders. The encoder receives the source sentence as input. Then, it learns its representation and sends the representation to the decoder. The decoder receives the representation learned by the encoder as input and generates the output sentence [16].

The encoder consists of two sub-layers: multi-head attention and feed-forward network, while the decoder has a similar architecture but with an extra sub-layer that consists of masked multi-head attention. The transformer network reads the words in parallel, unlike other networks that read the words in order like RNN, and so this order helps in the understanding of the meaning of the text. To solve this issue, transformers insert potential information about the token's position.

The transformer architecture is not constant and can be adopted according to the specific problem. Commonly, three different styles are used: the encoder-decoder, a set of encoders (for representation), or a set of decoders (for sequence generation) [17]. Several transformer variants, such as BERT, generative pre-training transformer (GPT), and T5, have been proposed over the last few years due to their great success [17].

2.2 BERT MODEL AND ITS VARIANTS

BERT is the most recent language representation model published by Google AI in 2019 [12]. It has yielded greater innovation in various NLP tasks, including question answering, text generation, sentence classification, and many others. BERT may be defined as a transformer model that consists of a set of encoders, as shown in Fig. 1. Each encoder employs a multi-head attention mechanism to fully understand the contextual significance of each word in a sentence. This mechanism establishes connections between every word in the sentence, facilitating the discovery of relationships and contextual meanings. Consequently, the sentence is sent as input to the encoders of BERT, which then generates the contextual representations for each word in the sentence as an output [16]. Some key factors behind the great success of BERT are:

a. It depends on the context when generating representations, unlike other traditional feature extraction methods that are considered context-free [16].

b. It is designed to pre-train deep bidirectional representations from unlabeled text by employing training techniques that read from both left and right directions in all layers, unlike other unidirectional models.

Prior to feeding data into BERT, it is necessary to transform the input into embeddings using the three specified embedding layers, as outlined in Fig. 2. In the token embeddings layer, the sentence is tokenized into individual tokens, and two special tokens, [CLS] and [SEP], are added. [CLS] is added at the beginning of the first sentence only, while [SEP] is added at the end of each sentence. Before inputting the tokens into BERT, it is essential to transform the tokens into embeddings using an embedding layer known as token embedding. It should be noted that the values of token embeddings will be acquired via the training process. Segment embedding is used to differentiate between the two provided sentences in certain tasks. Position embeddings add word order information to the transformer model, which operates without any form of recurrence and reads all words in parallel [12].

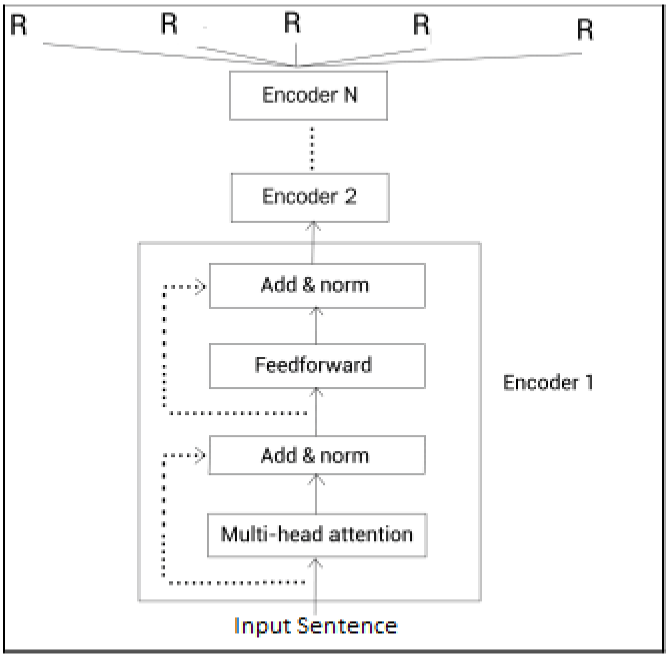

FIGURE 1. - BERT structure [16]

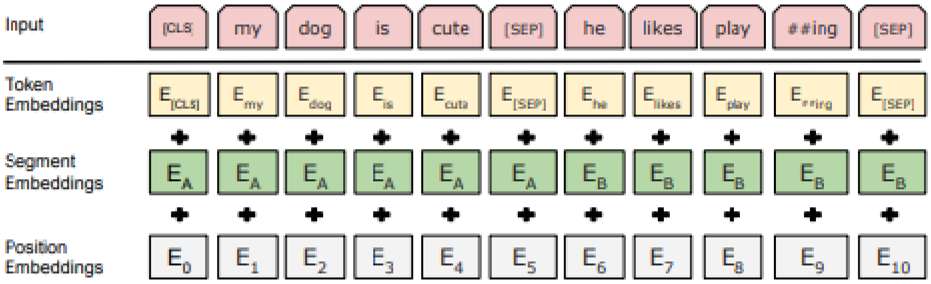

FIGURE 2. - BERT input representation [12]

Consequently, the pre-trained BERT model can be fine-tuned with a simple extra output layer to provide cutting-edge models for a variety of tasks [12], [16]. The BERT framework consists of two steps: pre-training and fine-tuning [5], [12].

a. BERT Pre-Training

Pre-training involves using large unlabeled text datasets to train a language model, which is why it is referred to as a pre-trained model. This pre-trained model can be highly beneficial for later tasks that may suffer from a limited number of labeled datasets. This technique, known as transfer learning, involves using a deep learning model that is previously trained for a specific task and applying it to another similar task [5], [18].

Instead of creating a model from scratch [18], [19], using a pre-trained model as a starting point can be a preferable option. Labeled data is not required for pre-training transformer-based models. The only requirement to train a transformer-based model is a substantial amount of unlabeled text data. Other NLP tasks, such as text categorization, named entity recognition, and text generation, can be performed using the learned model. Sometimes, overfitting can occur when training a transformer-based model from scratch on a small dataset. For this reason, it is preferable to employ a pre-trained BERT model that has been trained on a sizable dataset. The model can then be fine-tuned on our comparatively smaller dataset by performing additional training [18].

BERT is pre-trained using two techniques: the masked language model (MLM) and the next sentence prediction (NSP) method. MLM employs a straightforward approach of randomly masking a certain proportion of the input tokens and then predicting those masked tokens. It utilizes both left and right context, unlike other methods that only consider context in one direction. The NSP technique is designed to enhance BERT's ability to recognize the links between sentences by capturing knowledge of extended dependencies that span several sentences. Both techniques help BERT mitigate the limitations of unidirectionality [12]. To finish the pre-training process, Google has invested in expensive machinery and extensive corpora. BERT has been trained using Wikipedia and the Books Corpus (800M words), (2,500M words) on 4 Cloud Tensor Processing Units (TPUs) (16 TPU chips total) for four days [5].

b. BERT Fine-Tuning

Utilizing pre-trained language models and fine-tuning them for specific downstream NLP tasks is a well-known practice. The pre-trained parameters were originally used to initialize the BERT model for fine-tuning. Then, a labeled dataset from a downstream task is used to fine-tune each parameter [5]. The pre-trained BERT model can be improved to build the most cutting-edge model for different tasks by adding just one extra output layer and making minimal changes to the task-specific architecture.

Fine-tuning can be applied in different directions [20]. It can depend on whether to train all the layers of the pre-trained model or train some layers while freezing others. The types of layers added on top of the architecture also vary. It may include feed-forward layers, convolutional layers, LSTM layers, or others. The datasets used for training can come from a specific task or from several related tasks, which is called multi-task fine-tuning [21].

BERTbase and BERTlarge are two parameter-intensive settings [5]. BERTlarge has 24 layers, 16 attention heads, and 340 million parameters, whereas BERT-base has an encoder with 12 Transformer blocks, 12 self-attention heads, and 110 million parameters. BERT reads a maximum of 512 tokens and creates a 768-dimensional vector representation of a token sequence. Each of BERTbase and BERTlarge has two versions: uncased and cased. The uncased version only contains lowercase letters. The BERT model takes a sequence of tokens as input, with a maximum length of 512 tokens. As an output, BERT produces a 768-dimensional vector to represent each input sequence [21], [22].

The success of BERT has led to the emergence of many variants. BERT and its variants are designed to be exploited for various NLP tasks. Some examples include:

a. RoBERTa:

RoBERTa (Robustly optimized BERT approach) [23] is an improved version of BERT and was published in 2019 by the Facebook AI team. It has the same architecture but different pre-training steps. The improvements include training the model for a longer duration, using larger batches, incorporating more data, deleting the NSP method, training on longer sequences, and dynamically modifying the masking pattern used in the training data. All of these changes have the potential to significantly improve performance. This enhanced pre-training method yields cutting-edge outcomes.

b. Xlm-R

The Facebook AI team continued and introduced the XLM-R (Cross-lingual XLM-RoBERTa) [24] model in 2020. It was trained on data from over 100 different languages, totaling more than two terabytes. Consequently, it outperforms the multilingual mBERT in a variety of cross-lingual benchmarks. It has been noticed that pre-training multilingual language models on a large-scale lead to significant improvements in performance for various cross-lingual transfer tasks.

c. ALBERT:

ALBERT (A Lite BERT) [25], proposed in 2020, provides two parameter-reduction strategies to reduce memory usage and speed up BERT training in order to solve the existing issues of Graphics processing unit (GPU), TPU memory constraints and long training durations. The techniques used are a factorized embedding parameterization method and a cross-layer parameter-sharing method. Detailed empirical data demonstrates that their suggested approaches produce models that scale much better than the original BERT.

d. DistilBERT:

Researchers [26] also introduced a technique for pre-training a smaller, general-purpose language representation model called DistilBERT (A Distilled Version of BERT). This model can be further fine-tuned and yield good results on a variety of tasks, similar to its bigger equivalents. While most of the earlier research focused on using distillation to create task-specific models, they applied it during the pre-training phase and demonstrated that it is feasible to reduce the size of a BERT model by 40% while still maintaining 97% of its language comprehension skills and being 60% quicker.

