Islamophobic Hate Speech Detection from Electronic Media Using Deep Learning

Qasim Mehmood(✉), Anum Kaleem, and Imran Siddiqi

Department of Computer Science, Bahria University, Islamabad 44000, Pakistan
01-249182-018@student.bahria.edu.pk,
{anumkleem.buic,imran.siddiqi}@bahria.edu.pk

Abstract. Islamophobic hate speech is the indiscriminate negative attitude and behavior towards Muslims and Islam. Speech indicating prejudice against Muslims has negatively impacted the perceptions of Islam. Online platforms like Twitter have carved out policies to stop users from promoting Islamophobic hate speech, however, such content still exists which causes problems for Muslim communities globally. Hence, it becomes pivotal to find solutions to eradicate such speech from social media platforms. This paper presents an effective methodology for Islamophobic hate speech identification in online tweets using deep learning techniques. The proposed technique relies on feature extraction using a one-dimensional Convolutional Neural Network and classification using Long Short-Term Memory network based classifier. The proposed technique is validated on a dataset comprising of 1290 pre-processed online tweets and an accuracy of more than 90% is reported.

Keywords: Hate speech · Islamophobia · Word embeddings · Convolution Neural Networks (CNN) · Bi-directional Long Short-Term Memory (LSTM)

1 Introduction

The history of hateful content on print and electronic media is spanned over many decades. Due to rapid growth of the Internet and the availability of low-cost devices, the number of social and electronic media users has increased tremendously in the recent past. A downside of this growth is the increase in conflicts, hate speech, cyber trolling and bullying. In this domain, several research studies have been carried out on recognition of hate speech including gender, racism, religion, color, disability and citizenship. Among these, the focus of our current research lies on hate speech identification and more specifically the Islamophobic hate speech. Islamophobia composed of the term 'Islam' with the postfix 'phobia', refers to the 'fear of Islam'. Multiple mediums of expression like text, audio, images or videos are commonly exploited to promote hate speech by online users.

C. Djeddi et al. (Eds.): MedPRAI 2021, CCIS 1543, pp. 187–200, 2022.
https://doi.org/10.1007/978-3-031-04112-9_14

Islamophobic content has resulted in biasness, discrimination, and exclusion of Muslims in societies from social, civic, and political life [1].

Hate speech in the form of tweets, posts, or articles has caused problems for inhabitants of Muslim communities living in the western countries, especially after the 9/11 attacks. According to the Runnymede trust in the United Kingdom, Islamophobia existed in premise before the September 11, 2001 incident, but after these terror attacks it has increased significantly. The Runnymede trust, also identified eight components of Islamophobia in a report published in 1997. A followup report was produced in 2004, which deduced that the aftermath of the terror attacks had made life more difficult for Muslims in the United Kingdom and other countries [2]. Moreover, the report also stated that it is almost impossible to stop the domino of Islamophobic statements from spreading over social media [3]. Hence, there is a strong need to develop tools and techniques to identify and classify derogatory hate speech against Muslims at large.

Though social media platforms like YouTube, Twitter and Facebook have established usage policies that forbid hate speech [4–6], still they fail to eradicate such content completely. It is therefore important to develop solutions that can automatically identify hate speech and suggest the required measures. Several research studies have been carried out on recognition of classical hate speech including gender [7–9] racism [10–12] and religion [13–15]. The literature is relatively limited when it comes to Islamophobic hate speech identification [16].

This paper presents an effective method for identification of Islamophobic hate speech from online tweets. The technique relies on converting the pre-processed tweets to word embeddings which are subsequently fed to one-dimensional convolutions. A bi-directional Long Short-Term Memory network (LSTM) is then employed for classification. The key highlights of this study are outlined in the following.

- An effective technique for identification of Islamophobic hate speech from online tweets is presented.
- A combination of 1D convolutions with Recurrent Neural Networks (RNN) is employed for feature learning and classification.
- A comprehensive experimental study is carried out using different variants of RNNs and the reported results validate the effectiveness of the proposed method.

