



A Hybrid Deep Learning Model Using Grid Search and Cross-Validation for Effective Classification and Prediction of Suicidal Ideation from Social Network Data

Akshma Chadha¹ · Baijnath Kaushik¹

Received: 15 December 2021 / Accepted: 1 October 2022 / Published online: 16 October 2022
© Ohmsha, Ltd. and Springer Japan KK, part of Springer Nature 2022

Abstract

Suicide deaths due to depression and mental stress are growing rapidly at an alarming rate. People freely express their feelings and emotions on social network sites while they feel hesitant to express such feelings during face-to-face interactions with their dear ones. In this study, a dataset comprising 20,000 posts was taken from Reddit and preprocessed into tokens using a variety of effective word2vec techniques. A new hybrid approach is proposed by combining the attention model in a convolutional neural network and long-short-term-memory. The objective of this research is to develop an effective learning model to evaluate the data on social media for the efficient and accurate identification of people with suicidal ideation. The proposed attention convolution long short-term memory (ACL) model uses hyperparameter tuning using a grid search to select optimized hyperparameters. From the experimental evaluation, it is shown that the proposed model, that is, ACL with Glove embedding after hyperparameter tuning gives the highest Accuracy of 88.48%, Precision of 87.36%, F1 score of 90.82% and specificity of 79.23% and ACL with Random embedding gives the highest Recall of 94.94% when compared to the state-of-the-art algorithms.

Keywords Hyperparameters · Glove embedding · Random embedding · Deep learning · Social networking sites · Suicide · Attention convolutional long short term memory

✉ Akshma Chadha
akshma2013@gmail.com

¹ School of Computer Science and Engineering, Shri Mata Vaishno Devi University, Katra, Jammu, India

1 Introduction

Mental disorders are a global problem. Almost one out of four people suffer from some kind of mental illness worldwide, and three out of four do not receive any kind of treatment, which worsens the problem [1]. Suicide has a direct association with mental disorders. Almost 46% of people who have committed suicide have some kind of mental disorder [2]. The second primary cause of death among young adults worldwide is suicide. With the advancement in social media, people have started sharing their thoughts online and try to take help from different web forums [3]. Sentiment analysis of social media is a trending research topic. Studying the sentiments, opinions, and thoughts of people over different social media sites can help in predicting the results of an election, a movie review, individuals with suicidal ideation, etc. Millions of people use Twitter, which is a widely used social networking site and has over five million tweets per day [4].

Social networking websites are just about every single adult in this era of life. These are increasingly attracting the younger generation as young adults like to meet new people and talk to them rather than talking to a person next to them [5, 6]. Moreover, it is easy to access social media rather than access to appropriate medical help [7, 8]. Social networking sites have become very important in everyday life, and we cannot imagine our lives without social media [9]. People are so addicted to these sites that they share their everyday routines and feelings on social media. Thus, to find people with suicidal ideation, the best way is to go through their social media accounts [10]. Reddit is an open social networking site, so it is convenient to take the data from Reddit, after which the data needs to be preprocessed for applying different learning algorithms, as shown in Fig. 1.

Social networking sites are associated with several social phenomena, such as depression and suicide [11]. It is necessary to find people with this kind of mentality early to decrease the number of suicides [12]. Suicide is not just a word, it can be someone's last breath. Many lives can be saved by identifying such individuals [4].

Suicide is an act of intentionally harming oneself with the purpose of death. Approximately eight lakh individuals die because of suicide every year, which is one individual every 40 s [13]. For each young adult who died by suicide, more than 20 others attempted suicide. Suicide is a major cause of death in the