3. METHODOLOGY

To accomplish the stated objective, this review paper will adopt a methodical approach by conducting a comprehensive literature search of studies that utilize BERT-based models for hate speech detection. The methodology used depends on several factors, such as the key terms, the data repositories, and the time limit used for the search. The key terms used include many related terms to hate, like offensive and toxic, combined with machine learning, deep learning, transformers, and BERT terms. The data repositories used for search include Google Scholar, IEEE Xplore, Science Direct, and Scopus. Since the BERT model was first proposed in 2018, the paper search is limited to a four-year range (2019 to present) for this review work. After collecting the papers, they undergo an initial analysis to ensure their relevance to our topic. Then, they are stored according to their publication date and grouped according to the taxonomy presented in the following sections.

4. BERT-BASED MODELS

Usually, models used for detecting hate speech undergo the following stages: data collection, preprocessing, feature extraction, classification, and evaluation. The feature extraction and classification steps can be treated as two independent stages or as one complementary stage. In this paper, we will outline the hate speech detection models that specifically employed BERT and their pipeline, as illustrated in Fig. 3. The following subsections provide a detailed explanation of each stage in the pipeline.

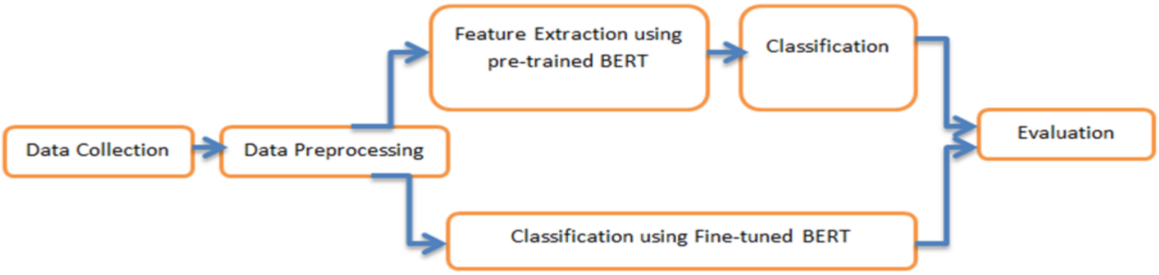
Data Collection
Data Preprocessing
Feature Extraction using pre-trained BERT
Classification
Classification using Fine-tuned BERT
Evaluation

FIGURE 3. - BERT-based hate speech detection pipeline

4.1 DATA COLLECTION

Choosing how and where to gather data for training BERT-based models is a decision that the researcher will make at this stage. A researcher might choose to obtain a published dataset or decide to start from scratch and create a new dataset. When deciding whether to use an existing dataset or create a new one, availability and relevance must be considered [18]. The dataset might not even exist or could be completely outdated. Researchers in this situation have the choice to either create a new dataset or update an existing one. Although it can be difficult, time-consuming, and expensive [27], creating a new dataset is usually worthwhile [1]. Table 1 shows the most common datasets available and used for training BERT-based models, as well as other machine and deep learning models.

A group of open challenges and disagreements have faced the dataset creators. For example, there are differences in the hate definitions, the conflation between hate and offensive labels [28], and the methods of collecting and annotating data. Moreover, another controversial issue that was noticed is the class imbalance [29]. While [7] attempted to create a balanced dataset, [30] decided to keep it unbalanced because it reflects a natural phenomenon. Later, researchers leveraging these datasets have addressed the issue of class imbalance using techniques like under-sampling or oversampling. Another clear limitation is the lack of available labeled datasets in low-resource languages and the lack of multi-class labels. Many available datasets are also biased and suffer from overfitting [31]. It has been observed that a majority of datasets are sourced from Twitter, indicating the need of including other platforms during the training phase in order to attain more generalizability.

Table 1. - Some popular training datasets used by BERT-based models for hate speech detection task

Dataset	Total records for each Language	Source	Classes
Davidson 2017 [28]	24,802 English	Twitter	3 classes: Hate, Offensive not hate, Neither hate nor offensive
SemEval-2019 [32]	13,000 English 6,600 Spanish	Twitter	2 classes for several tasks: Hate or not, Group or individual, Aggressive or not

HASOC2019 [33]	5,852 English 3,819 German 4,665 Hindi	Twitter Facebook	2 classes: NOT (non-hate-offensive) and HOF (hate and offensive) 3classes: HATE, OFFN (offensive) and PRFN (profane)
HASOC2020 [34]	3,700 English 2,373 German 2,963 Hindi	Twitter	2classes: NOT and HOF 3classes: HATE, OFFN and PRFN
HASOC2021 [35]	3,843 English 4,594 Hindi 1,874 Marathi	Twitter	2classes: NOT and HOF 3classes: HATE, OFFN and PRFN
Waseem & Hovy [30]	16,000 English	Twitter	3 classes
ETHOS, 2021 [7]	998 binary labeled English 433 multi-labeled English	Social media platforms	2classes 8 classes

4.2 PREPROCESSING

The preparation of text data is a crucial step in making the text simpler to extract information from. Some preprocessing steps are general, while others are special for BERT models. Choosing the preprocessing steps should be done carefully since they can have a vital effect on the model's performance. The general steps aim to remove irrelevant noise in order to determine the sentiment of social media posts [18]. After analyzing numerous research papers, it was observed that the preprocessing steps differ among researchers. Some steps are standardized and agreed upon by all researchers, like stemming or lemmatization [36], [37]. These steps include converting all characters to lowercase for English texts [18], [19], [20], recognizing lengthy words and shortening them to conform to conventional usage (e.g., converting "noooooo" to "no") [20], [38], removing all punctuation marks, extra white spaces, and unknown characters in most cases [20], [39], and deciding whether to keep or eliminate stop words [40], [39].

Other preprocessing steps are inconsistently considered, and researchers handle certain text features differently. For example, URLs and mentions of users are sometimes considered unhelpful and therefore removed [18], [19], while other times they are kept and addressed [39]. Hashtag texts and emojis have the same issue. Some consider them to be useful elements in the detection task, so they replace hashtags with separate tokens [36], [2] and replace emoticons with alternative words [22], [20]. On the contrary, others totally remove them [19], [41]. This clear difference confirms the need for research in the future to analyze the impact of these symbols on the classification task.

Finally, some preprocessing steps are mandatory when dealing with BERT models. As mentioned in Section 2, the BERT tokenizer adds two segments, [CLS] and [SEP], to the sentence. [CLS], which is the first token in the input, is considered the classification token, while [SEP] is used to separate two sentences in certain tasks [12].

4.3 FEATURE-BASED AND FINE-TUNING APPROACHES

Following the selection of the dataset and the application of preprocessing, the subsequent steps are to obtain text representations and conduct classification. Here, previous researchers applied two BERT-based model-based approaches. Some used it solely for text feature extraction, followed by another classification method, while others fine-tuned it specifically for text classification and hate detection tasks. The following subsections illustrate the situation.

4.3.1 Using BERT-Based Models for Feature Extraction

Typically, texts are unstructured data. However, since mathematical modeling is a fundamental component of all machine and deep learning algorithms, the unstructured character of text input must be transformed into a structured feature space. Deep learning methods are extensively used to learn the representations and not only process these representations to extract the output [42]. After cleaning the dataset, it can be transformed into a vector space using feature representation techniques [1]. Many classical text representation methods were used, such as Term Frequency-Inverse Document Frequency (TF-IDF) and Bag-of-Words. However, BERT was one of the transformer-based language models that outperformed others.