The content of this paper is divided into five sections. Section 2 presents an overview of the relevant literature primarily focusing on the recent trends in this domain. Section 3 introduces the dataset, pre-processing and the details of the proposed methodology. Experimental results and the related discussion are detailed in Sect. 4 while Sect. 5 concludes this paper with a recall of our key findings.

2 Related Work

This section discusses some notable contributions to hate speech identification using pattern classification techniques. Formally, hate speech is defined as the *‘negative speech against a person or members of groups identified by protected characteristics that express the speaker’s emotions or feelings’* [17]. In general, the social media platforms (like Twitter) provide an open space to its users to share their views. While this freedom of speech has many positive and constructive aspects, it also propagates biasness or negativity, as a result of conflicts. Consequently, social and electronic media are being continuously used to attack people with hateful content. Due to huge volume of such content, naturally, human inspection to identify hate content is not practical and there is a need to have effective automatic analysis techniques which can identify hate speech so that corrective measures can be taken.

Studies indicate that a tweet’s polarity is an important indicator of a potential hateful content [18]. Typically, the polarity is classified into three categories: clean, offensive, and hateful [19,20]. From the view point of methodology, identification of hate speech has been investigated using a variety of techniques. These include lexical, machine learning, hybrid and, deep learning-based approaches.

The lexical-based approaches rest on the idea that the most important part of classification task is to understand the lexical phrases. Such techniques were introduced in the early 1990s s for understanding semantic and grammatical patterns of a sentence [21]. Among these methods, a study by MacDonald et al. [22] presented feature extraction from text including patterns of language, grammar, manually created rules and domain base knowledge. Likewise, Ruwandika et al. [3] also employed a lexical approach for identification of hate speech. In [23], Gitari et al. presented a three step methodology for classification of hate speech. In the first step, a rule-based approach is used to detect the subject text. In the second step, a lexicon for hate speech is developed. These lexicon are used as features based on ‘negative polarity words’, ‘hate verbs’ and ‘theme-based grammatical patterns’. These three types of features are used to classify text as hate speech. Though simple and intuitive, lexicon-based methods are not very robust in terms of performance.

Machine learning approaches are among the most popular techniques applied to classification of text in general and hate speech identification in particular. Among well-known studies, Davidson et al. [24] present a multi-class classifier to distinguish between hate speech, offensive language, and politically correct text. The authors employed logistic regression with L2 regularization to build a model, which produced effective results. In [16], Yasseri et al. presented a study to distinguish between Islamophobic and non-Islamophobic hate speech on tweets. The authors classify hate speech as weak or strong, for which they have created a text-only model using one-hot encoding. Secondly, they derived the non-text features which include sentiment polarity and count of swear words, speech parts, and named entities. For classification, six different methods are investigated. These include Naïve-Bayes, random forest, logistic regression, decision tree and Support Vector Machine (SVM). Among these, a multi-class SVM produced the most effective results. In another study, Sahi et al. [25] also proposed a model to automatically detect derogatory speech in online tweets. Among the investigated classifiers, authors concluded that Naïve Bayes and SVM outperform other methods.

Combination of multiple techniques (hybrid approaches) have also been employed for hate speech identification. Among such methods, Wester et al. [26] have proposed a hybrid of learning-based and lexical-based approaches for classification of hate speech. The authors used lexicon-based approach to extract complex syntactic and semantic features which are subsequently fed to a learning algorithm. Results suggest that this hybrid model produced better results as opposed to individual lexical and learning methods. In another work, Nagaraju et al. [27] employs a hybrid model for sentiment analysis on football specific tweets. The model uses a combination of Glove, CNN and LSTM for classification and reports promising results.