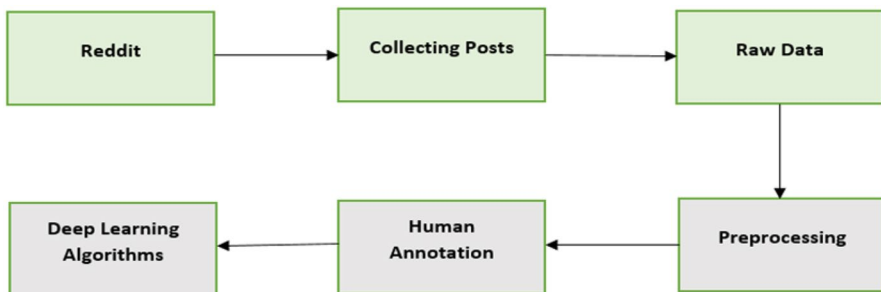


Fig. 1 Learning algorithms used

age group of 15–29 years in India [14]. Suicides spontaneously act as a result of stress, depression, harassment/bullying, relationship problems, domestic violence, mental illness, ragging, family issues, academic issues, financial issues, etc. [15]. Some recent crises may also lead to suicidal ideation, which was seen during the COVID-19 pandemic, as most of the people were seen doing the same [16]. It can be seen that the number of suicides is correlated with a specific marital status, the lack of opportunities for higher education may also lead to suicidal ideation, and a particular gender may also attempt suicide [17].

Loss of quality of life may lead to suicide [18]. Anything that makes us depressed for a longer period of interval pulls down our spirit and love for life and hence may lead to committing suicide [19]. The clinical diagnosis of mental illness is difficult, and seeking help from a professional is even more difficult. Young adults have found a way to share their feelings on social media without being exposed [11, 20]. Deaths by suicide are not certain and are premature deaths resulting in a large number of years of life lost. It is not possible to quantify the risk factors associated with suicidal ideation [21]. Suicides can be prevented if there is a method to find individuals having such ideations beforehand so that different approaches such as promoting psychological motivational sessions and meditation and yoga can be applied to stop them from doing so [22]. Many statistical methods were used to do so, but they did not provide good results. Currently, work is performed in this field using machine learning and deep learning to provide improved results [23].

In this paper, the authors attempt to determine how social media is used to identify suicidal ideation. The use of the Internet and social media has increased considerably, providing patients and experts with a communication medium to develop methods to sense mental health problems among the users of social media [24].

In this paper, some contributions and novelties are added as given below:

- A total of 20,000 tweets were collected based on suicidal features using the Public Reddit Application Programming Interface.
- *Data preprocessing steps*: tokenization, normalization using (Lemmatization and Stemming), and removal of stopwords were used. Further, tokens were converted into vectors using genism.
- TextBlob is applied to label the preprocessed data into the Suicidal (1) and Non-Suicidal (0) classes.
- Word embedding models, such as random embedding and pre-trained glove embedding, were applied to the dataset.
- Hyperparameter tuning is performed using a grid search to select the optimal hyperparameters that can be further used by deep learning models.
- In this study, three different deep learning models, convolutional neural network (CNN), a combination of convolutional neural network and long short-term memory network, and a proposed hybrid model of attention over convolution and long short-term memory neural network (ACL) were used.
- The proposed model ACL (attention over convolution and long short-term memory neural network) combines the best features of the attention model, convolutional neural network, and long short-term memory network.

- The objective of these models is to evaluate the performance and to achieve the highest accuracy on the dataset.
- The proposed model has shown significant accuracy and outperforms the other models.
- To calculate the resulting validity of the model's prediction, a validation test employing a tenfold cross-validation method was used.

The summary of the workflow is shown in Fig. 2.

The remainder of this work is divided into the following sections in the following sequence. The second section of the paper is devoted to a survey of the literature on this topic. The methods used to discover individuals with suicidal thoughts are described in Sect. 3. The paper's results and discussion are presented in Sect. 4. Section 5 brings the paper to a conclusion.

2 Literature Review

Xiaolei Huang et al. have done work specifically on Chinese social media. Data were balanced using oversampling and 10 fold cross-validation was performed. Different classical machine learning algorithms, such as naïve Bayes, logistic regression, J48, random forest, SMO, and SVM were applied to classify the data. SVM outperformed all other models [25].

Michael Mesfin Tadesse et al. performed data preprocessing using different techniques, such as TFIDF, bag of words, statistics, and word embedding. Different machine learning algorithms such as random forest, SVM, XGBoost, and deep learning algorithms such as long short-term memory (LSTM), convolutional neural network (CNN), and a combination of LSTM and CNN were also applied. The hybrid model with a combination of LSTM and CNN yielded the best results [4].