Below are models that use BERT only for feature extraction, followed by another classification method. These models treat feature extraction and classification as independent steps. Their contributions and results are illustrated. Table 2 provides a brief description of these models, including the versions used, classification algorithms, datasets, languages, and types of classification. The models giving the best results are written in bold. Each model was used to handle a certain challenge or problem from a set of open challenges in the hate speech detection task.

a) Lack of Generalization

The authors of [43] considered that there is no assurance that the single-platform models will effectively generalize across platforms. Therefore, the researchers merged datasets from four different platforms and tested a variety of feature extraction techniques, including BERT. Other traditional techniques that were tested include Bag-of-Words, TF-IDF, Word2Vec, and their combinations. The output of these methods was also tested on a group of classification algorithms, including Logistic Regression, Naive Bayes, SVM, XGBoost, and Neural Networks. According to the results, BERT traits have the greatest influence on the predictions. Moreover, researchers [44] merged four datasets to assess the generalizability of their proposed model. Their model employed a one-class approach, where the detection classifier is exclusively trained on hate-class examples. Their model used a one-class SVM for classification and a BERT-BiLSTM module for feature extraction. An extensive analysis showed that their model performed better than existing methods and confirmed the benefits of training the model using a combination of datasets.

b) Lack of Training Data

The disagreement on the hate definition and the complexity and high cost of the annotation process are the main factors leading to a lack of annotated data [26]. Researchers [27] focused on this problem. The problem of a lack of data may also result in other subproblems, like bias. Consequently, they examined the impact of using more data, both labeled and unlabeled. They accomplished this by utilizing a variety of traditional machine learning techniques as well as several deep learning models. One approach employed the RoBERTa model as a feature extractor, and then the output was fed into traditional machine learning techniques. According to their findings, incorporating additional labeled data from a separate data collection often proved beneficial.

c) Explainability

Machine learning techniques work as a black box and do not offer a clear explanation of how the output was produced [45]. That is why explainable artificial intelligence (XAI) is needed. It is a new level of artificial intelligence in which we can seek answers to “why” questions that were previously impossible. In the work of [45], two datasets that used XAI (Google Jigsaw and HateXplain) were employed. Exploratory data analysis was applied to both datasets to identify various trends and gain insights. Different classifiers were implemented using explainable methods like local interpretable mode (LIME). In comparison to the other models, the combination of BERT with artificial neural networks (ANN) and BERT with Multilayer Perceptron (MLP) showed the best performance in terms of explainability.

d) Preprocessing Impact

One of the few research projects that studied the impact of preprocessing on classification performance was conducted by [46]. Utilizing the HASOC2021 dataset, BERT-based architecture and word- and character-based LSTM models were implemented to categorize tweets into offensive and non-offensive categories. The text data was converted into vectors using the BERT language model, and the resulting vectors were fed into a Gated Recurrent Units (GRU) network. In addition to the superiority of BERT, the results showed that trials with preprocessed data outperformed the others.

e) Multilingual System

Multilingual hate speech detection is the process of classifying a set of texts written in different languages while adhering to a fixed set of labels across languages [21]. In the study by [47], the goal was to achieve online multilingual hate speech recognition: abusive, hateful, or neither. The work included using six publicly accessible datasets, which were pooled into a single homogenous dataset. These datasets were then categorized into three distinct labels. The architecture of the proposed model was as follows: The performance of Hindi and English contextual word embeddings in the multilingual model was captured using pre-trained BERT embeddings. The output was then fed to a bidirectional LSTM. The suggested model outperformed a variety of baseline monolingual models, yielding similar or better results. The model operates in an online setting in close to real-time and produces competitive performance on pooled data.

f) Unbalanced Data

1.1. As we have mentioned previously, researchers have contradictory opinions about the issue of unbalanced data. In [41], they introduced a novel BiLSTM with deep CNN and Hierarchical ATtention-based model (BiCHAT) and compared its performance on both balanced and unbalanced dataset. The tweets are entered into the proposed model, which subsequently runs them through a BERT layer and an attention-aware deep convolutional layer. An attention-aware bidirectional LSTM network is used to further process the convolutionally encoded representation. Through a softmax layer, the model assigns either a hostile or neutral label to the tweet. The experimental analysis revealed that the BiCHAT model performed better than the current standard approaches. Additionally, the performance of unbalanced datasets was better than that of balanced ones.

g) High False-Positive Rates

Researchers in [6] improved the performance of hate speech detection in terms of specificity, indicating that the model is excellent at correctly identifying non-hate speeches. This issue is very important because it protects the user's freedom. They examined the impact of static BERT embeddings and neural networks on their results in hate speech identification. As a result, it marks less offensive non-hate speech as such, preserving the right to freedom of expression.

h) Low-Resource Languages

Another challenge is the limited availability of datasets and models for low-resource languages [48], [49], [50]. The authors of [48] discussed their technique for identifying objectionable language in Dravidian languages. The semantic information characteristics of the text were extracted using the XLM-RoBERTa pre-training model, and the output features were further processed using deep pyramid convolutional neural networks (DPCNNs). The training impact is also improved by using the hierarchical cross-validation approach. The final findings demonstrated that the model performed satisfactorily. In [49], the language of the training dataset was Bengali. The results of two different models were combined. The first model contains LSTM with BERT. The second model contains the AdaBoost algorithm with BERT. The accuracy of the proposed model exceeded that of the baselines. On the other hand, cross-lingual models were used to handle low-resource languages (as mentioned in [50]). They created a customized architecture using frozen, pre-trained Transformers to investigate cross-lingual zero-shot and few-shot learning, as well as unilingual learning, using the HatEval challenge dataset. BERT and XML were employed for feature extraction. On the English and Spanish subsets, they achieved extremely competitive results with their unique attention-based classification block, AXEL.

Table 2. - BERT-based models for feature extraction

Ref	year	Feature Extraction	Classification Model	Dataset	Language	Type of classification	Best Result
[43]	2020	Bag-of-Words TF-IDF Word2Vec **BERT**	Logistic Regression Naïve Bayes Support Vector Machines **XGBoost** Neural Network	Youtube dataset Reddit dataset Wikipedia (Kaggle-18) Twitter (Davidson)	English	Binary	F1=0.916
[50]	2020	BERT, **XLM**	LSTM novel classification block **AXEL**	HatEval dataset: part of the SemEval task5	English and Spanish	Binary	F1=0.711
[47]	2020	TF-IDF, POS Word embedding **BERT**	Logistic Regression Bi-LSTM+CNN **LSTM**	HASOC2019 TDavidson Elsherif Ousidhoum SemEval 2019 Task 5 PMathur	English Hindi	Binary	F1=0.92 on Davidson dataset
[27]	2021	**RoBERTa**	SVM Logistic Regression Linear Discriminant K-NN Random Forest **Ensemble of Roberta and Fasttext**	HASOC2019 OLID	English	Binary	Macro f1=0.794
[48]	2021	**XLM-**	Linear classifier	Dravidian	Dravidian	Multi-class	F1=0.92

		RoBERTa + DPCNN		language EACL2021			
[6]	2021	Fasttext GLove **BERT**	GRU CNN+Attention Bi-LSTM+Attention CNN+Bi-LSTM **BiLSTM**	ETHOS	English	Binary	F1=0.79
[49]	2021	TF-IDF Word2vec Fasttext **BERT**	LSTM AdaBoost **L-Boost**	newly created dataset	Bengali	Binary	F1=0.918
[46]	2022	**BERT**	Char-LSTM Word-LSTM **GRU network**	HASOC 2021	Indo-European languages	Binary+ Multi-class	F1=0.86
[41]	2022	Glove Word2vec **BERT**	CNN LSTM Bi-LSTM GRU **BiCHAT**: BiLSTM+CNN+ attention	Founta et al. Davidson et al.	English	Multi-class	F1=0.91 on dataset 2 F1=0.84 on dataset 1
[44]	2022	**BERT+ BiLSTM**	One Class SVM	HASOC 2019 SemEval Stormfront 2019 Davidson	English	One class	F1=0.88 on Stormfront dataset
[45]	2022	**BERT**	MLP **ANN**	Google Jigsaw HateXplain	English	Multi-class	F1=0.941 on HateXplain dataset

4.3.2 Using Fine-Tuned BERT for Classification

As mentioned, fine-tuning refers to the process of training the pre-trained BERT model using task-specific datasets, along with the additional untrained classifier layers of 768 dimensions. This technique can be applied in different ways. Below is a list of hate speech detection models that utilize BERT for both feature extraction and classification purposes. Table 3 illustrates the methods, different fine-tuning techniques, datasets, languages, and types of classification used. The models achieving the best results are written in bold.

a) Lack of Generalization

Cross-dataset or cross-domain is one of the techniques used to estimate generalization [51]. Researchers in [52] comprehensively studied the concept of cross-dataset model generalization using nine public datasets and employed different models, including BERT and ALBERT. They applied a special normalization method to compare the class labels in the datasets. They concluded that transformer-based models when using specific, precise, and non-correlated hate speech categories, can improve model generalization. Also, in [53], they analyze the generalizability using four datasets and various methods. They revealed that generalizability is affected by the combination of datasets more than the methods used.

