Federated-Learning Topic Modeling Based Text Classification Regarding Hate Speech During COVID-19 Pandemic

Muhammad Kamran[1], Ammar Saeed[2], Ahmed Almaghthawi[3]
Department of Cybersecurity, College of Computer Science and Engineering, University of Jeddah, 21959, Kingdom of Saudi Arabia[1]
Department of Computer Science, COMSATS University Islamabad, Wah Campus, WahCantt, Pakistan[2]
Department of Computer Science, College of Science and Art at Mahayil, King Khalid University, Abha 62529, Saudi Arabia[3]

Abstract—One of the most challenging tasks in knowledge discovery is extracting the semantics of the content regarding emotional context from the natural language text. The COVID-19 pandemic gave rise to many serious concerns and has led to several controversies including spreading of false news and hate speech. This paper particularly focuses on Islamophobia during the COVID-19. The widespread usage of social media platforms during the pandemic for spreading of false information about Muslims and their common religious practices has further fueled the existing problem of Islamophobia. In this respect, it becomes very important to distinguish between the genuine information and the Islamophobia related false information. Accordingly, the proposed technique in this paper extracts features from the textual content using approaches like Word2Vec and Global Vectors. Next, the text classification is performed using various machine learning and deep learning techniques. The performance comparison of various algorithms has also been reported. After experimental evaluation, it was found that the performance metric like F1-score indicate that Support Vector Machine performs better than other alternatives. Similarly, Convolutional Neural Network also achieved promising results.

Keywords—Knowledge extraction; text mining; pandemics and society; hate speech; Islamophobia

I. INTRODUCTION

One of the most challenging tasks in knowledge discovery is extracting the semantics of the content regarding emotional context from the natural language text. The COVID-19 pandemic gave rise to many serious concerns and has led to several controversies including spreading of false news and hate speech. Hate speech becomes more emotionally hurting if it targets someone's belief. In this paper, we particularly focus on Islamophobia which is a type of racism that is being practiced by anti-Muslim communities, individuals, groups, and organizations against Islam and Muslims [1]. It is one of the most visible forms of racism in the modern-day and several relevant incidents are reported on daily basis. but it is still not being given due attention and consideration as a global issue. The internet and social media are one of the primary means of disseminating fake news and false information around the world [2], [3], [4], [5]. Consequetenly, Muslim community is facing several challenges in their daily as well as professional life where they are in minority in different parts of the world [6]. Moreover, global Islamophobia has increased significantly because of COVID-19. On social media platforms, false information, hate speech, and conspiracy theories regarding Muslims have been circulated, further stigmatizing them. Also, the stigmatization of Muslims and others of Asian heritage has resulted from the pandemic's genesis in Wuhan, China. Discriminatory laws, such as the travel bans imposed by some nations on nations with most Muslims, have made the issue worse by feeding already-existing anti-Islamic attitudes. As a result, the epidemic has acted as a trigger for the escalation of Islamophobia, maintaining prejudice and unfavorable views towards Muslims. The role of social media usage during the pandemic has evidently played a major role in spreading Islamophobia [7], [8].

Conspiracy theories and false allegations about Muslims being to blame for the virus's spread have been propagated over social media during COVID-19. The spread of this misinformation on social media sites has fueled an upsurge in anti-Muslim sentiment. Muslims have been held responsible for the virus's spread on multiple occasions. For instance, after a religious gathering was conducted in New Delhi in March 2020, there were rumors of Muslims being held responsible for spreading the disease throughout India. Islamophobia increased as a result, with some calling the illness the "Muslim virus" or the "Tablighi virus" in India [9], [10].

Although some of the social media platforms implement the procedures for preventing the spread of hate speech; however, automation of such measures is still an ongoing research area. In this respect, the Natural Language Processing (NLP), Machine Learning (ML), and Deep Learning (DL) can assist developing such automated methods [11]. These ML and DL techniques are most widely used in the sentiment analysis of the textual content from social media platforms and have yielded excellent outcomes thus far. Moreover, they are several other applications of ML and DL techniques in various domains like mentioned in [12], [13] and [14]. Our goal is to use them for tackling the spread of hate speech such that results in Islamophobia.

In this paper, we focus on the identification and classification of Islamophobic content originated during the COVID-19 pandemic. First of all, we perform the data collection step followed by preprocessing of the data. For this, we extracted one dataset from the Google fact-checking

platform while another dataset was collected from the tweets of social media platform X (formerly known as Twitter). We also performed the analysis of data using approaches like) Bag of Words (BoW), Term Frequency – Inverse Document Frequency (TF-IDF), etc. For classification, we used: (i) ML techniques like, Support Vector Machine (SVM), Naïve Bayes (NB), Logistic Regression (LR), and Random Forest (RF); (ii) NLP transformer-based algorithms like Generative Pre-Trained Transformer (GPT) and Bidirectional Encoders Representation from Transformers (BERT); and (iii) DL models Long Short-Term Memory Network (LSTM) and Convolutional Neural Network (CNN). We also conducted a comparative study of these methods usingvarious performance measures. To summarize, the major contributions of the proposed work are:

- The extraction of various features from the textual data containing COVID-19 and Islamophobia related content.
- Data preprocessing for making it suitable for use for knowledge extraction using various ML and DL techniques.
- Classification of the data using various likes like ML, DL, BERT, and GPT.
- Evaluation of the performance of various classification techniques using performance metrics like Accuracy, Precision, Recall, F1-score, and AUC (area under the curve).

The rest of the paper bas been organized as follows. Section II provides the relevant details about the related work. The working of proposed approach has been described in Section III followed by its experimental evaluation in Section IV.Finally, Section V provides conclusion of the proposed work.