Han-Chin Shing et al. gathered data from Reddit specifically from the SuicideWatch forum and then evaluated them by some clinicians. Data preprocessing steps such as a bag of words, word embedding, linguistic inquiry and word count, NRC, and mental disease lexicons were then applied to the data. Subsequently, the authors applied a support vector machine and convolutional neural network to classify the data as suicidal

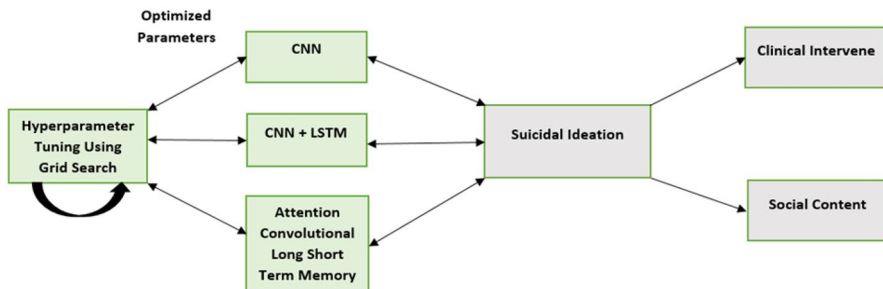


Fig. 2 Summarization of workflow

and non-suicidal. As a result, the Support Vector Machine surpasses the convolutional neural network [26].

Shaoxiong Ji et al. used different feature extraction techniques such as POS count, TFIDF vectors, LIWC features, and topic probability features. Classification models such as random forest, GBDT, XGBoost, SVM, multilayer feed-forward neural network, and long short-term memory with different kinds of features were applied. It is seen that random forest provides the best performance in comparison to the other algorithms [27].

Faisal Muhammad Shah et al. performed different feature extraction techniques such as trainableEmbed features, gloveEmbed features, Word2VecEmbed features, fastextEmbedded features, and metadata features. The different measurement criteria are: early risk detection error (ERDE), latency, and latency weighted F1. Word2VecEmbed in combination with the meta-feature set gives the highest F1 score, precision, and recall at-risk window 23. The same feature set offers the highest F1 at Risk Window 15 and ERDE50 at Risk Window 10 [28].

Michael Sanderson et al. proposes a model to quantify the risk factors of deaths due to suicide. LR and XGB were used for prediction and classification. Results showed that XGB performs really well with promising results and gives a discrimination (AUC: 0.88) and calibration that leads to clinical applications [29].

Syed Tanzeel Rabani et al. explains that every pandemic elevates the rate of suicides and so does COVID-19 as it affects the mental health of the individuals. A dataset was collected from Twitter for a period of 8 months and it consists of 7582 tweets. Some more posts were extracted from Reddit using SuicideWatch subreddit which gives a total of 2678 posts. The complete dataset consists of 10,260 posts. Human annotation is done and the posts are divided into three classes i.e. high risk, low risk and no risk. Data preprocessing is done on the dataset by removing stopwords, performing lemmatization etc. and then Bag of Words (BOW) technique is applied. Different machine learning algorithms as DT, SVM, MNB, LR, 3 ensemble methods as Bagging, AdaBoost, Stochastic Gradient Boosting is then applied on the dataset. It is seen that DT and Bagging outperformed all the algorithms [30].

Shaoxiong Ji et al. proposed a model that works in two steps post representation and relational encoding. Post representation involves two steps i.e. risk related state indicators extraction and LSTM text encoder. The relational encoding includes Vanilla Relation Network (RN) and attention mechanism for giving more importance to high relation scores of text encoding. FastText, CNN, LSTM, Region Based Convolutional Neural Networks (RCNN) and the proposed model RN were applied. The result shows that RN gives the best results with Accuracy of 83.85%, Precision of 83.81%, Recall of 83.85% and F1 Score of 83.77% [31].

Table 1 shows the limitations of the work done in the past.

Table 1 Previous work limitations

| Author | Publication year | Limitation |
|--------------------------------|------------------|--|
| Han-Chin Shing et al. [26] | 2018 | Data is limited to Reddit that fails to generate some specific characteristics. There were no clinicians' assessments |
| Akshma Chadha et al. [32] | 2019 | Only online data is taken into consideration most of the time, offline data should also be considered for better results |
| Mark E. Larsen et al. [33] | 2015 | Mostly the data collection period taken is very less but this period should be increased in the future |
| Chandler McClellan et al. [34] | 2017 | Better filtering techniques should be applied to the data so that the results can be more accurate |
| Kate H. Bentley et al. [35] | 2016 | Usually the individuals who are considered are very less in number, the number of individuals should also be increased for a better research outcome |
| Helen Christensen et al. [36] | 2014 | The work is mainly performed on publically available data i.e. on Twitter, Reddit, etc. but work needs to be performed on private social sites such as Facebook, Instagram |
| Stephen P. Lewis et al. [37] | 2012 | Usually the individuals who are considered are young adults, but we cannot just neglect the middle age and old age people as well |