Another approach to handling generalization problems is to conduct multi-task learning [2]. It means to benefit from datasets for relevant tasks. It was applied to some research papers like [2], [54], [55]. After fine-tuning the BERT models, the authors in [56] added multi-task learning. They found that the best results were achieved by testing with in-domain data while using out-domain data resulted in worse outcomes. The authors of [55] proposed the first multi-task technique that leverages shared emotional knowledge to detect hate speech in Spanish tweets using BERT-based models. Their findings suggest that the combination of polarity and emotional knowledge improves the detection of hate speech across datasets. In addition, [54] considers that the central concept of multi-task learning is to use identical tasks as model regularizers. This is accomplished by adding the specific loss functions for each task to the model's overall loss function. Thus, the model is compelled to simultaneously optimize for all the various tasks, resulting in a model that can generalize across multiple tasks in the dataset. Moreover, the authors in [2] suggested a new model called AngryBERT, which relies on multi-task learning to address the challenges of imbalanced data and data scarcity. In addition, they applied data augmentation to three popular datasets and showed the superior performance of their model compared to other baselines. AngryBERT succeeded in accurately identifying hate speech. In contrast, the study conducted by [57] compares the use of domain-specific word embedding followed by a classification algorithm with utilizing BERT. BERT outperformed the first approach slightly. Despite the significant variation in corpus size, the first approach attained results that were very close to those of BERT. BERT is trained on a massive corpus of data, but the first approach is trained on data from the same domain.

b) Bias

When a model makes more mistakes in classifying due to the existence of keywords, it is deemed biased [58]. To reduce bias, the authors of [58] focused on examining the association between hate speech detection models and a collection of hateful words drawn from three well-known datasets. The experimental findings showed that fine-tuning the models using hateful texts without the presence of keywords may minimize bias towards hateful keywords. This reduction in bias may lead to an improvement in classification performance. Another paper [22] implemented generalization techniques to reduce bias in datasets. To achieve this goal, the input samples were reweighted using a well-established regularization technique. This technique helps reduce the impact of highly correlated n-grams from the training set on the class labels. After reweighting, the pre-trained BERT-based model was fine-tuned using the newly reweighted data. Two publicly accessible datasets, which have been annotated for racist, sexist, hateful, or other harmful content on Twitter, were used to assess the proposed model. The proposed model was able to reduce racial bias. The authors of [59] suggested a new idea of personalized, human-centered NLP that depends not only on the text but also on the user.

c) Explainability

A group of research papers [36] and [60] tried to achieve good explainability. They introduced HateXplain, the first hate speech benchmark dataset with human-level explanations. Each post in this dataset is annotated from three perspectives: the basic, widely used three-class classification—hate, offensive, or normal; the target community—the community that has been the target of hate speech/offensive speech in the post; and the rationales, which are the parts of the post on which the labeling decision (as hate, offensive, or normal) is based. They used existing state-of-the-art models and discovered that even those that excel at classification perform poorly on explainability criteria, such as model plausibility and fidelity. Furthermore, they discovered that models that incorporate human training goals are more effective in reducing unintended bias. While in [36], they proposed a novel model called DeepHateExplainer to reduce ambiguity and improve explainability. They employed sensitivity analysis (SA) and Layer-wise Relevance Propagation (LRP) to provide local and global explanations. This research demonstrated that feature selection can have a non-trivial influence on the learning capabilities of machine and deep learning models.

d) Unbalanced Data

Some researchers have investigated methods for handling unbalanced data. Researchers in [61] have proved the efficiency of a multilingual architecture by using several transformer-based MLMs (such as mBERT, XLM-RoBERTa, and DistilmBERT). They experimented with SOUP (similarity-based oversampling and undersampling processing) to address the issue of unbalanced data in the HASOC2021 competition. However, they found that the classification accuracy decreased. An alternative solution was to utilize the class weight procedure to achieve a balanced dataset [79].

e) Fine-Tuning

Researchers in [20] focused on the fine-tuning procedure itself. They proposed a novel transfer learning approach based on BERT. Instead of using the classical method of adding a simple classification layer, they implemented four new methods to fine-tune the BERT model for the required classification task. These methods include adding a fully connected network with a softmax activation function, adding nonlinear layers, adding a Bi-LSTM layer, or adding a CNN layer. For evaluation purposes, they used two publicly available datasets that had been labeled for instances of racism, sexism, hate, or offensive content on Twitter. The results showed that fine-tuning by adding CNN layers outperformed all other methods. In contrast, the paper by [19] conducted exhaustive experiments to investigate various fine-tuning methods of BERT on a text classification task and presented a general solution for BERT fine-tuning. They further investigated pre-training and multi-task fine-tuning. Within-task and in-domain pre-training achieved the best performance improvement.

f) Single Classifiers

Several approaches were used to improve the performance of the classifier. Firstly, the fusion of two viewpoints was applied to enhance the performance in [5]. The principles of three different text categorization methods, ELMo (Embeddings from Language Models), BERT, and CNN, were outlined and applied to the detection of hate speech. The outcomes demonstrated that fusion processing is an effective strategy for improving hate speech detection efficiency. It can be said to be acceptable to pay a little bit more to get performance that has practical use. In the study by [62], a basic ensemble of transformers was proposed for the task of detecting hate speech. The results of the HASOC challenge demonstrated that this ensemble could achieve state-of-the-art performance. Furthermore, they were able to improve the outcomes obtained through additional pre-training using in-domain data.

g) Pre-Training on General Data

While the BERT model is pre-trained using general data, the authors of [63] attempted to further pre-train the BERT model using datasets containing hateful content. The new model, called HateBERT, outperformed BERT in hate and offensive speech detection tasks.

h) Unimodal Systems

Most previous work concerning hate speech has focused solely on text, while many future work suggestions recommend expanding the scope. For this reason, researchers [64] developed a multimodal system called SocialHaterBERT that considers both user characteristics and the text. This contribution significantly enhanced the results. The work [65] also implemented a multimodal model, but instead of handling text only, it incorporated images. They focused especially on detecting racist speech against migrants in Greece.

i) The Challenge of Time

The authors of [66] assessed the temporal resilience of several hate speech prediction systems in terms of language and topic change over time by conducting two experiments. In the first experiment, they trained the models on data from a single month and tested them on data from the next month. In the second scenario, they expanded the size of the training set by introducing recent historical information. This was done by utilizing data from all months before the one from which the test sample was drawn. Results showed that injecting recent data into the training corpus significantly improved the classification performance.

j) Multilingual, Cross-lingual, and Code-Mixed

As mentioned previously, multilingual systems have the capability to classify text written in different languages. This problem is distinct from cross-lingual text classification, in which a text written in one language must be categorized by a classification system learned in another language [21]. The authors of [21] managed multilingualism by experimenting with two approaches: a joint-multilingual approach that aims to build a single classification system and a translation-based approach that requires translation before the classification step. Combining the translation-based method with AraBERT outperformed others. The work of [38] also increased performance by using translation. They proposed a multichannel model that exploits three versions of BERT: English, Chinese, and multilingual BERTs for hate speech detection. Also, they attempted to enhance the input by employing translation. They performed fine-tuning by extracting the representation of the [CLS] token from the last layer of the BERT model and pooling it using the pooling layer. They then added a dropout layer for regularization, followed by a fully connected feed-forward layer and a softmax layer for classification. Their results showed the success of the multichannel fine-tuning model compared to other state-of-the-art models. The goal of the study [67] was to automatically classify hate speech and objectionable content using datasets in different languages. They carried out several tests utilizing transfer learning models, such as the pre-trained BERT model and the multilingual BERT model. The multilingual BERT is similar to BERT, but it was pre-trained in multiple languages.

To implement cross-lingual classification, researchers in [68] merged two datasets in a cross-lingual format. They utilized two versions of transfer-based models, mBERT and xlm-Roberta, to detect the presence of hate. To predict the class, they fine-tuned the models by inserting a dropout layer with a ReLu activation function and two linear layers. They concluded that cross-lingual hate speech detection is a viable solution for the problem of limited training datasets, especially in non-English data. For the same purpose, the work conducted by [69] employed a zero-shot cross-lingual design in which they only used English-labeled data to identify hate speech in German. They chose the English training and German test datasets based on their similarities in hate speech definitions. Their experiments demonstrated the efficiency of the cross-lingual technique.

Code-mixed text (text written in more than one language) is also common on social media platforms. Consequently, [70] provided automated methods for detecting hate speech in code-mixed text scraped from Twitter. They paid particular attention to techniques that use transformers and mixed English-Hindi content. While conventional methods examine language on its own, they additionally leverage content text in the form of parent tweets. In single-encoder and dual-encoder scenarios, they attempted to assess the performance of multilingual BERT and Indic. The initial strategy involved using a separator token to concatenate the target text and context text in order to obtain a unified representation from the BERT model. The two texts were independently encoded using a dual BERT encoder in the second method, and the resulting representations were then averaged. They demonstrated that the dual-encoder method with independent representations produces superior results.

k) Low-Resource Languages

Since most of the training datasets available are in English, some researchers have shifted their focus to other low-resource languages. In [71], a suitable method for low-resource languages was proposed. They conducted experiments in nine languages and from different sources. They used the BERT model in combination with other methods in both monolingual and multilingual forms. The results suggested using Language-Agnostic SEntence Representations (LASER) + Logistic Regression for low-resource datasets, while BERT and mBERT are more suitable for high-resource datasets. In this study, the monolingual models, especially MahaBERT, yielded the best results on the given dataset. Similarly, in [72], they compared basic Marathi monolingual models (MahaBERT, MahaALBERT, and MahaRoBERTa) to common multilingual models like mBERT, indicBERT, and xlm-RoBERTa. They further demonstrated that, in five distinct downstream fine-tuning studies, Marathi monolingual models outperformed the multilingual BERT versions.