II. Related Work

Various studies have been conducted on Twitter datasets for detecting and removing hate speech related to COVID-19 and hate speech. Chandra et al. [15] used the coronaBias dataset containing 410,990 tweets related to COVID-19 and explored three different approaches for feature derivation and sentiment learning, including LDA, NMF, and Top2Vec. Mehmood et al. [16] worked on a dataset of 1290 tweets for hate speech detection and utilized 1D CNN with RNN for feature extraction and classification. Khan et al. [17] collected 8438 English and 8790 Hindi tweets from Twitter, and used Word2Vec, GloVe, BERT, and n-gram methods for the classification of tweets polarity classes. Alraddadi et al. [18] performed Arabic text classification using a dataset compiled using the Octoparse scrapping tool, and utilized ML algorithms such as KNN,SVM, LR, MNB, and NB for data classification. Vidgen and Yasseri [19] proposed a technique for classifying Islamophobic hate speech using KNN, SVM, LR, MNB, and CNN. To detect hate speech related to COVID-19 and Islamophobia, these studies utilized various approaches for feature extraction, classification, and sentiment learning. The datasets used in these studies were annotated and passed through various pre-processing steps such as case folding, tokenization, stop words removal, cleaning, and normalization. The ML algorithms used for classification included KNN, SVM, LR, MNB, and NBfor Arabic text classification, and DT, RF, LR, NB, SVM, and CNN for X datasets. The best results were obtained by using various combinations of ML algorithms with feature extraction methods such as GloVe, Word2Vec, n-gram and BERT methods. These studies provided fruitful results for detecting and removing hate speech from social media platforms, and their findings can be further utilized for developing efficient tools for hate speech detection.

In their work, Massey et al. [20] analyzed data from social platforms for the detection of Islamophobic content using machine learning and trend analysis approaches. The dataset, used in this work, was scraped using predetermined Islamic keywords and includes political opinions from the left, right, and center. ML techniques such NB, SVM, Boosting, MAXENT, CART, and RF were used by the researchers using 10-fold cross-validation to 400 hand-labeled comments. The accuracy of the Bagging and RF classifiers was practically identical at 0.66%, according to the data collected using multiple performance indicators, and stemming did not enhance the outcomes in this instance. In a further study, Gata and Bayhaqy [21] examined tweets concerning Islamophobia in the wake of the 2019 Christchurch assault in New Zealand. A dataset of 3115 collected tweets from March 15, 2019, the day of the incident, was used in the study. The dataset underwent various steps of preparation, including scraping, stop words elimination, and tokenization. The two ML models, NB and SVM, were combined with the random oversampling technique for result derivation and comparison. The best accuracy of 91.390% was provided by SVM with SMOTE, which was superior to other combinations. Ayan et al.'s [22] sentiment analysis of Twitter data was done to look for anti-Islamic content. From August to September 2018, the researchers gathered 162,000 tweets that had been manually positive and negative rated by professional annotators. To prepare the data for pre-processing by ML algorithms like Ridge Regression (RR) and NB, weblinks, converted letters, word-level TF-IDF, and redundancy removal were removed from the input. The Bayesian classifier took more time and had a lower accuracy of 98.1% than the RR classifier. In a study by F. González-Pizarro and S. Zannettou [23], nasty attitudes on political data from Papasavva were analyzed using contrastive learning. 134.5 million Political postings from June 2016 to November 2019 were included in the collection, coupled with a dataset of 5,859,439 photos from Zannettou. The data was pre-processed, and severe toxicity levels were calculated to identify and classify Islamophobic content. Another study [24] by Saha et al. looked at hate speech in Hindi and the rise in hate crimes in India. They made use of the 2019 HASOC dataset, which was made available to the public and included translations in English, German, and Hindi. The Gradient Boosting model, along with mBERT and LASER embeddings, was used to achieve language neutrality. Due to the unbalanced data, the model they constructed performed better on Hindi data than on English and German data.

In [25], 5,846 Lebanese and Syrian political tweets, categorized as normal, abusive, or hostile were used by Mulki et al. [25] to construct the L-HSAB dataset. An integration of

SVM and NB classifiers was used with n-gram BoW and TF-IDF vectorization methods. ML classifiers with n-gram vectorization frequently outperform neural networks for text classification. These strategies, however, are domain-specific and might not work well if the context of the information is removed or if negative remarks are given good connotations. Gitari et al. [26] provided a three-step methodology for classifying hate speech. A rule-based approach is utilized to determine the text's subject in the first stage followed bythe creation of a lexicon for hate speech. Finally, a text is deemed to be hate speech if it contains any of the three characteristics like hate verbs, negative polarity, and theme-based grammatical patterns. Despite being simple to understand, lexicon-based approaches are not totally reliable. A multi-class classifier was used by Davidson et al. [27] to distinguish between political correctness, offensive language, and hate speech. They created a precise model with L2 regularization using LR, and the results were encouraging. By merging different techniques, hybrid approaches have also been employed to detect hate speech. For the classification of hate speech, Wester et al. [28] have presented a hybrid technique that blends learning and lexical-based methods. Using a lexicon-based technique, complicated syntactic and semantic aspects are extracted using this method, and a learning algorithm is then used. In comparison to the distinct lexical and learning approaches, the hybrid model has performed better. Although, the work in [28] address hate speech but a relevant problem that needs is Interest in the detection of Islamophobic textual content has increased because of the rise is: Islamophobic occurrences during COVID-19. However, because to the dearth of publicly accessible datasets and the sparse application of numerous textual features and transformer-based core NLP approaches, there has been little research in this field. There is a huge research gap because of the majority of studies concentrating on either traditional textual features or word embeddings with ML and DL models. For this, in this work, we attempt to overcome these research issues and construct an efficient model for accurate Islamophobic content identification in the proposed work.