3 Methodology

The methodology in this paper comprises data collection, preprocessing of data, word embedding, hyperparameter tuning, applying deep learning algorithms, and comparison of these algorithms with the proposed method. The methodology is summarized in Fig. 3.

Section 3.1 explains the data collection done from a public application programming interface (Reddit); Sect. 3.2 covers the cleaning and preprocessing of data; Sect. 3.3 covers the deep learning techniques applied; Sect. 3.4 describes the hyperparameter tuning, Sect. 3.5 explains the word embedding; and Sect. 3.6 describes a comparison of the other techniques with the proposed model.

3.1 Data Collection

Data collection was carried out from the Reddit Social Networking Site with the help of the SuicideWatch Subreddit. A public application programming interface push shift was used. The data collection procedure was completed with uniformly distributed dataset of 20,000 posts i.e. 10,000 posts were suicidal and 10,000 were non-suicidal which makes the dataset balanced. The downloaded data were filtered using a textblob. Label 1 is assigned to the post that has a specificity greater than 0.5264 and a polarity less than 0; otherwise, label 0 is assigned. The labelled comments are then verified with the help of the psychiatrist, that whether the labelling performed using textblob is accurate or not i.e. a comment labelled 1 shows suicidal situation and a comment labelled 0 shows non-suicidal situation. The unprocessed data are shown in Fig. 4.

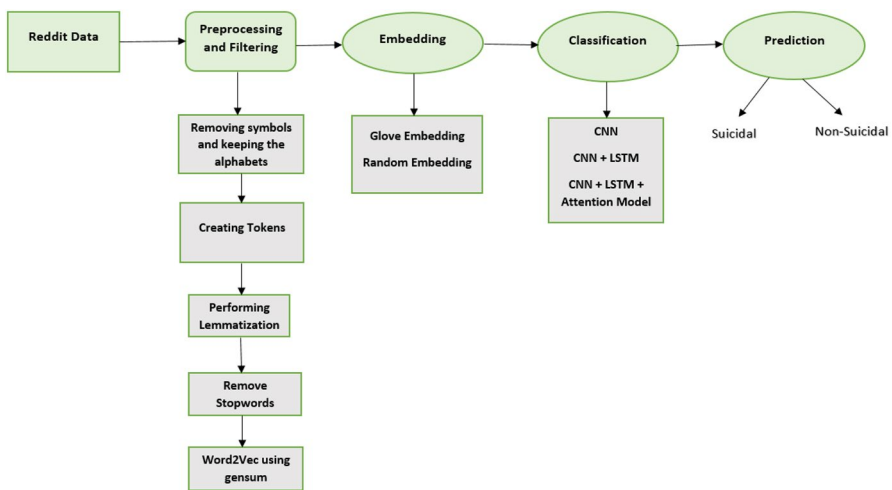


Fig. 3 Steps for data preprocessing, embedding and classification

| Transcripts | Class |
|--|-------|
| It feels good to know I might not wake up tomorrow. | 0 |
| I wrote a note. It's pretty, but i aint nor my life | 0 |
| Life wasnt meant for me I'm fucking tired of living | 1 |
| Im sad every single day, so why shouldn't i end it? | 1 |
| Then I look like a dick for telling them off. Nice. | 0 |
| I am looking for ethical hacker. Please contact me. | 0 |
| I fucking sucks, and I feel like shit all the time. | 1 |
| Pretty much sums it up. Fuck everything. I give up. | 0 |
| Any other shows that i can replace my life with? : | 0 |
| Maybe I'll find something that can cut more easily. | 0 |
| I am feeling sick. Anxious. I need some help please | 1 |
| The world must end now. It's full of stupid people. | 1 |
| Fuck life im out.-im drinkin til my heart gives out | 1 |
| if the day goes on much longer without help ill die | 1 |
| They're both in no matter what, right? Wondering if I should bother having to listen to Joe and Booger again this season (ever??). | 0 |
| I have opinions and they quickly i am excited then depressed my mind is stressed im all over the place all the time any suggestions. | 0 |
| What are the most common words a person new to polyamory might want defined? And feel free to include your definition If so inclined | 0 |
| I'm also a massive fat alcoholic. The literal only reason I'm still alive is because I'm too scared to kill myself. | 1 |