For the Arabic language, several papers were concerned [39], [73], [74], [75]. All of them created a new Arabic dataset crawled from Twitter in order to conduct their experiments [72]. As an example, [38] examined various models of RNN and CNN to detect hate speech. Then, they ran a series of tests on two datasets to compare the performance of four models: CNN, CNN + GRU, BERT, and GRU. The outcomes of the research were encouraging and demonstrated the usefulness of the suggested models for the detection task. While in [74], a new model called the Arabic BERT-mini model (ABMM) was proposed, trained, tested, and evaluated on the newly created corpus. The results outperformed other models evaluated on Arabic datasets. Many BERT variants were trained on an available Arabic language dataset, including both standard and dialects [75]. They also trained them to translate Arabic texts into English. The English BERT outweighed AraBERT, possibly due to the larger training datasets, while the multilingual BERT performed the worst.

Dravidian, Bengali, Hindi, and Marathi languages were also studied as low-resource languages. In [76], the authors proposed pooling the last layers of the multilingual BERT model and MuRIL(Multilingual Representations for Indian Languages) model for the task of detecting offensive content in Dravidian mixed languages. The results were encouraging; for example, [37] deals with the underserved Bengali language. They presented a method for identifying hate speech in a multimodal context, which involves analyzing both textual and visual information. The present study investigated the relevance of feature extraction to the learning capabilities of ML and DNN models. It was found that memes were highly effective in detecting hate speech in Bengali. None of the multimodal models outperform the unimodal algorithms that solely analyze textual data. XLM-RoBERTa outperformed all transformer models and proved to be the most suitable. In [77], the work was similar; however, the difference was that researchers proposed a hierarchical approach for identifying hate and offensive speech in Hindi and Marathi languages. The first step in the hierarchy is binary classification. The binary classification was applied in two different ways. One way to perform feature extraction and classification is by using BERT-based models, such as IndicBERT, Mbert, HiRoberta, and MrRoberta. The alternative approach involves using FastText embedding and employing various deep-learning architectures for the classification. The transformer-based models gave the best results. In the second step, a multi-class classification was applied. In the same context and to enhance the resources for the Marathi language, the work [62] created the first dataset for detecting hate speech in Marathi called L3CubeMahaHate. The dataset is large because it consists of more than 25,000 tweets categorized into four classes: hate, offensive, profane, and not. After that, this dataset was used to train various deep learning models for classification, such as CNN, LSTM, and transformers. Among transformers, they utilized monolingual and multilingual variants of BERT like Ma-haBERT, IndicBERT, mBERT, and xlm-RoBERTa.

The authors of [78] have proposed a new model by fine-tuning a larger pre-trained model, AlBERTo, for the Italian language. The new model was fine-tuned and assessed depending on the HaSpeeDe dataset. Four separate tasks were assigned to the data, including phrases taken from Facebook and Twitter. The first two models were trained on data from the same domain as the test data. The remaining two “cross” tasks, on the other hand, required the classification of data from a domain other than the training domain. When the model is assessed using data from the same distribution as the training data, the results demonstrate great performance.

The first Roman Urdu pre-trained BERT version was created in [82] by pre-training BERT on a huge amount of Roman Urdu datasets. The model was evaluated using different machine-learning techniques. The proposed model combined pre-trained models with deep-learning models and outperformed others.

In order to offer carefully calibrated reliability estimates, researchers in [79] developed a Bayesian technique that employs Monte Carlo dropout within the attention layers of transformer models in different low-resource languages. They assessed and represented the outcomes of the suggested technique on issues with detecting hate speech in several languages. Additionally, they examined whether emotional factors could improve the data collected by the BERT model for classifying hate speech. Their research showed that Monte Carlo dropout offers a workable method for estimating reliability in transformer networks. It provided cutting-edge classification performance when used within the BERT model and had the ability to identify less reliable predictions.

Table 3. - Fine-tuned BERT-based models for classification tasks

Ref	**Year**	**Bert version**	**Fine-tuning approach**	**Dataset**	**Languages**	**Type of classification**	**Best Result**
[20]	2019	**BERTbase**	Fully connected network Add nonlinear layers Add bi-LSTM layer	Waseem&Hovey Davidson	English	Multi-class	F1=0.88 on Waseem dataset

			Add CNN layers				F1=0.92 on Davison dataset
[78]	2019	**AlBERTo**	Adding a custom classification layer	Facebook and Twitter datasets	Italian	binary	F1=0.79
[38]	2019	BERT MBERT **Multichannel-BERT**	Pooling layer + dropout layer+feed-forward layer+softmax layer	HatEval(SemEval 2019 task 5) GermEval HaSpeeda	Spanish Italian German	binary	Macro F1= 0.799 on HaSpeeda dataset
[5]	2020	ELMo CNN BERT Fusion of BERT+ CNN+ELMo **Fusion of 3 CNN**	Adding classification layer	SemEval 2019 Task 5	English	Binary	F1=0.712
[22]	2020	**BERT**	BERT based fine-tuning Insert nonlinear layers Insert Bi-LSTM layer **Insert CNN layer**	Waseem and Hovy Waseem Davidson	English	Multi-class	F1=0.81 F1=0.91
[40]	2020	RoBERTA **5-fold ensemble Roberta models**	Adding classification layers	HASOC2019 OffensEval	English	Binary	Macro f1= 0.802
[39]	2020	GRU **CNN** CNN+GRU Multilingual BERT	adding classification layer	New Arabic dataset	Arabic	Multi-class	F1=0.79
[67]	2020	SVM ELMo+ SVM **BERT** **m-BERT**	Adding a fully connected neural network + softmax activation function	HASOC2020	English German Hindi	Binary	F1=0.88
[71]	2020	CNN-GRU **LASER+LR** **Translation+ BERT** **m-BERT**	Adding classification layer	Multiple datasets Davidson et al. Waseem et al Basile et al. Founta et al. …...	Arabic English German Indonesian Italian Polish Portuguese Spanish French	Binary	F1=0.71 for English F1=0.83 on Arabic datas
[56]	2020	mBERT XLM-RoBERTa AlBERTo, **UmBERTo** PoliB-ERT	add a simple linear layer with a softmax on top of it	HaSpeeDe 2 dataset	Italian	Binary	F1=0.809
[79]	2020	MCD LSTM BAN(Bayesian Attention Network) BERT **MCD BERT (MonteCarlo Dropout)**	Adding drop out	Davidson Croatian dataset Slovene dataset	English Croatian Slovene	Binary	F1=90.4
[66]	2020	**AlBERTo** SVM	Adding a dense layer with a softmax function	TWITA Haspeede+	Italian	binary	F1=0.69
[19]	2020	BERT	Adding classification layer	8 datasets IMDb Yahoo! Answers…	English Chinese	binary	
[36]	2021	Bangala BERT-base m-BERT XLM-RoBERTa	Adding a fully connected softmax layer	Bengali Hate Speech Dataset	Bengali	Multi-class	F1=0.88

[2]	2021	CNN LSTM BERT CNN-GRU DeepHate SP-MTL MTL-Gatedencoder **AngryBERT**	BiLSTM layer + MLP layer	FOUNTA Davidson WZ-LS	English	Multi-class	F1=0.90 on Davidson
[21]	2021	BERT **AraBERT**	Adding feed-forward network	Semi-supervised Offensive Language Identification Dataset (SOLID)	multilingual	binary	F1=0.93
[68]	2021	**m-BERT** **Xlm-Roberta**	Adding a drop-out layer	MLMA CONAN	English French Cross-lingual	binary	F1=0.67
[61]	2021	m-BERT DistilmBERT **XLM-RoBERTa**	Add classification layer	HASOC 2021	English Hindi Marathi	Binary+ multi-class	F1=0.79 on English
[80]	2021	**m-BERT** **XLM-R** Indic-BERT Dehate-BERT	Adding a classifier layer	HASOC 2019	Hindi English Marathi	Binary+ multi-class	F1=0.80
[70]	2021	m-BERT, Indic-BERT **Ensemble of 4 BERT variants**	Dense layers + softmax classifier	HASOC 2021 ICHCL task	Code-mixed languages	binary	F1=0.73
[63]	2021	BERT **HateBERT**	Adding a custom classification layer	OffensEval 2019 AbusEval HatEval	English	Multi-class	F1=0.809
[60]	2021	CNN-GRU **BiRNN** **BERT** BiRNN-Att BERT	Add a fully connected layer	HateXplain	English	Multi-class	F1=0.68
[77]	2021	m-BERT **IndicBERT** **RoBERTa-Hi**	Adding classification layer	HASOC 2021	Hindi Marathi	Binary+ Multi-class	F1=0.86
[52]	2021	SVM FastText **BERT** **ALBERT**	Adding classification layer	9 datasets: Davidson Waseem&Hovy Kaggle…..	English	Multi-class	F1=0.92
[76]	2021	Mbert **MuRIL**	Custom pooled output	HASOC 2021	Tanglish	Binary	F1=0.67
[55]	2021	mBERT BETO Multichannel BERT SVM **Proposed MTL**	Adding classification layer	HatEval MEX-A3T	Spanish	binary	F1=0.86
[65]	2021	BERTaTweetGR **greek-bert** **resnet18 +greek-bert(multimodal)**	Adding linear layer	Create a new multimodal dataset	Greek	binary	F1=0.947
[54]	2021	**mBERT** **multi-task approach (MTL)**	Adding classification layer	HASOC2019	English Hindi German	Multi-class	F1=0.84
[62]	2022	CNN LSTM Bi-LSTM **MahaBERT**	Adding classification layer	Create a new dataset L3Cube-MahaCorpus	Marathi	Multi-class	Accuracy= 0.909