In the recent years, thanks to developments in different areas of neural networks and deep learning, end-to-end trainable features extractors and classifiers have been proposed [28]. In most cases, deep learning techniques [29] are fast replacing the handcrafted features with automated machine-learned features and classification. In one such study, Saksesi et al. [13], employs a dataset of 1235 tweets from Balai Bahasa of West Java province, Indonesia which were labeled for the binary classification task (hate speech or no hate speech). The technique relies on pre-processing the text and converting words in embeddings while classification was carried out using an LSTM. In the context of the current pandemic, Kumar et al. [30] implemented sentiment analysis on coronavirus public reviews. The authors employed Glove, CNN and bi-directional LSTM for classification of public views. Likewise, Vimali et al. [31] also employs LSTMs for text based sentiment analysis.

A critical review of the existing techniques on the problem of hate speech identification suggests that LSTMs have emerged as a popular choice of researchers in the recent years. While most of the existing techniques target sentiment analysis or identify hate speech in general, the specific problem of Islamophobic hate speech is relatively less explored and makes the subject of our study. The technique proposed in this regard is presented in the following section.

3 Methods

The proposed methodology to identify Islamophobic hate speech relies on a deep learning-based solution. The data is first pre-processed with key steps of case folding, tokenization, cleaning, stemming and removal of stop words. The pre-processed data is then converted to word embeddings using Word2Vec and the resulting sequence of vectors is fed to one-dimensional convolutional layers to extract meaningful features. A bi-directional Long Short-Term Memory network is subsequently employed for sequence modeling and classification. An overview of key processing steps is illustrated in Fig. 1 while the details of each of these steps along with the dataset employed in our study are presented in the following sections.

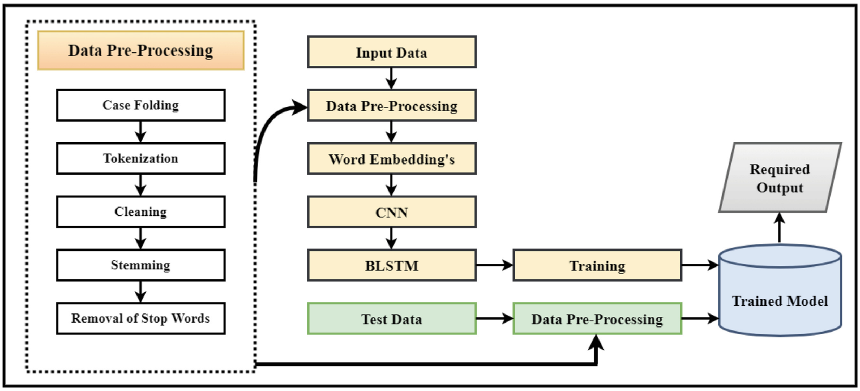

Fig. 1. An overview of key processing steps in the proposed method

3.1 Dataset

While several datasets have been developed for sentiment analysis and hate speech identification, to the best of our knowledge no public dataset is available that is specific to Islamophobic hate speech. Consequently, for the experimental study of our system, we collected multiple publicly available datasets of tweets and filtered out their subsets with purely Islamophobic hate speech content. Likewise, tweets related to Islam but without any hate content were also collected to serve as negative examples for model training. The data was labeled into positive ('1') and negative ('0') examples. We collected a total of 1290 tweets out of which 1032 were employed in the training and 258 in the test set. 10% of training data was employed for validation during the model training phase. A summary of the dataset is presented in Table 1.

Table 1. Statistics of the collected dataset

Total tweets		1290	
Negative tweets	566	Positive tweets	724
Training set	1032	Test set	258

3.2 Data Pre-processing

Data pre-processing is a typical task in text classification that includes cleaning the data and representing it in an appropriate form for further processing. In our study, data pre-processing includes case folding, tokenization, cleaning, stemming and removal of stop words (Fig. 2), as outlined in the following.

- Case Folding: is the conversion of all characters in the text to lower case letters.

- Tokenization: is the division of text stream into phrases, words, symbols etc. These units are termed as tokens.
- Cleaning: is the process to filter unnecessary words, characters and symbols from the text e.g. ‘@’, ‘RT’, ‘https://’, ‘#’ etc.
- Stemming: comprises of minimizing the number of different indexes of a document, e.g. the words ‘useful‘ and ‘usefulness’ have the same semantic.
- Removal of stop words: the non-meaningful words comprising of prepositions, conjunctions, or pronouns are removed from text.