III. Proposed Work

This section will delve into the detailed discussion of the proposed framework along with justification of adopted methods.

A. Proposed Fremwork

Here, the presented framework will be demonstrated and discussed in detail. Fig. 1 provides a compact overview of proposed model.

For classifying Islamophobia related social media content,first, datasets were collected from X and Google fact-checking API. This step is followed by data preprocessing using various techniques such as stop words removal, data balancing, lemmatization, and tokenization. Additionally, for feature extraction through word embeddings and n-grams,different methods like Word2Vec, GloVe, TF-IDF, and BoW are used. For classification, transformer-based techniques BERT and GPT, and topic modelingare utilized. After that, the classification is performed using some selected conventional ML, DL, and transformed-based techniques. Performance evaluation measures Accuracy, Precision, Recall, F-Measure, and AUC are recorded evaluating the performance of these classifiers.

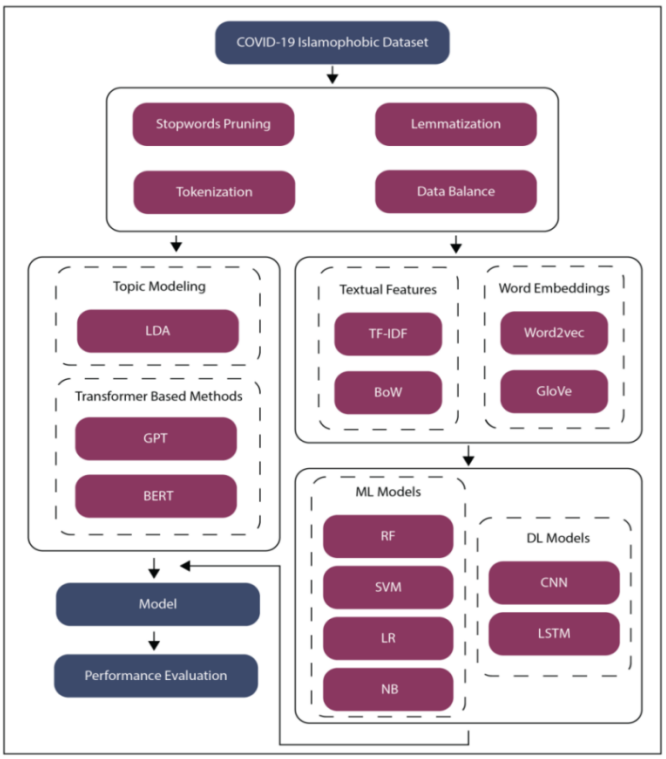

Fig. 1. Proposed model framework.

B. Feature Extraction

This phase involves extracting features that are particularly useful in experiments for achieving desirable outcomes. We present the detail of this phase in the following.

1) TF-IDF: TF-IDFcalculates the frequency of words in each document by taking into the account the inverse frequency of such words appearing in multiple documents consistently [29]. The weight of each document in the corpus can be computed using Eq. (1):

$$wgt_{d,c} = freq^{t}_{d,c} x\ log\ log(\frac{N}{freq_d} \quad (1)$$

Where, $wgt_{d,c}$ is denoted as the total weightage of both data points, $freq^{t}_{d,c}$ computes the frequency of occurrences of the data point d in c. N represents total number of documents in the corpus, $log\ log(\frac{N}{freq_d})$ computes the log of all the documents present in the corpus with the frequency of data point d.

2) BoW: To extract useful features from textual data for classification purposes, the Bag of Words (BoW) [30] method is employed. This approach considers a document or phrase as a set of its constituent words and checks for the presence of familiar words irrespective of their order. BoW generates word bags using Eq. (2), as follows:

$$doc_c = \sum_{d=1}^{N} \quad weight^{c}_{d} x weight_d \quad (2)$$

Where doc_c denotes the documents that contain the concerned data point d. $weight_d^c$ are the scalar weights of the frequent word d for the data point c in the document. While $weight_d$ indicates the weight of frequent word d.

3) Word2vec: The Word2Vec approach uses a three-layer deep neural network to analyze the context of a document and connect related phrases. Unlike BoW, Word2Vec offers two models - Continuous Bag of Words (CBoW) and Skip-Gram [31]. To ensure proper word embedding, it is recommended that Word2Vec is trained on a large and high-quality dataset. The computation of Word2Vec through the Skip-Gram method for an M-dimensional data corpus contain a word wd_o at location q can be seen in Eq. (3).

$$\frac{1}{T}\sum_{q=1}^{M} \sum_{-s\le a\le s, a\neq 0} \log \text{logprob}(\text{wd}_{q+1}|\text{wd}_o) \quad (3)$$

Where $log\ logprob(wd_{q+1}|wd_o)$ denotes the logarithm of wd_o with respect to placements and co-occurrences within the document.

4) GIoVE: To perform unsupervised learning, GloVe generates word embedding by constructing a count-based matrix based on word co-occurrence and analyzing each term individually [32]. It uses a less-weight approach to produce factors and creates a lower-dimensional matrix. The entire working logic of GloVe can be seen in Eq. (4).

$$\text{h} = \sum_{c,d=1}^{m} \text{g}(\text{f}_{c,d})(\text{d}_b^t \text{d}_a - \log \text{logf}_{c,d})^2 \quad (4)$$

5) Transformer-based models: Language models in NLP are built on transformers, which consist of an Encoder and Decoder. In this work, we used two transformer-based models GPT and BERT. GPT is an autoregressive decoders model, and it has two versions: GPT-2 and GPT-3 while BERT is a variant of bidirectional encoders-based models, and there are several types of BERT models. BERT uses Mask Language Modeling to overcomes the unidirectional constraint by utilizing and an attention mechanism and utilizes an encoder, decoder, and various layers. On the other hand, GPT is an autoregressive decoder model that works to benefit from unlabeled text datasets for using them on limited supervised datasets. GPT has two variants: GPT2 and GPT3, with the latter having 175 billion parameters and the capability to perform several NLP tasks such as text classification, question answering, text generation, and named entity recognition. Overall, these transformer-based models have revolutionized NLP tasks and continue to provide state-of-the-art performance.