		IndicBERT mBERT and xlm-RoBERTa					
[37]	2022	Bangla BERT mBERT **XLM-RoBERTa** **XLM-RoBERTa+ DenseNet**	Fully connected softmax layer	Extend Bengali Hate Speech Dataset	Bengali	Binary	F1=0.82
[72]	2022	mBERT, indicBERT xlm-RoBERTa MahaBERT **MahaALBERT,** **MahaRoBERTa**	Adding a custom classification layer	HASOC-2021 L3Cube-MahaHate	Marathi	binary	Accuracy= 0.89
[75]	2022	**BERTen** AraBERT mBERT LSTM LinearSVC	Adding classification layer	Arabic dataset	Arabic	binary	F1=0.98
[74]	2023	AraBERT CNN-LSTM Decisiom Tree **Arabic BERTMini Model ABMM**	Dropout layer+ fully connected layer	New dataset	Arabic	Multi-class	F1=0.986
[53]	2023	**BERT** CNN LG CNB LSVC	Adding classification layer	GermEval2018 GermEval2019 GermEval2021 ETHOS	German	binary	F1=0.726
[64]	2023	BERT BETO **SocialHaterBERT**	Adding classification layer	ACHaterNet	Spanish	binary	F1=0.802
[58]	2023	**BERT** RoBERTa	Adding classification layer	Hateval W&H Founta	English	binary	F1=0.8004
[57]	2023	LR Bi-LSTM **BERT**	Adding classification layer	Davidson Waseem Waseem	English	binary	F1=0.964
[69]	2023	CNN Bi-LSTM **mBERT**	Adding classification layer	Stormfront dataset GermEval2018	English German Cross-lingual	Binary	F1=0.98
[81]	2023	Mbert+ BiLSTM **BERT-English+ Bi-LSTM** BERT-RU + BiLSTM	Adding bilstm layers	Roman Urdu Hate Speech Dataset	Roman Urdu	binary	F1=0.79

4.4 EVALUATION METRICS

As a general method for detecting hate speech, the performance evaluation of BERT-based models typically involves using the classic metrics of precision, recall, and F1-score. These are mostly used due to the imbalanced nature of most hate speech datasets. For any balanced dataset, accuracy is the best metric [1]. Researchers also used micro and macro averages of these metrics. However, for unbalanced data, using micro-average-based metrics is considered unsuitable [82]. Tables 2 and 3 illustrate the best classification results obtained by each paper. It is important to note that these measures are not comparable because the work was done on different training and testing datasets, languages, and types of classification (binary or multi-class).

5 DISCUSSION

Depending on its context-based nature, bidirectionality, and pre-training on a large amount of general data, the BERT model and its variants have consistently outperformed other deep learning and machine learning methods. They have achieved superior results in addressing different challenges and issues in the field of hate speech detection. The main difference in the implementation was the utilization of BERT layers. Some used it solely as a feature extractor, while others utilized it as a complete classifier. The ability of BERT to play both roles provides researchers with the flexibility to conduct more experiments in their research. Each approach has its advantages and drawbacks, depending on the specific datasets and tasks. As we have seen, the feature-based approach enables the integration of BERT with various machine learning and deep learning classification algorithms. The fine-tuned approach depends solely on BERT, with the ability to add a few neural layers. Even when deciding to use a fine-tuned approach, there are many strategies for fine-tuning that can be experimented with.

From the above sections, it is evident that while a significant amount of research has been conducted using transformer models, especially BERT and its variants, for hate speech detection, there is a notable disagreement regarding certain controversial issues. For example, researchers disagreed about the imbalance of data. Some tried to solve the unbalanced training datasets since machine learning techniques need balanced data, while others denied this need since unbalancing is considered a natural phenomenon. Moreover, researchers disagreed about the impact of preprocessing and the importance of specific symbols and text features like hashtags, emojis, and emoticons. Some paid great attention to them, while others ignored them totally. Furthermore, the use of monolingual, multilingual, or cross-lingual models, in-domain, out-domain, cross-domain pre-training, multi-task or single-task learning, is still under research.

Many open challenges were studied. It is clear that many of these problems are interrelated and solving one can lead to solutions for others. The problem of a lack of labeled training datasets can be considered the foundation of all other problems. This is because increasing and merging data from different platforms, domains, languages, and tasks, as demonstrated in the literature, improves model generalization, reduces bias, and helps address the challenges posed by low-resource languages, multilingual texts, and cross-lingual texts. Moreover, the BERT model needs both general and specific data. Balancing between these two types of data during pre-training, further pre-training, and fine-tuning processes has had a positive effect on the model’s performance. Other problems were also handled, such as explainability, unimodal models, and code-mixed text. However, these challenges are still being researched and require further contributions and effort. Another clear limitation is that most research papers focus on binary classification, and there is a lack of multi-class annotated datasets that categorize the type of hate and the targeted group.

6 CONCLUSION AND FUTURE WORK

This work provides a review of many research publications that used BERT-based models for hate speech detection. The review covers an examination of the datasets, preprocessing steps, approaches, methods used, challenges addressed, and results obtained. Motivated by the revolution of transformers in NLP, most papers have reported superior results when using BERT and its variants.

Based on an examination of the challenges faced and the results of previous studies, it is evident that the task of hate speech detection requires highly critical and accurate results. Consequently, there are several potential future directions for research concerning data and architecture. At the data level, although there are many training datasets available online, it is still necessary to expand the training corpus by creating new labeled datasets, combining existing ones, or using data augmentation techniques. When creating new datasets, it is recommended to focus on low-resource languages, multiple platforms, and multiple labels to specify the hate categories. Employing a critical annotation process is essential to producing high-quality training data. Moreover, further studies are required on text preprocessing, particularly in developing strategies to effectively handle the unique characters found in social media texts like emojis and hashtags. At the model level, the architecture of BERT-based models can be altered, enlarged, or simplified. They can be pre-trained, further pre-trained on task-specific datasets, and fine-tuned using new strategies and datasets. One potential approach to enhancing the accuracy of findings is to build multimodal models that combine text models with image models.

FUNDING:

None

ACKNOWLEDGMENT:

None

CONFLICTS OF INTEREST:

None

REFERENCES

[1] N. S. Mullah and W. M. N. W. Zainon, "Advances in machine learning algorithms for hate speech detection in social media: A review," IEEE Access, vol. 9, pp. 88364–88376, 2021. https://doi.org/10.1109/ACCESS.2021.3089515.

[2] M. R. Awal, R. Cao, R. K.-W. Lee, and S. Mitrovic, "AngryBERT: Joint learning target and emotion for hate speech detection," in PAKDD 2021, Springer International Publishing, Cham, vol. 12712, pp. 701–713, 2021. [Online]. https://doi.org/10.1007/978-3-030-75762-5_55.

[3] Z. Mossie and J. H. Wang, "Vulnerable community identification using hate speech detection on social media," Information Processing & Management, vol. 57, no. 3, p. 102087, May 2020. https://doi.org/10.1016/j.ipm.2019.102087.

[4] P. Sharmila, K. S. M. Anbananthen, D. Chelliah, S. Parthasarathy, and S. Kannan, "PDHS: Pattern-based deep hate speech detection with improved Tweet representation," IEEE Access, vol. 10, pp. 105366–105376, 2022. https://doi.org/10.1109/ACCESS.2022.3210177.

[5] Y. Zhou, Y. Yang, H. Liu, X. Liu, and N. Savage, "Deep learning based fusion approach for hate speech detection," IEEE Access, vol. 8, pp. 128923–128929, 2020. https://doi.org/10.1109/ACCESS.2020.3009244.

[6] G. Rajput, N. S. punn, S. K. Sonbhadra, and S. Agarwal, "Hate speech detection using static BERT embeddings," in Big Data Analytics, Springer International Publishing, Cham, pp. 67–77, 2021. [Online]. https://doi.org/10.1007/978-3-030-93620-4_6.

[7] Mollas, I., Chrysopoulou, Z., Karlos, S. et al., "ETHOS: a multi-label hate speech detection dataset", Complex & Itelligent Systems, vol. 8, no. 6, pp. 4663–4678, Jun. 2022. https://doi.org/10.1007/s40747-021-00608-2.