Once the standard pre-processing steps are carried out, we convert the words into embeddings using Word2Vec. The key motivation of an embedding representation is to exploit the relationship between different words (unlike one-hot encoding which treats each word as an independent entity). As a result of this process, each word in the pre-processed tweet is represented by a 300 dimensional vector.

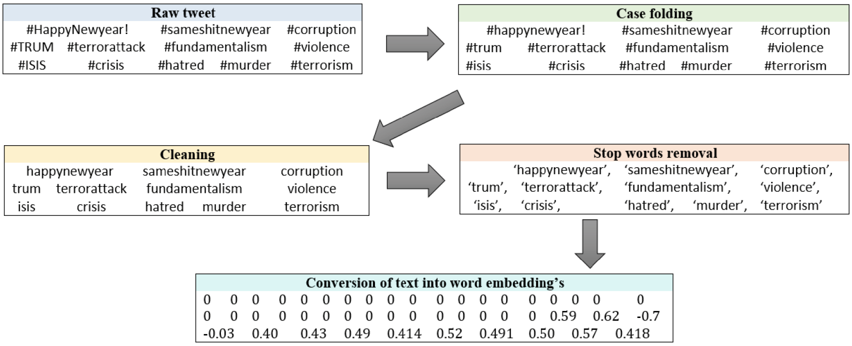
Raw tweet
#HappyNewyear! #sameshitnewyear #corruption
#TRUM #terrorattack #fundamentalism #violence
#ISIS #crisis #hatred #murder #terrorism
Case folding
#happynewyear! #sameshitnewyear #corruption
#trum #terrorattack #fundamentalism #violence
#isis #crisis #hatred #murder #terrorism
Cleaning
happynewyear sameshitnewyear corruption
trum terrorattack fundamentalism violence
isis crisis hatred murder terrorism
Stop words removal
‘happynewyear’, ‘sameshitnewyear’, ‘corruption’,
‘trum’, ‘terrorattack’, ‘fundamentalism’, ‘violence’,
‘isis’, ‘crisis’, ‘hatred’, ‘murder’, ‘terrorism’
Conversion of text into word embedding’s
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0.59 0.62 -0.7
-0.03 0.40 0.43 0.49 0.414 0.52 0.491 0.50 0.57 0.418

Fig. 2. An overview of pre-processing steps employed in our study

3.3 Feature Extraction and Classification

To identify tweets with Islamophobic hate speech, we propose a deep learning-based framework that combines feature extraction and classification in a single module. A combination of 1D Convolutional Neural Network with (different variants of) Recurrent Neural Network is employed for this purpose. The 1D convolutional layers extract robust hierarchical feature representations while the recurrent layers exploit the sequential information in the input tweets to categorize them into positive and negative examples.

From the view point of architectural details, the model comprises of seven convolutional layers with progressively increasing number of filters (from 32 to 512). All conv layers use the ReLU activation function while max pooling is employed to control the spatial dimension (which also prevents over-fitting). The conv layers are followed by a stack of two bi-directional LSTM layers with 64 and 128 hidden units respectively. LSTMs are preferred over simple RNNs due to their ability to model long-term dependencies in the input sequence. Likewise, the motivation of using bi-direction layers is to traverse the input sequence in both forward and backward directions hence exploiting the past as well as the future information to model the sequence. Finally, a single neuron at the output layer (with sigmoid activation function) is employed in the binary classification framework. A generalized overview of the C-BLSTM model is presented in Fig. 3 while the complete architecture of the model is summarized in Table 2. The total number of trainable parameters in the proposed model sums to 21,55,521.

4 Experimental Results and Analysis

To evaluate the effectiveness of the proposed technique, we carried out a comprehensive series of experiments. All experiments are carried out using the dataset distribution listed in Table 1 while the performance is quantified using classification accuracy.