C. Topic Modeling

To identify topics from a collection of documents, topic modeling is a useful technique. LDA is an effective approach for text classification in which the text of a document is classified based on its relation to a particular topic. The fundamental principle of LDA's functioning is demonstrated by Eq. (5).

$$\text{p}(\text{s}, \text{k}) = \text{q}(\text{k}|\text{d}) * \text{q}(\text{s}|\text{k}) \quad (5)$$

Where $q(k, d)$ is the probability of the topic per document and $q(s, k)$ is the probability of words per topic equaling the $p(s, k)$ denoted as the probability of word with the topic.

IV. Experiments and Results

To evaluate the performance of the proposed scheme, the experiments were conducted in systematic manner. In the first set of experiments, the n-gram method was used to extract features and classify the data using four ML algorithms. In the second set of experiments, word embedding features were classified using deep LSTM and CNN models. Next, LDA was applied to the data for topic modeling, and the classification step was performed. Finally, core NLP transformer-based methods, namely BERT and GPT, were evaluated. The dataset wqs balanced before conducting experiments.

A. Datasets

To investigate the global prevalence and impact of Islamophobia, we collected two distinct datasets. The first was obtained from the Google Fact Check platform and consisted of news articles that were fact-checked by websites such as PolitiFact and Snopes. We extracted articles relevant to Islam, including those with terms like Islam, Muslims, Quran, Jihad, and women, resulting in a total of 1555 articles. The second dataset was sourced from Twitter and included posts from users worldwide. We used predetermined hashtags, including #fuckIslam, #Jihadi, #Coronajihad, #Tablighijamat, and #TablighiJamaatVirus, as well as lexicons from Hatebase, to collect tweets from January 2020 to August 2020. The dataset is diverse as it retrieves data using an unbiased mechanism. The English-language dataset consists of 9612 tweets and was pre-annotated by three English-proficient annotators. During the annotation process, the annotators were not provided with any information about users' identities. The annotators were tasked with categorizing each tweet into one of three categories: Islamophobic, related to Islam but not Islamophobic, or neither about Islam nor Islamophobic. The annotations were assigned with great care, and in cases of disagreement, a majority vote was utilized. Of the 2930 tweets marked as Islamophobic, 4336 were related to Islam but not Islamophobic, and 2346 were neither Islamophobic nor related to Islam.

B. Dataset Preprocessing and Balancing

In the proposed work, various pre-processing techniques were applied, including converting all letters to lowercase, removing stop words and hyperlinks, and half-sentences. It also involves lemmatization, and tokenization. For data balancing, it is made sure in this phase that the balanced data is used for experiments and result analysis.

After performing pre-processing and balancing the dataset, the vocabulary size was determined for the English data. The vocabulary size for unigrams was found to be 17861 with an average tweet length of 14 words. After pre-processing the data to contain 8 words per tweet, the vocabulary size decreased to 16580 unigrams. Table I presents some of the most frequent words extracted from the dataset as part of feature extraction.

TABLE I. WORDS FRQUENCY IN DATASET AFTER PRE-PROCESSING

Sr. No	Words
1	Muslim
2	Islam
3	Islamic
4	Quran
5	Pakistan
6	Allah
7	Radical
8	Jehadi
9	Mohammed
10	Hindu

It is important to mention here that the utilized dataset had an imbalanced distribution, which was addressed using under-sampling. Tokenization and lemmatization were then applied to the pre-processed dataset, and during tokenization, both unigrams and bigrams were used, and the LDA model was employed to identify the best topics, which were visualized using an Intertypic Distance Map. The top 20 phrases from the first topic are visualized in Fig. 2 which account for 12.6% of the tokens. The bigram themes in the bar chart are distinguished from one another using an underscore. The use of these techniques can help identify the most pertinent phrases and topics in large datasets, making it easier to analyze and understand the data.

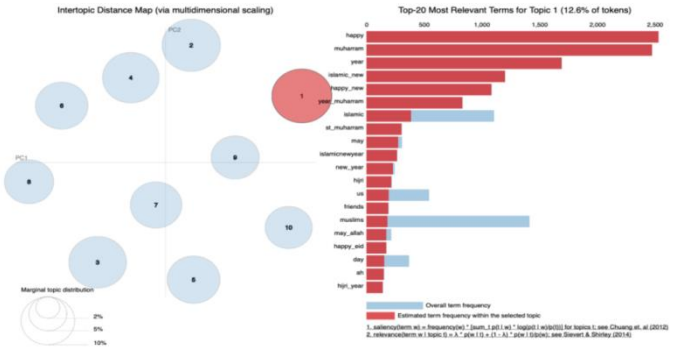

Fig. 2. Visualizing the most salient terms of first topic using topic modeling.

C. Machine Learning Algorithm with Textual Features

In initial experiment, we evaluated the n-grams based features using SVM. Tables II and III present the performance of various ML classifiers with BoW and TF-IDF, respectively, while maintaining the performance standards mentioned earlier. In the proposed work, SVM with n-gram based textual feature extraction techniques were applied to the categorical Islamophobia data in Python language. A train/test ration of 90/10 was used. The SVM model combined with the BoW method achieved a slightly higher accuracy of 91.7% compared to other counter parts.