[8] M. Fazil, S. Khan, B. M. Albahlal, R. M. Alotaibi, T. Siddiqui, and M. A. Shah, "Attentional multi-channel convolution with bidirectional LSTM cell toward hate speech prediction," IEEE Access, vol. 11, pp. 16801–16811, 2023. https://doi.org/10.1109/ACCESS.2023.3246388.

[9] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based model for Arabic language understanding," in LREC, Marseille, France, 2020. [Online]. https://doi.org/10.48550/arXiv.2003.00104.

[10] A. Vaswani et al., "Attention is all you need." in NIPS, 2017.

[11] M. S. Jahan and M. Oussalah, "A systematic review of hate speech automatic detection using natural language processing," Neurocomputing, vol. 546, p. 126232, Aug 2023. https://doi.org/10.1016/j.neucom.2023.126232.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," Oct. 2018, [Online]. Available: http://arxiv.org/abs/1810.04805.

[13] F. Alkomah and X. Ma, "A literature review of textual hate speech detection methods and datasets," Information (Switzerland), vol. 13, no. 6, p. 273, May 2022. https://doi.org/10.3390/info13060273.

[14] S. MacAvaney, H. R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Hate speech detection: Challenges and solutions," PLoS One, vol. 14, no. 8, p. e0221152, Aug. 2019. https://doi.org/10.1371/journal.pone.0221152.

[15] J. A. García-Díaz, S. M. Jiménez-Zafra, M. A. García-Cumbreras, and R. Valencia-García, "Evaluating feature combination strategies for hate-speech detection in Spanish using linguistic features and transformers," Complex & Intelligent Systems, vol. 9, no. 3, pp. 2893–2914, 2023. https://doi.org/10.1007/s40747-022-00693-x.

[16] S. Ravichandiran, *Getting started with Google BERT: Build and train state-of-the-art natural language processing models using BERT*, USA: Packt Publishing Ltd, 2021.

[17] T. Lin, Y. Wang, X. Liu, and X. Qiu, "A survey of transformers," AI Open, vol. 3, pp. 111–132, 2022. https://doi.org/10.1016/j.aiopen.2022.10.001.

[18] B. Wei, J. Li, A. Gupta, H. Umair, A. Vovor, and N. Durzynski, "Offensive language and hate speech detection with deep learning and transfer learning," Aug. 2021, [Online]. Available: http://arxiv.org/abs/2108.03305.

[19] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?," in CCL, Springer International Publishing, Cham, 2019, pp. 194–206 [Online]. https://doi.org/10.1007/978-3-030-32381-3_16.

[20] M. Mozafari, R. Farahbakhsh, and N. Crespi, "A BERT-based transfer learning approach for hate speech detection in online social media," in Complex Networks and Their Applications, Springer International Publishing, Cham, 2019, pp. 928–940 [Online]. https://doi.org/10.1007/978-3-030-36687-2_77.

[21] F. zahra El-Alami, S. Ouatik El Alaoui, and N. En Nahnahi, "A multilingual offensive language detection method based on transfer learning from transformer fine-tuning model," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 8, pp. 6048–6056, Sep. 2022. https://doi.org/10.1016/j.jksuci.2021.07.013.

[22] M. Mozafari, R. Farahbakhsh, and N. Crespi, "Hate speech detection and racial bias mitigation in social media based on BERT model," PLoS One, vol. 15, no. 8, p. e0237861, Aug. 2020. https://doi.org/10.1371/journal.pone.0237861.

[23] Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," Jul. 2019, [Online]. Available: http://arxiv.org/abs/1907.11692.

[24] A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," Nov. 2019, [Online]. Available: http://arxiv.org/abs/1911.02116.

[25] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “ALBERT: A lite BERT for self-supervised learning of language representations,” Sep. 2019, [Online]. Available: http://arxiv.org/abs/1909.11942.

[26] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,” Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.01108.

[27] G. Kovács, P. Alonso, and R. Saini, “Challenges of hate speech detection in social media: Data scarcity, and leveraging external resources,” SN Computer Science, vol. 2, no. 2, Apr. 2021. https://doi.org/10.1007/s42979-021-00457-3.

[28] Davidson, D. Warmsley, M. Macy, and I. Weber, “Automated hate speech detection and the problem of offensive language,” in ICWSM, vol. 11, no. 1, pp. 512–515, May 2017. https://doi.org/10.1609/icwsm.v11i1.14955.

[29] J. Kocoń, A. Figas, M. Gruza, D. Puchalska, T. Kajdanowicz, and P. Kazienko, “Offensive, aggressive, and hate speech analysis: From data-centric to human-centered approach,” Information Processing & Management, vol. 58, no. 5, p. 102643, Sep. 2021, https://doi.org/10.1016/j.ipm.2021.102643.

[30] Z. Waseem and D. Hovy, “Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter.” in NAACL (pp 88-93), 2016.

[31] F. Poletto, V. Basile, M. Sanguinetti, C. Bosco, and V. Patti, “Resources and benchmark corpora for hate speech detection: A systematic review,” Language Resources and Evaluation, vol. 55, no. 2. pp. 477–523, Jun. 2021. https://doi.org/10.1007/s10579-020-09502-8.

[32] X. Yang, S. Obadinma, H. Zhao, Q. Zhang, S. Matwin, and X. Zhu, “SemEval-2020 task 5: Counterfactual recognition,” Aug. 2020, [Online]. Available: http://arxiv.org/abs/2008.00563.

[33] T. Ranasinghe, M. Zampieri, and H. Hettiarachchi, “BRUMS at HASOC 2019: Deep learning models for multilingual hate speech and offensive language identification.” [Online]. Available: https://github.com/TharinduDR/HASOC-2019.

[34] T. Mandl et al., “Overview of the HASOC track at FIRE 2020: Hate speech and offensive content identification in Indo-European languages” 2020, [Online]. Available: http://ceur-ws.org.

[35] T. Mandl et al., “Overview of the HASOC subtrack at FIRE 2021: Hate speech and offensive content identification in English and Indo-Aryan” 2021, [Online]. Available: http://ceur-ws.org.

[36] Md. R. Karim *et al*., “DeepHateExplainer: Explainable hate speech detection in under-resourced Bengali language,” in IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), Porto, Portugal, 2021, pp. 1–10 [Online]. https://doi.org/10.1109/DSAA53316.2021.9564230.

[37] Md. R. Karim, S. K. Dey, T. Islam, Md. Shajalal, and B. R. Chakravarthi, “Multimodal hate speech detection from Bengali memes and texts,” in SPELLL 2022, Springer, Cham, vol. 1802, pp. 293–308, 2022. [Online]. https://doi.org/10.1007/978-3-031-33231-9_21.

[38] H. Sohn and H. Lee, “MC-BERT4HATE: Hate speech detection using multi-channel bert for different languages and translations,” in IEEE International Conference on Data Mining Workshops (ICDMW), Beijing, China, pp. 551–559, 2019. [Online]. https://doi.org/10.1109/ICDMW.2019.00084.

[39] R. Alshalan and H. Al-Khalifa, “A deep learning approach for automatic hate speech detection in the Saudi Twittersphere,” Applied Sciences (Switzerland), vol. 10, no. 23, pp. 1–16, Dec. 2020. https://doi.org/10.3390/app10238614.

[40] P. Alonso, R. Saini, and G. Kovács, “Hate Speech Detection using Transformer Ensembles on the HASOC dataset,” in SPECOM 2020, Springer, Cham, vol. 12335, 2020, pp. 13–21 [Online]. https://doi.org/10.1007/978-3-030-60276-5_2.

[41] S. Khan *et al*., “BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection,” Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 7, pp. 4335–4344, Jul. 2022. https://doi.org/10.1016/j.jksuci.2022.05.006.

[42] Y. Bengio, I. Goodfellow, and A. Courville, Deep learning, USA, 2015.

[43] J. Salminen, M. Hopf, S. A. Chowdhury, S. gyo Jung, H. Almerekhi, and B. J. Jansen, “Developing an online hate classifier for multiple social media platforms,” Human-centric Computing and Information Sciences, vol. 10, no. 1, Dec. 2020. https://doi.org/10.1186/s13673-019-0205-6.

[44] S. Bose and G. Su, “Deep one-class hate speech detection model,” in Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France pp. 7040–7048, 2022.

[45] H. Mehta and K. Passi, “Social media hate speech detection using explainable artificial intelligence (XAI),” Algorithms, vol. 15, no. 8, p. 291, Aug. 2022. https://doi.org/10.3390/a15080291.

[46] S. Mohtaj, V. Schmitt, and S. Möller, “A feature extraction based model for hate speech identification,” in FIRE 2021 - Hate Speech and Offensive Content Detection (HASOC) Track, 2022, [Online]. Available: http://arxiv.org/abs/2201.04227.

[47] N. Vashistha and A. Zubiaga, “Online multilingual hate speech detection: Experimenting with Hindi and English social media,” Information (Switzerland), vol. 12, no. 1, p. 5, Jan. 2021. https://doi.org/10.3390/info12010005.