Fig. 4 Unprocessed data

3.2 Data Preprocessing

Data is read first. The data are preprocessed through the semantics of the recognized features. All the entries were preprocessed, and the preprocessing steps were as follows:

- By deleting all symbols, only alphabets are left.
- Split words into tokens.
- Performing Stemming and Lemmatization.
- For consistency, the entire sentence has been converted to lower case.
- Created and removed custom stop words.
- Word2Vec using gensum

Because stemming produces word stems that aren't always identical to the morphological root of the term, as demonstrated in Fig. 5, lemmatization is used instead of stemming in the proposed work. Lemmatization, on the other hand, returns a

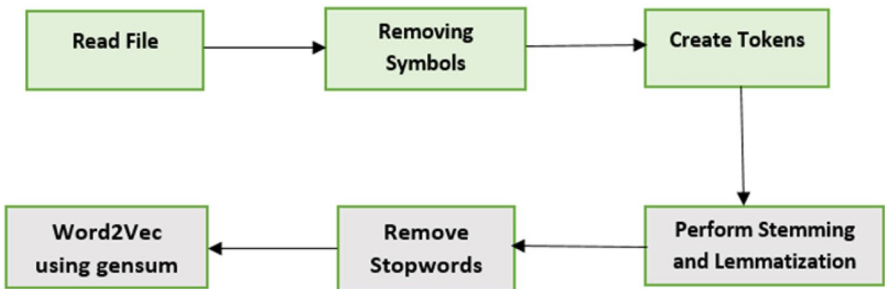


Fig. 5 Steps of data preprocessing

proper dictionary form of a term. The dataset resulted in 17,383 tokens after applying tokenization. The preprocessed data are then stored, as shown in Fig. 6.

In NLP, the context of the words is of most importance; therefore, to resolve this drawback, we use another approach called word embedding, which captures the semantics and context of the words in the document [38]. It is a depiction of words where words with similar meanings have similar representations [25].

3.3 Deep Learning Classification Algorithms

The work on the approaches employed, model design, implementation, and training is described in this part. For the classification challenge, three deep neural network models (CNN, CNN-LSTM, and CNN-LSTM with attention) were used. Two embedding strategies are used to create these models: randomly embedding and glove embedding. All of these algorithms were executed for 25 epochs. The deep learning algorithms were executed in Python 3.7 through Anaconda Navigator, and Gensim, Pandas, Keras, and Sklearn libraries.

3.3.1 Convolutional Neural Network

A one-dimensional (1D) convolutional neural network (CNN) was employed in this study [39]. These network layers can learn numerous and complicated features that a basic neural network cannot [40] due to the underlying architecture. Every input is first processed via the embedding layer, after that the 1D convolution layer, and lastly the max-pooling layer. The result of the convolution is then fed to two dense layers [41]. Determining the probabilities of both classes, sigmoid activation is applied to the final dense layer. In both the dense layers which are in the end, the dropout is employed between them, and in addition to this, it is also utilized in the middle of two layers i.e. convolution 1D and max-pooling. The types of embedded layers in the deep convolutional neural network model are as follows:

| Transcripts | Class |
|---|-------|
| feel good know might not wake up tomorrow | 0 |
| wrote note pretty aint nor my life | 0 |
| life wasnt meant for me im fucking tired living | 1 |
| im sad every single day so shouldnt end | 1 |
| look like dick telling them off nice | 0 |
| looking for ethical hacker please contact me | 0 |
| fucking suck feel like shit all time | 1 |
| pretty much sums up fuck everything give up | 0 |
| any other that can show replace my life with | 0 |
| maybe ill find something that cut easily | 0 |
| feeling sick anxious need some help please | 1 |
| world must end now full stupid people | 1 |
| fuck life im out im drinkin til heart give up | 1 |
| day go much longer without help ill die | 1 |
| theyre both no matter right wondering should bother having listen joe booger again season ever | 0 |
| have opinion they quickly excited depressed my mind stressed im all over placeall time any suggestion | 0 |
| what most common word person new polyamory might want defined feel free includeyour definition inclined | 0 |
| im also massive fat alcoholic literal reason im still alive im scared kill myself | 1 |

Fig. 6 Preprocessed data

Convolution layer: the convolution layers use a convolution operation on the input data and filters to compute the new feature value (convolution kernels). Coefficient values are accommodated in the convolution kernel. Numerous features that are convolved were generated on the given data using various convolution kernels, which are often more valuable than the original input parameters, enhancing the model's efficiency [42].