In the next experiment, the RF classifier is used to detect Islamophobic content based on the same n-gram features as before. RF-BoW achieves significantly higher accuracy than RF-TF-IDF. The following experiment uses LR classifier for categorical data classification. LR-BoW outperforms LR-TF-IDF. In the last experiment, GNB is used for classification with the same features as before. TF-IDF was observed to achieve better results than BoW.

TABLE II. RESULTS OF VARIOUS ML MODELS WHILE USING TF-IDF

TF – IDF			
Algorithm	**Accuracy (%)**	**F1 Score (%)**	**AUC (%)**
RF	86.7	87.0	97.3
SVM	90.5	91.0	97.8
LR	90.3	90.0	97.8
NB	86.9	87.0	90.2

TABLE III. RESULTS OF ML MODELS WITH BOW

BoW			
Algorithm	**Accuracy (%)**	**F1 Score (%)**	**AUC (%)**
RF	87.6	88.0	97.0
SVM	91.7	92.0	98.0
LR	91.6	92.0	98.5
NB	77.4	77.0	82.6

D. Word Embeddings with Deep Learning Algorithms

We examined the performance of four ML models that used derived n-gram features and then explored the effectiveness of DL models with word embeddings as input. We experimented with a customized CNN, which is a type of deep neural network designed for rapid classification of vectorial data, using features extracted from the GloVe and Word2Vec word embedding models. We trained and tested the CNN model using the same data split as the ML algorithms, first with Word2Vec features using 32 epochs and a batch size of 10 and then with GloVe features using 100 epochs and a batch size of 32. For validation, the batch size remained the same while the number of epochs was set to 5. The results of both embedding models with CNN showed that CNN performs marginally better with GloVe than Word2Vec, exhibiting better accuracy and evaluation rates. The next experiment involves LSTM, which uses a batch size of 10, 20 epochs, and essential layers, including embedding, dense, and SoftMax layers. The accuracy of the LSTM model improves over time for both GloVe and Word2Vec features with a decrease in the loss ratio as the number of epochs increases. It was observed that the results of Word2Vec Features with LSTM were better than results with GloVe with Accuracy =88.6%, Precision, recall, and F1-score=89%, and AUC=97.2%.

The results indicate that Word2Vec embeddings provide better representation of the text data for the LSTM model. These findings are consistent with previous research that suggests that the choice of word embeddings can significantly impact the performance of deep learning models in natural language processing tasks. Therefore, selecting the appropriate word embeddings is crucial for the effectiveness of the model.

In another set of experiments, we test various machine learning algorithm with topic modeling which includes the experiments conducted using the LDA algorithm and ML

models. The derived topics are then scaled using a standard scalar before being classified. The selection of these topics is based on their grammar weightage, which helps identify the most relevant and significant terms for each topic.

Next experiment involved extracting unigrams and bigrams from the pre-processed dataset, which were then used as input for LDA. The LDA algorithm generated extracted topics after fine-tuning of the model. Again, four ML classifiers were then evaluated on the selected topics using the same split of 90/10 for training and testing sets respectively. The results of this experiment have been presented in Fig. 3.

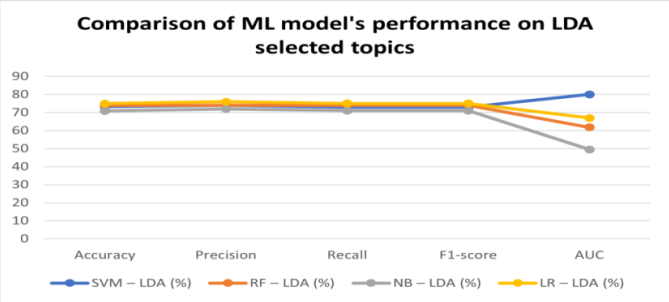

Fig. 3. Comparison of ML model's performance with LDA selected topics.

E. Using Transformer-based Techniques for Classification

As mentioned earlier, we also investigated the use of two popular transformer-based models, BERT and GPT, for the NLP task. BERT model is fed with the pre-processed dataset as input, which is then encoded into an embedding representation. After performing several transformations on the embeddings, the representations are decoded back into vocabulary-based representations. From the results, it was observed that GPT outperforms BERT, achieving an accuracy of 91.6% compared to BERT accuracy of 89%%. Similarly, the precision, recall, and f1-score values also show a similar trend, where GPT outperforms BERT. These results indicate that GPT is more effective in extracting contextual features and capturing the nuances of the text data for classification tasks.

The performance of the transformer-based NLP models, BERT and GPT, was also evaluated. The pre-processed dataset is fed into BERT and classification results are recorded. Fig. 4 shows the results of this experiment. The BERT model outperforms other models with a significantly higher accuracy of 89.31%. The results of this experiment demonstrate the effectiveness of BERT in text classification tasks and highlight its potential as a powerful NLP tool.

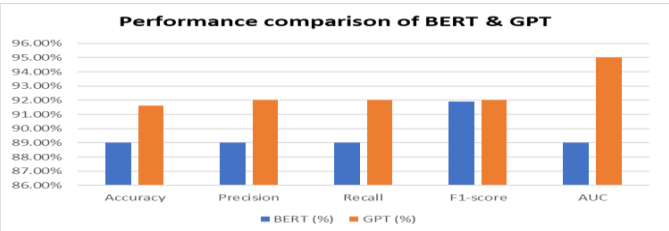

Fig. 4. Comparison of BERT and GPT based on various performance measures.