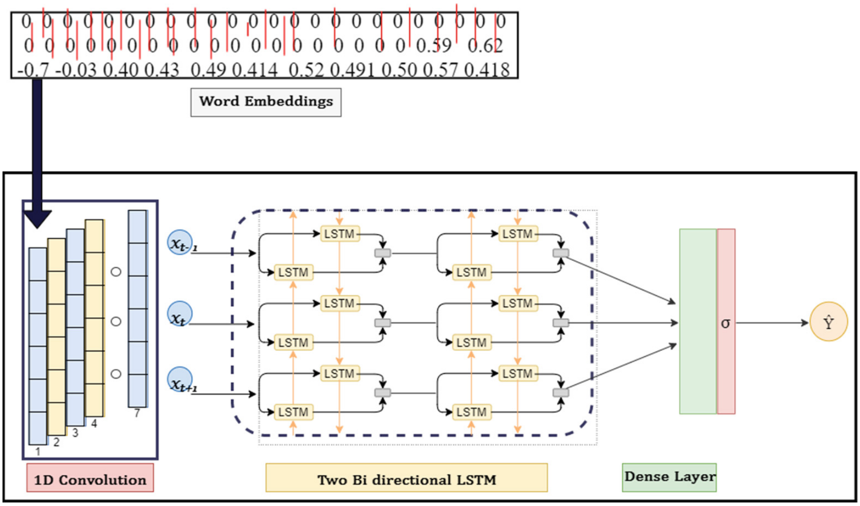

Fig. 3. General architecture of the model with 1D conv layers followed by bi-directional LSTM layers

In the first series of experiments, we directly feed the tweets (represented as embedding vectors) to different variants of RNNs without any convolutional layers to serve as the baseline results. We have employed the vanilla RNNs, GRUs and LSTMs both in single and bi-directional modes. The results of these experiments are summarized in Table 3 which allows drawing some interesting conclusions. In all cases, using bi-directional layers results in an enhanced classification accuracy as opposed the single-direction layers. This is very much natural as exploiting the future information in a sequence contributes to performance improvement. Among the different variants, LSTMs outperform GRUs and simple RNNs. Another interesting observation is that in most cases, the difference between the performance on the training and the test sets is not very high indicating that the model does not over-fit.

In the second series of experiments, we introduce convolutional layers to study the impact of feature learning from raw embeddings. The results of these experiments are presented in Table 4 where it can be seen that in all cases including the convolutional layers serves to significantly enhance the classification accuracy. The highest accuracy is reported by the combination of CNN with bi-directional LSTM reading 90.13%. A comparative overview of the results presented in Table 3 and Table 4 is also illustrated in Fig. 4 indicating the effectiveness learning robust representations through conv layers.

Table 2. Architectural details of proposed 1D CNN + BLSTM model

Layers	Filter size	No. of filters	Output shape	Parameters
Embedding	Vector Size = 300		(150, 300)	15,00,000
Conv1D 1	3	32	(150, 32)	28,832
MaxPooling1D (Pool Size = 2)			(75, 32)	0
Conv1D 2	3	64	(75, 64)	6,208
MaxPooling1D (Pool Size = 2)			(37, 64)	0
Conv1D 3	3	64	(37, 64)	12,352
MaxPooling1D (Pool Size = 2)			(18, 64)	0
Conv1D 4	3	128	(18, 128)	24,704
MaxPooling1D (Pool Size = 2)			(18, 128)	0
Conv1D 5	3	128	(18, 128)	49,280
MaxPooling1D (Pool Size = 2)			(18, 128)	0
Conv1D 6	3	256	(18, 256)	65,792
MaxPooling1D (Pool Size=2)			(18, 256)	0
Conv1D 7	3	512	(18, 512)	2,62,656
MaxPooling1D (Pool Size = 2)			(18, 512)	0
Bidirectional LSTM (Hidden Units: 64)			(18, 64)	1,39,520
Bidirectional LSTM (Hidden Units: 128)			(128)	66,048
Dense layer			(1)	129
Total parameters				21,55,521
Trainable parameters				21,55,521
Non-Trainable parameters				0

We also carried out a number of ablation studies to study the evolution of system performance as a function of number of training examples (Fig. 5), the number of layers in the convolutional (Fig. 6) and the recurrent (Fig. 7) parts of the model. The performance naturally improves with the increase in the number of training examples. Likewise, the performance varies with respect to the number of layers in the model but the variation is not very dramatic indicating the stability of the model.