[48] Y. Zhao and X. Tao, “ZYJ123@DravidianLangTech-EACL2021: Offensive language identification based on XLM-RoBERTa with DPCNN,” in Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, pp 216–221,2021.

[49] M. F. Mridha, M. A. H. Wadud, M. A. Hamid, M. M. Monowar, M. Abdullah-Al-Wadud, and A. Alamri, "L-Boost: Identifying offensive texts from social media post in Bengali," IEEE Access, vol. 9, pp. 164681–164699, 2021. https://doi.org/10.1109/ACCESS.2021.3134154.

[50] L. Stappen, F. Brunn, and B. Schuller, "Cross-lingual zero- and few-shot hate speech detection utilising frozen transformer language models and AXEL," Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.13850.

[51] W. Yin and A. Zubiaga, "Towards generalisable hate speech detection: A review on obstacles and solutions," PeerJ Computer Science, vol. 7, pp. 1–38, 2021. https://doi.org/10.7717/peerj-cs.598.

[52] P. Fortuna, J. Soler-Company, and L. Wanner, "How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets?," Information Processing & Management, vol. 58, no. 3, p. 102524, May 2021. https://doi.org/10.1016/j.ipm.2021.102524.

[53] N. Seemann, Y. S. Lee, J. Höllig, and M. Geierhos, "Generalizability of abusive language detection models on homogeneous German datasets," Datenbank-Spektrum, vol. 23, no. 1, pp. 15–25, Mar. 2023. https://doi.org/10.1007/s13222-023-00438-1.

[54] S. Mishra, S. Prasad, and S. Mishra, "Exploring multi-task multi-lingual learning of transformer models for hate speech and offensive speech identification in social media," SN Computer Science, vol. 2, no. 2, Jan. 2021. https://doi.org/10.1007/s42979-021-00455-5.

[55] F. M. Plaza-Del-Arco, M. D. Molina-Gonzalez, L. A. Urena-Lopez, and M. T. Martin-Valdivia, "A multi-task learning approach to hate speech detection leveraging sentiment analysis," IEEE Access, vol. 9, pp. 112478–112489, 2021. https://doi.org/10.1109/ACCESS.2021.3103697.

[56] E. Lavergne, R. Saini, G. Kovács, and K. Murphy, "TheNorth @ HaSpeeDe 2: BERT-based language model fine-tuning for Italian hate speech detection," in 7th Evaluation Campaign of Natural Language Pro-cessing and Speech Tools for Italian. Final Workshop, EVALITA, 2020.

[57] H. Saleh, A. Alhothali, and K. Moria, "Detection of hate speech using BERT and hate speech word embedding with deep model," Applied Artificial Intelligence, vol. 37, no. 1, 2023. https://doi.org/10.1080/08839514.2023.2166719.

[58] G. L. De la Peña Sarracén and P. Rosso, "Systematic keyword and bias analyses in hate speech detection," Information Processing & Management, vol. 60, no. 5, p. 103433, Sep. 2023. https://doi.org/10.1016/j.ipm.2023.103433.

[59] P. Kazienko et al., "Human-centered neural reasoning for subjective content processing: Hate speech, emotions, and humor," Information Fusion, vol. 94, pp. 43–65, Jun. 2023. https://doi.org/10.1016/j.inffus.2023.01.010.

[60] B. Mathew, P. Saha, S. M. Yimam, C. Biemann, P. Goyal, and A. Mukherjee, "HateXplain: A benchmark dataset for explainable hate speech detection," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 17, 2021, pp. 14867–14875 [Online]. https://doi.org/10.1609/aaai.v35i17.17745.

[61] M. Bhatia et al., "One to rule them all: Towards joint Indic language hate speech detection," in HASOC-FIRE Shared Task on Hate Speech and Offensive Language Detection, 2021. Available: http://arxiv.org/abs/2109.13711.

[62] A. Velankar, H. Patil, A. Gore, S. Salunke, and R. Joshi, "L3Cube-MahaHate: A Tweet-based Marathi hate speech detection dataset and BERT models," in TRAC, Gyeongju, Republic of Korea, pp. 1–9, 2022. [Online]. Available: https://aclanthology.org/2022.trac-1.1.

[63] T. Caselli, V. Basile, J. Mitrović, and M. Granitzer, "HateBERT: Retraining BERT for abusive language detection in English," Oct. 2020, [Online]. Available: http://arxiv.org/abs/2010.12472.

[64] G. del Valle-Cano, L. Quijano-Sánchez, F. Liberatore, and J. Gómez, "SocialHaterBERT: A dichotomous approach for automatically detecting hate speech on Twitter through textual analysis and user profiles," Expert Systems with Applications, vol. 216, p. 119446, Apr. 2023. https://doi.org/10.1016/j.eswa.2022.119446.

[65] K. Perifanos and D. Goutsos, "Multimodal hate speech detection in greek social media," Multimodal Technologies and Interaction, vol. 5, no. 7, p. 34, Jul. 2021. https://doi.org/10.3390/mti5070034.

[66] K. Florio, V. Basile, M. Polignano, P. Basile, and V. Patti, "Time of your hate: The challenge of time in hate speech detection on social media," Applied Sciences (Switzerland), vol. 10, no. 12, p. 4180, Jun. 2020. https://doi.org/10.3390/app10124180.

[67] S. Dowlagar and R. Mamidi, "HASOCOne@FIRE-HASOC2020: Using BERT and multilingual BERT models for hate speech detection," 2020. [Online]. https://doi.org/10.48550/arXiv.2101.09007.

[68] T. Tita and A. Zubiaga, "Cross-lingual hate speech detection using transformer models," arXiv preprint, 2021, https://doi.org/10.48550/arXiv.2111.00981.

[69] I. Bigoulaeva, V. Hangya, I. Gurevych, and A. Fraser, "Label modification and bootstrapping for zero-shot cross-lingual hate speech detection," Language Resources and Evaluation,2023. https://doi.org/10.1007/s10579-023-09637-4.

[70] R. Nayak and R. Joshi, "Contextual hate speech detection in code mixed text using transformer based approaches," Oct. 2021, [Online]. Available: http://arxiv.org/abs/2110.09338.

[71] S. S. Aluru, B. Mathew, P. Saha, and A. Mukherjee, "Deep learning models for multilingual hate speech detection," in *ECML-PKDD*, 2020, [Online]. Available: http://arxiv.org/abs/2004.06465.

[72] A. Velankar, H. Patil, and R. Joshi, "Mono vs multilingual BERT for hate speech detection and text classification: A case study in Marathi," Apr. 2022, https://doi.org/10.1007/978-3-031-20650-4_10.

[73] S. Bakheet and A. Regina, “Hate and offensive speech detection on Arabic social media,” Online Social Networks and Media, vol. 19, p. 100096, 2020. https://doi.org/10.1016/j.osnem.2020.100096.

[74] M. Almaliki, A. M. Almars, I. Gad, and E. S. Atlam, “ABMM: Arabic BERT-mini model for hate-speech detection on social media,” Electronics (Switzerland), vol. 12, no. 4, p. 1048, Feb. 2023. https://doi.org/10.3390/electronics12041048.

[75] Z. Boulouard, M. Ouaissa, M. Ouaissa, M. Krichen, M. Almutiq, and K. Gasmi, “Detecting hateful and offensive speech in Arabic social media using transfer learning,” Applied Sciences (Switzerland), vol. 12, no. 24, p. 12823, Dec. 2022. https://doi.org/10.3390/app122412823.

[76] S. Benhur and K. Sivanraju, “Pretrained transformers for offensive language identification in Tanglish,” in FIRE 2021, Oct. 2021, [Online]. Available: http://arxiv.org/abs/2110.02852.

[77] A. Velankar, H. Patil, A. Gore, S. Salunke, and R. Joshi, “Hate and offensive speech detection in Hindi and Marathi,” in FIRE2021, Oct. 2021, [Online]. Available: http://arxiv.org/abs/2110.12200.

[78] M. Polignano, P. Basile, M. De Gemmis, and G. Semeraro, “Hate speech detection through AlBERTo Italian language understanding model,” 2019. [Online]. Available: https://github.com/marcopoli/AlBERTo-it.

[79] K. Miok, B. Škrlj, D. Zaharie, and M. Robnik-Šikonja, “To BAN or Not to BAN: Bayesian attention networks for reliable hate speech detection,” Cognitive Computation, vol. 14, no. 1, pp. 353–371, Jan. 2022. https://doi.org/10.1007/s12559-021-09826-9.

[80] S. Banerjee, M. Sarkar, N. Agrawal, P. Saha, and M. Das, “Exploring transformer based models to identify hate speech and offensive content in English and Indo-Aryan languages,” Nov. 2021, [Online]. Available: http://arxiv.org/abs/2111.13974.

[81] M. Bilal, A. Khan, S. Jan, S. Musa, and S. Ali, “Roman Urdu hate speech detection using transformer-based model for cyber security applications,” Sensors, vol. 23, no. 8, p. 3909, Apr. 2023. https://doi.org/10.3390/s23083909.

[82] 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. https://doi.org/10.1109/ACCESS.2020.3037073.