In the final part of the experiment, GPT-2 was used for text classification. The results of this experiment indicated that the GPT-2 model achieved an accuracy of 91.6%, which is significantly higher than the accuracies by other models.

F. Comparison of Results among Applied Techniques

From the results reported in the previous section, we got the motivation for comparing the results of various techniques. The results of the first experiment showed that BoW outperformed other techniques, particularly when TF-IDF is used with GNB models. These results suggest that BoW-based features are better suited for use during classification. During the classification, SVM showed the best performance achieving an accuracy of 90.7%. We believe this is because SVM is able to get good parameter settings without parameter tuning. Next, two DL models, LSTM and CNN, were compared using GloVe and Word2Vec. The results indicated that the custom CNN model outperformed LSTM in when evaluated against various performance metrics. This experiment highlights the importance of selecting the appropriate DL model and word embedding for achieving optimal performance in natural language processing tasks. It is worth noting that the performance of the ML and DL models can vary depending on the specific task and dataset. Therefore, it is crucial to conduct comprehensive experiments and compare the results before selecting the optimal model for a particular task. Fig. 5 shows the comparison of performance of CNN while using Word2vec and GloVe.

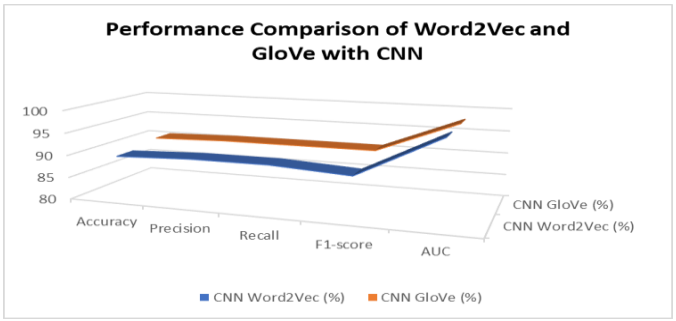

Fig. 5. Comparison of performance of Word2Vec and GloVe while using CNN.

In the second experiment, the LSTM model was used to classify Islamophobic content using the same word embedding models, Word2Vec and GloVe. The results showed that the LSTM-Word2Vec model had decent performance when compared to GloVe. In contrast to the CNN model, where the combination of CNN and Word2Vec performed better, the LSTM model with Word2Vec had better results. This comparison is also presented in Fig. 6. The comparison of different models and feature extraction techniques is important to determine the best approach for a given task. In this study, it was found that BoW-based features performed better than TF-IDF-based features when used with ML models, while CNN-GloVe outperformed LSTM-Word2Vec in DL models for classifying Islamophobic content. These findings can be useful in future studies and real-world applications for detecting and addressing hate speech online.

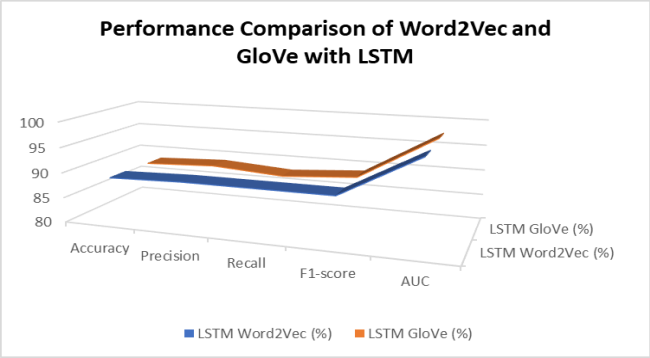

Fig. 6. Comparison of performance of Word2Vec and GloVe while using LSTM.

The second DL model, CNN, achieved the highest accuracy of 90.6% in the second experiment. LDA topic modeling combined with ML classifiers showed that LR and RF performed best among the classifiers. LR achieved the highest accuracy of 74.9%. The last experiment includes transformer-based models BERT and GPT, with BERT achieving an accuracy of 89.31%, which is in between the maximum accuracies of ML and DL models. Fig. 7 shows the comparison of BERT results with ML's best performing algorithm: SVM, DL models, and LDA's best performing model LR.

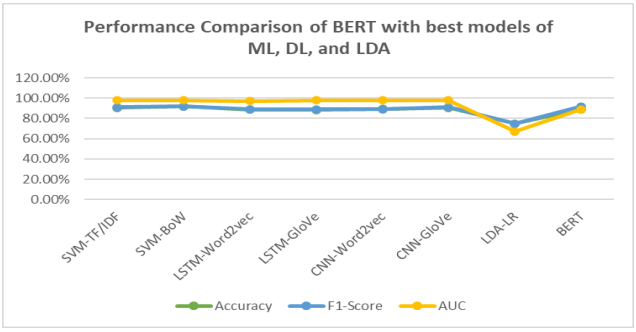

Fig. 7. Performance comparison of BERT with other classifiers.

In the next phase of the experiment, GPT-2 was utilized for the classification of Islamophobic content. The results indicate that GPT-2 outperformed all the previous methods used in this study. The F1 score achieved by GPT-2 was 92%, which is significantly higher than the other models. The comparison of GPT-2 results with other best-performing models can be visualized in Fig. 8. These findings highlight the effectiveness of GPT-2 in the classification of Islamophobic content and suggest that it could be used in similar tasks.

G. Comparison of Proposed Technique with Existing Islamophic Classification Techniques

In this section, we compare the results of the proposed study with those of prior art. It should be noted that previous studies did not use both textual features and word embeddings to test the performance of classification models. Nevertheless, we have compared their results with those achieved in the proposed study. Table IV presents a comparison of the F1 score results obtained by previous studies and the proposed study for Islamophobic content detection. It is evident from this table that the proposed study achieved better results than the previous studies, indicating that utilizing both textual features and word embeddings is an effective approach for improving the performance of classification models in this domain.