Table 3. Classification performance with different variants of RNNs

Recurrent network	Training accuracy	Test accuracy
RNN	70.62	68.62
Bi-directional RNN	72.01	70.14
LSTM	74.11	71.58
Bi-directional LSTM	78.09	75.33
GRU	67.91	66.82
Bi-directional GRU	77.60	73.50

Table 4. Classification performance with convolutional and recurrent layers

Recurrent network	Training accuracy	Test accuracy
CNN + RNN	85.59	83.84
CNN + Bi-directional RNN	88.01	85.17
CNN + LSTM	89.11	86.82
CNN + Bi-directional LSTM	92.39	90.13
CNN + GRU	84.60	83.07
CNN + Bi-directional GRU	89.60	87.21

We also present a performance overview of known studies on this problem (Table 5). It is however important to mention that an objective comparison of different techniques is not possible as the reported methods have been evaluated on different datasets with different experimental protocol. The studies are listed with the motivation of providing readers with an idea on the current state-of-the-art on the problem of hate speech identification and the comparison is more of subjective rather than objective. An overall classification accuracy of more than 90% by our system is indeed quite promising.

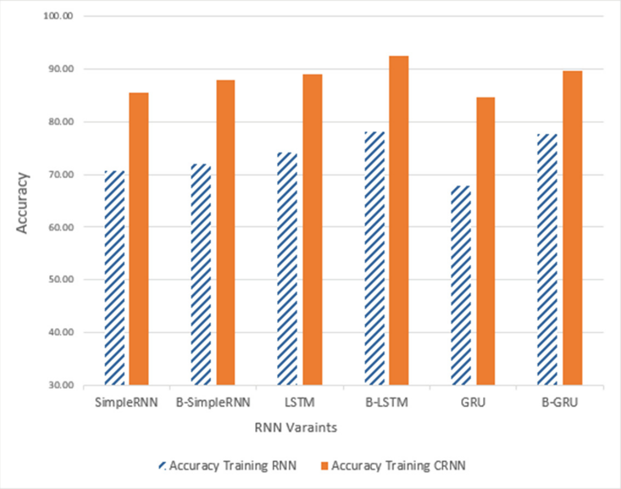

Fig. 4. Performance comparison of recurrent nets with raw embeddings and conv layers