MaxPooling layer: a technique that yields a lower-dimensional matrix by extracting values through the data which is convolved is termed a pooling layer. As a consequence, a layer generates matrices which are lower-dimension, which can be thought of as a simplified form of convolved features [43]. As a result of the downsampling technique, the system will be more stable because slight changes in the input will not affect the pooled results.

Fully connected layer: in a layer that is fully connected each and every input is associated with every output, across the weights. Its objective is to carry out genuine classification jobs. A traditional CNN cannot separate the expected classes without this layer. A layer that is fully connected, comparable to the layers in artificial neural networks which are hidden [44, 45], passes the flattened feature map or feature vector.

Algorithm: Procedure Convolutional Neural Network

```

1: Input: Features: 17383, training data (80%), testing data (20%), epochs: 25
2: Output: The output is classified in two classes: (0: Non-Suicidal) and (1: Suicidal).
3: Data is read from .pkl file.
4: Iteration = 0
5:   For  $a_i$  in  $(a_1, a_2, a_3, \dots, a_n) \in A$ ,  $A$  is vector having size 20000*100
6:     Iteration = Iteration + 1
7:     if (iteration < epoch)
8:       E = Embedding(vocab_len, 100, input_len= 100) //Create Embeddings
9:       Conv1D(filters, kernel_size, activation, strides= 1, kernel_initializer)
10:      MaxPooling1D(pool_size)
11:      Dense(units, activation = 'sigmoid', kernel_initializer)
12:      Classify as Suicidal or Non-Suicidal
13:     End if
14:   End For

```

The limitations of CNN when processing long sentences to identify semantics to decide whether suicidal or non-suicidal are as follows:

- No mechanism to preserve the time ordering or context of the previous sentence.
- Past information cannot be stored or remembered.

3.3.2 Blend of Convolutional Neural Network and Long Short Term Memory

One-dimensional (1D) convolution and one bidirectional long short-term memory (LSTM) layer were used. The processing of input is done via three-layer, firstly by embedding layer, convolution 1D layer, and max-pooling layer, respectively. After the convolution is performed, the result is passed through the bidirectional LSTM. LSTM is a type of recurrent neural network (RNN) that can utilize feedback connections to memorize long-term dependencies [46]. These abilities were designed to improve the vanishing and exploding gradient problems. This network is powerful because it recalls the past. The output from the LSTM is transferred to the last dense layer in more detail. The dense layer's activation is sigmoid [47], which outputs both classes' probabilities and establishes smooth curves in the range of 0–1, while the model remains differentiable. Between the LSTM layer and the dense layer, a dropout was applied.

Algorithm: Procedure Convolutional Neural Network + Long Short Term Memory

```

1: Input: Features: 17383, training data (80%), testing data (20%), epochs: 25
2: Output: The output is classified in two classes: (0: Non-Suicidal) and (1: Suicidal).
3: Data is read from .pkl file.
4: Iteration = 0
5:   For  $a_i$  in  $(a_1, a_2, a_3, \dots, a_n \in A, A$  is vector having size  $20000 \times 100$ )
6:     Iteration = Iteration + 1
7:     if (iteration < epoch)
8:       E = Embedding(vocab_len, 100, input_len= 100) //Create Embeddings
9:       Conv1D(filters, kernel_size, activation, strides= 1, kernel_initializer)
10:      MaxPooling1D(pool_size)
11:      lstm = Bi-LSTM(m pool) // Features which are extracted fed from 1D Convolution
12:      Classify as Suicidal or Non-Suicidal
13:     End if
14:   End For

```

A sigmoid activation is make use in the cell state and apiece of three gates [48] Forget Gate, Input Gate and Output Gate.