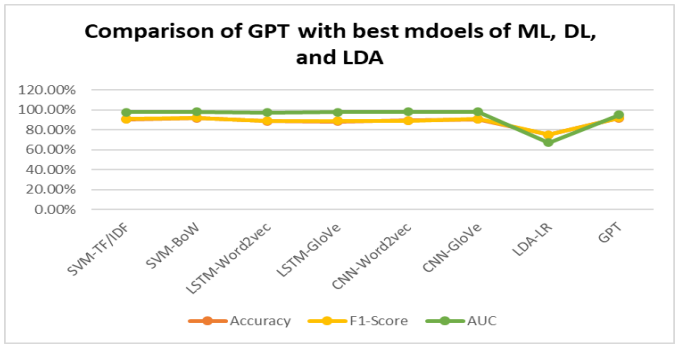

Fig. 8. Comparison of GPT with other classifiers.

TABLE IV. PERFORMANCE COMPARISON OF PROPOSED SCHEME WITH PRIOR ART

Ref	Algorithm	Results
Chandra et al. [15]	BERT	F1 score = 88.0
Mehmood et al. [16]	CNN	F1 score = 90.1
Alraddadi et al. [18]	NB	F1 score = 89.0
Vidgen et al. [19]	SVM	F1 score = 77.3
Massey et al. [20]	RF	F1 score = 66.0
Proposed Model	CNN, RF-LDA BERT, RF, SVM, GPT	CNN – F1 score = 91.0 RF – F1 score = 88.0 BERT – F1 score = 91.9 SVM – F1 score = 92.0 GPT – F1 score = 92.0

Table IV demonstrates that the proposed study outperformed the earlier investigations, even though the earlier studies did not make use of various Transformer technique variations or DL techniques with various word embeddings.

The main limitations of the proposed work, as noted by the findings of the experimental evaluation, are that more data is required to get better result that needs to be improved and can be a potential research area. Similarly, knowledge extraction from other hate speech related content related to any pandemic and its potential impact on the society can also to be investigated by extension of the proposed work.

V. CONCLUSION

Globally, there has been a substantial increase in content that is anti-Islamic because of the COVID-19 pandemic. Rapidly proliferating erroneous information and narratives have influenced negative attitudes and actions. Automated methods based on data science and AI, however, have emerged as useful resources for identifying and classifying racist content, enabling the detection and avoidance of damaging narratives. The proposed classifier performance evaluation in this study extracted significant features from the data using processes like Word2Vec, GloVe, etc. In addition, important themes were identified using topic modeling using LDA. Several ML and DL methods, such as LSTM and CNN with word embeddings and transformer-based models like BERT and GPT, were tested in this study. With an F1 score of 92%,

GPT was found to be the model that performed the best. Future studies might concentrate on employing various GPT iterations, such as GPT-3, and investigating additional DL models, such as RNN and GANs. Additionally, expanding the dataset can enhance the precision of the findings. Society may lessen Islamophobia and foster greater acceptance and tolerance by using these tools. To find bad content, stop it from spreading, and encourage a more open and tolerant society, it is essential to keep developing and improving automated tools.

ACKNOWLEDGMENT

This work was funded by the University of Jeddah, Jeddah, Saudi Arabia, under Grant No. (UJ-22-DR-82). The authors, therefore, acknowledge with thanks the University of Jeddah technical and financial support.

REFERENCES

[1] L. Cervi, S. Tejedor, and M. Gracia, ‘What Kind of Islamophobia? Representation of Muslims and Islam in Italian and Spanish Media’, Religions, vol. 12, no. 6, p. 427, 2021.

[2] R. A. Alraddadi and M. I. E.-K. Ghembaza, “Anti-Islamic Arabic Text Categorization using Text Mining and Sentiment Analysis Techniques,” International Journal of Advanced Computer Science and Applications (IJACSA), vol. 12, no. 8, 2021.

[3] Moreno-Vallejo, Patricio Xavier, Gisel Katerine Bastidas-Guacho, Patricio Rene Moreno-Costales, and Jefferson Jose Chariguaman-Cuji. "Fake News Classification Web Service for Spanish News by using Artificial Neural Networks," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 14, no. 3, pp. 301–306, 2023.

[4] BAYRAKLI, Enes, and Farid HAFEZ. "European Islamophobia Report (EIR) 2022." Medya ve Din AraştırmalarıDergisi 6, no. 1, pp. 221-225, 2023.

[5] J. Qian, ‘Historical Ethnic Conflicts and the Rise of Islamophobia in Modern China’, Ethnopolitics, pp. 1–26, 2021.

[6] Sukabdi, Zora Arfina, Muhammad Adlin Sila, Chandra Yudistira Purnama, FathulLubabinNuqul, Seta AriawuriWicaksana, Ali Abdullah Wibisono, and Yanwar Arief. "Islamophobia Among Muslims in Indonesia." Cogent Social Sciences, vol. 9, no. 1pp. pp. 1-29, 2023.

[7] Riaz, Marwa, Khadija Shahbaz, and Maryam Ali. "Islamophobia in the US and Europe: An Analytical Study." Annals of Human and Social Sciences, vol. 4, no. 2, pp. 615-625, 2023.

[8] T. Mirrlees and T. Ibaid, ‘The Virtual Killing of Muslims: Digital War Games, Islamophobia, and the Global War on Terror’, Islam. Stud. J., vol. 6, no. 1, pp. 33–51, 2021.

[9] Ahuja, K.K. and Banerjee, D.. The “labeled” side of COVID-19 in India: Psychosocial perspectives on Islamophobia during the pandemic. Frontiers in Psychiatry, vol. 11, p.604949, 2021.