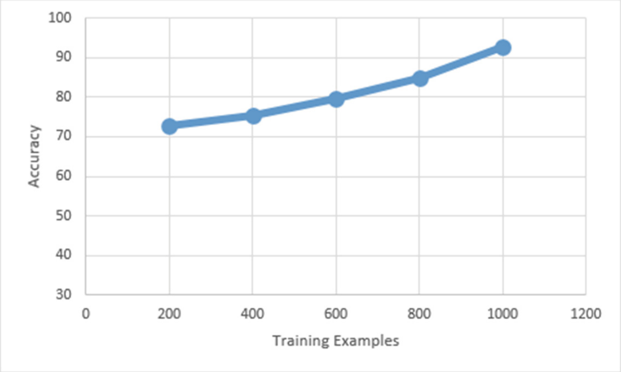

Fig. 5. Classification accuracy as a function of the number of training examples

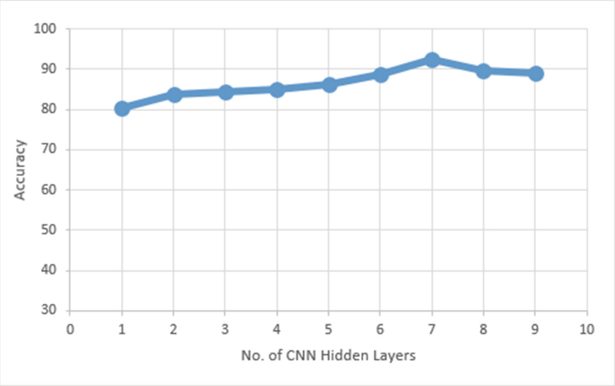

Fig. 6. Classification accuracy as a function of number of conv layers

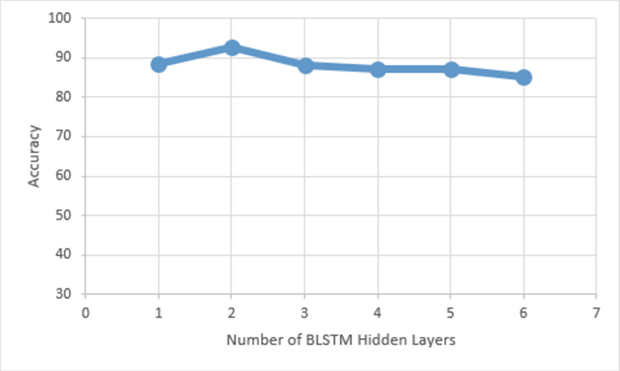

Fig. 7. Classification accuracy as a function of number of BLSTM layers

Table 5. Performance comparison with existing techniques

Reference	Classifier	Dataset	Results(%)
B. Vidgen 2019 [16]	Word Embeddings, gloVe Multi class SVM	4000 tweets	83.00
Y. Kim 2014 [32]	Word Embeddings, CNN	Online tweets	89.00
N.D.T. Ruwandika 2018 [3]	Naïve bayes Tf-idf	1080 tweets	71.90
A. Sucia Saksesi 2018 [13]	Word Embeddings, RNN	1235 tweets	88.00
Nagaraju Y. 2021 [27]	Glove, CNN, LSTM	10007 tweets	85.00
Proposed technique	Word Embeddings, CNN, Bi-directional LSTM	1290 tweets	90.13

5 Conclusion

Islamophobic hate speech identification is a complex problem due to challenges like context sensitive text and non-availability of standard datasets. In this paper, we have presented, an effective technique for classification of such hate speech from online tweets. The technique relies on learning robust feature representations from input tweets using a sequence of convolutional layers while bidirectional LSTMs are employed for sequence modeling. Experimental study on a dataset of nearly 1300 tweets reported a classification accuracy of 90.13%.

In our further exploration on the same subject, we intend to first enhance the size of the dataset and make the compiled data publicly available. Furthermore, in addition to a binary classification problem, specific hate speech classes can also be identified to pose it as a multi-class problem. In addition to tweets, the system can also be extended to identify such hate speech from articles and News. Another possible extension is to analyze the content of News and entertainment channels by generating audio transcriptions of spoken content and applying techniques similar to the one proposed in the current study. From a technical viewpoint, adversarial learning techniques which have been applied to the sentiment analysis problem, can also be investigated for the specific case of Islamophobic hate speech.

References

1. Shield for Muslims (31), 18 July 2021. https://shieldformuslims.wordpress.com/
2. Trust, R.: Islamophobia: a challenge for us all. Runnymede Trust UK **39**(11). www.runnymedetrust.org/uploads/publications/pdfs/islamophobia.pdf
3. Ruwandika, N., Weerasinghe, A.: Identification of hate speech in social media. In: 2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer), pp. 273–278 (2018). IEEE
4. Inc, Y.: Youtube inc. youtube community guidelines [online]. Soc. Media Usage Policy **25**, 3389–3402 (2020)
5. Inc, T.: Twitter inc. the twitter rules [online]. Twitter Usage Policy **9**, 3389–3402 (2020)
6. Inc, F.: Facebook inc. facebook comment policy [online]. Facebook Usage Policy **6**, 3389–3402 (2020)
7. KhosraviNik, M., Esposito, E.: Online hate, digital discourse and critique: exploring digitally-mediated discursive practices of gender-based hostility. Lodz Pap. Pragmat. **14**(1), 45–68 (2018)
8. Weston-Scheuber, K.: Gender and the prohibition of hate speech. QUT Law Justice J. **12**(2), 132–50 (2012)
9. Cowan, G., Khatchadourian, D.: Empathy, ways of knowing, and interdependence as mediators of gender differences in attitudes toward hate speech and freedom of speech. Psychol. Women Q. **27**(4), 300–308 (2003)
10. Frías-Vázquez, M., Arcila, C.: Hate speech against central American immigrants in Mexico: analysis of xenophobia and racism in politicians, media and citizens, pp. 956–960 (2019)