[10] Rajan, B. and Venkatraman, S., Insta-hate: An Exploration of Islamophobia and Right-wing Nationalism on Instagram Amidst the COVID-19 Pandemic in India. Journal of Arab & Muslim Media Research, vol. 14, no.1, pp.71-91, 2021.

[11] Alghamdi, Jawaher, Yuqing Lin, and Suhuai Luo. "A Comparative Study of Machine Learning and Deep Learning Techniques for Fake News Detection." Information, vol. 13, no. 12, pp. 1-28, 2022.

[12] Ouassil, Mohamed-Amine, Bouchaib Cherradi, Soufiane Hamida, Mouaad Errami, Oussama EL Gannour, and Abdelhadi Raihani. "A Fake News Detection System based on Combination of Word Embedded Techniques and Hybrid Deep Learning Model." International Journal of Advanced Computer Science and Applications (IJACSA), vol. 13, no. 10, pp. 525–534, 2022.

[13] Kamran, Muhammad, and Ahmed Abdul-Aziz Almaghthawi. "Case-Based Reasoning Diagnostic System for Antenatal Research Database." International Journal of Online & Biomedical Engineering, vol. 18, no. 7, pp. 176-187, 2022.

[14] Alam, Furqan, Ahmed Almaghthawi, Iyad Katib, AiiadAlbeshri, and Rashid Mehmood. "IResponse: An AI and IoT-enabled Framework for Autonomous COVID-19 Pandemic Management." Sustainability, vol. 13, no. 7, pp. 3797, 2021.

[15] Chandra, Mohit, Manvith Reddy, Shradha Sehgal, Saurabh Gupta, Arun Balaji Buduru, and Ponnurangam Kumaraguru. "" A Virus Has No Religion": Analyzing Islamophobia on Twitter During the COVID-19 Outbreak." In Proceedings of the 32nd ACM conference on hypertext and social media, pp. 67-77, 2021.

[16] Mehmood, Qasim, Anum Kaleem, and Imran Siddiqi. "Islamophobic Hate Speech Detection from Electronic Media Using Deep Learning." In Mediterranean conference on pattern recognition and artificial intelligence, pp. 187-200, 2021.

[17] Khan, Heena, and Joshua L. Phillips. "Language agnostic model: Detecting Islamophobic content on social media." In Proceedings of the 2021 ACM Southeast conference, pp. 229-233. 2021.

[18] R. A. Alraddadi and M. I. E.-K. Ghembaza, ‘Anti-Islamic Arabic Text Categorization using Text Mining and Sentiment Analysis Techniques’, Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 8, 2021.

[19] B. Vidgen and T. Yasseri, ‘Detecting weak and strong Islamophobic hate speech on social media’, J. Inf. Technol. Polit., vol. 17, no. 1, pp. 66–78, 2020.

[20] T. Massey, C. Amrit, and G. C. van Capelleveen, ‘Analysing the trend of Islamophobia in Blog Communities using Machine Learning and Trend Analysis’, presented at the 28th European Conference on Information Systems, ECIS 2020: Liberty, Equality, and Fraternity in a Digitizing World, pp.1–14, 2020.

[21] W. Gata and A. Bayhaqy, ‘Analysis sentiment about islamophobia when Christchurch attack on social media’, TELKOMNIKA Telecommun. Comput. Electron. Control, vol. 18, no. 4, pp. 1819–1827, 2020.

[22] B. Ayan, B. Kuyumcu, and B. Ciylan, ‘Detection of Islamophobic Tweets on Twitter Using Sentiment Analysis’, Gazi Univ. J. Sci. Part C, vol. 7, no. 2, pp. 495–502, 2019.

[23] F. González-Pizarro and S. Zannettou, ‘Understanding and Detecting Hateful Content using Contrastive Learning’, ArXivPrepr. ArXiv220108387, 2022.

[24] P. Saha, B. Mathew, P. Goyal, and A. Mukherjee, ‘Hatemonitors: Language agnostic abuse detection in social media’, ArXivPrepr. ArXiv190912642, 2019.

[25] Mulki, Hala, Hatem Haddad, Chedi Bechikh Ali, and Halima Alshabani. "L-hsab: A levantine twitter dataset for hate speech and abusive language." In Proceedings of the third workshop on abusive language online, pp. 111-118, 2019.

[26] N. D. Gitari, Z. Zuping, H. Damien, and J. Long, ‘A lexicon-based approach for hate speech detection’, Int. J. Multimed. Ubiquitous Eng., vol. 10, no. 4, pp. 215–230, 2015.

[27] Davidson, Thomas, Dana Warmsley, Michael Macy, and Ingmar Weber. "Automated hate speech detection and the problem of offensive language." In Proceedings of the international AAAI conference on web and social media, vol. 11, no. 1, pp. 512-515, 2017.

[28] Wester, Aksel, Lilja Øvrelid, Erik Velldal, and Hugo Lewi Hammer. "Threat detection in online discussions." In Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 66-71. 2016.

[29] Kumar, Vipin, and Basant Subba. "A TfidfVectorizer and SVM based sentiment analysis framework for text data corpus." In 2020 national conference on communications (NCC), pp. 1-6. IEEE, 2020.

[30] Y. HaCohen-Kerner, D. Miller, and Y. Yigal, ‘The influence of preprocessing on text classification using a bag-of-words representation’, PloS One, vol. 15, no. 5, p. e0232525, 2020.

[31] K. W. Church, ‘Word2Vec’, Nat. Lang. Eng., vol. 23, no. 1, pp. 155–162, 2017.

[32] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp. 1532-1543.