11. Hernández, T.K.: Hate speech and the language of racism in Latin America: a lens for reconsidering global hate speech restrictions and legislation models. U. Pa. J. Int'l L. **32**, 805 (2010)
12. Matamoros-Fernández, A.: Platformed racism: The mediation and circulation of an Australian race-based controversy on twitter, facebook and youtube. Inf. Commun. Soc. **20**(6), 930–946 (2017)
13. Saksesi, A.S., Nasrun, M., Setianingsih, C.: Analysis text of hate speech detection using recurrent neural network. In: 2018 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), pp. 242–248. IEEE (2018)
14. Bonotti, M.: Religion, hate speech and non-domination. Ethnicities **17**(2), 259–274 (2017)
15. ElSherief, M., Kulkarni, V., Nguyen, D., Wang, W.Y., Belding, E.: Hate lingo: a target-based linguistic analysis of hate speech in social media. arXiv preprint arXiv:1804.04257 (2018)
16. Yasseri, T., Vidgen, B.: Detecting weak and strong islamophobic hate speech on social media. J. Inf. Technol. Polit. 2019 **17**(1) (2019)
17. Brown, A.: What is hate speech? part 1: the myth of hate. Law Philos. **36**(4), 419–468 (2017)
18. Calvert, C.: Hate speech and its harms: a communication theory perspective. J. Commun. **47**(1), 4–19 (1997)
19. Al-Hassan, A., Al-Dossari, H.: Detection of hate speech in social networks: a survey on multilingual corpus (2019)
20. Gaydhani, A., Doma, V., Kendre, S., Bhagwat, L.: Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. arXiv preprint arXiv:1809.08651 (2018)
21. Sarvabhotla, K., Pingali, P., Varma, V.: Sentiment classification: a lexical similarity based approach for extracting subjectivity in documents. Inf. Retr. **14**(3), 337–353 (2011)
22. MacDonald, M.C.: Lexical representations and sentence processing: an introduction. Lang. Cogn. Process. **12**(2–3), 121–136 (1997)
23. Gitari, N.D., Zuping, Z., Damien, H., Long, J.: A lexicon-based approach for hate speech detection. Int. J. Multimed. Ubiquitous Eng. 2015 **10**(4), 215–230 (2015)
24. Davidson, T., Warmsley, D., Macy, M., Weber, I.: Automated hate speech detection and the problem of offensive language. arXiv preprint arXiv:1703.04009 (2017)
25. Şahi, H., Kılıç, Y., Sağlam, R.B.: Automated detection of hate speech towards woman on twitter. 2018 3rd International Conference on Computer Science and Engineering (UBMK) 2018, pp. 533–536. IEEE (2018)
26. Wester, A., Øvrelid, L., Velldal, E., Hammer, H.L.: 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)
27. Hegde, S.U., Zaiba, A., Nagaraju, Y., et al.: Hybrid CNN-LSTM model with glove word vector for sentiment analysis on football specific tweets, pp. 1–8. IEEE (2021)
28. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **39**(11), 2298–2304 (2016)
29. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: Deep learning 1 (2016)
30. Mengistie, T.T., Kumar, D.: Deep learning based sentiment analysis on COVID-19 public reviews, pp. 444–449. IEEE (2021)

31. Vimali, J., Murugan, S.: A text based sentiment analysis model using bi-directional LSTM networks, pp. 1652–1658. IEEE (2021)
32. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)