3.3.3 Hybrid Deep Net Attention (CNN-LSTM with Attention) Model

The output from the long short-term memory (LSTM) is delivered to the attention layer. The attention layer helps the model focus on specific portions of the input sentence, which helps the model to classify accurately. The attention layer outputs the context vectors, which are the weighted sum of all input words [49]. To compute the context vector, attention weights are calculated, which indicates which word in the input to focus on [50]. Two dense layers were used to pass the context vector. The dense layer's activation is sigmoid, and the probabilities of both classes are output. Between the MaxPooling layer and the LSTM, a dropout was utilized. The architecture of this hybrid approach is illustrated in Fig. 7.

Algorithm: Procedure Attention-Convolution-Long Short Term Memory (ACL)

```

1: Input: Features: 17383, training data (80%), testing data (20%), epochs: 25
2: Output: The output is classified in two classes: (0: Non-Suicidal) and (1: Suicidal).
3: Data is read from .pkl file.
4: Iteration = 0
5:   For  $a_i$  in  $(a_1, a_2, a_3, \dots, a_n \in A, A$  is vector having size  $20000 \times 100$ )
6:     Iteration = Iteration + 1
7:     if (iteration < epoch)
8:        $E = \text{Embedding}(\text{vocab\_len}, 100, \text{input\_len} = 100)$  //Create Embeddings
9:        $\text{Conv1D}(\text{filters}, \text{kernel\_size}, \text{activation}, \text{strides} = 1, \text{kernel\_initializer})$ 
10:       $\text{MaxPooling1D}(\text{pool\_size})$ 
11:       $\text{lstm} = \text{Bidirectional-LSTM}(\text{m pool})$ 
12:      Build attention(input shape) // Build the attention model
13:       $\text{attention} = \text{Attention}(\text{lstm1})$  //Calculate vector using Attention weights
14:      Classify as Suicidal or Non-Suicidal
15:     End if
16:   End For

```

Long short-term memory (LSTM) has been proposed to overcome these difficulties. The proposed model can:

- Retain the context of past information,
- Evaluate the current context and keep only the relevant information,
- Also, update the current state and move the output to the next layer.

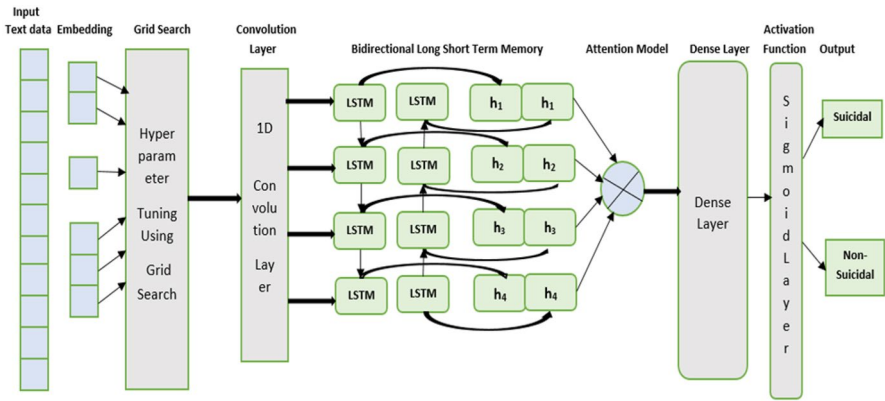


Fig. 7 Proposed hybrid learning algorithm

3.4 HyperParameter Tuning

Hyperparameter tuning is an approach to finding the best set of hyperparameters to configure the model to obtain the best results. Hyperparameter tuning of deep neural network architecture is cumbersome because it is slow to train a deep neural network, and there are several parameters to configure [51]. A grid search was performed to tune the hyperparameters, as shown in Fig. 8. In the case of the grid search, a set of values is given, out of which the best hyperparameter value is chosen. Hyperparameter tuning is executed in Python 3.7 through Anaconda Navigator, using the sklearn library. Hyperparameter tuning using grid search was completed after 25 epochs. The various parameters selected after hyperparameter tuning are listed in Table 2.

Various Hyperparameters that are tuned are:

- *Filters*: these are the learned weights of the convolutional layers. By applying a good filter, it can be assumed that the network is learned well.
- *Kernel size*: the width X height of a filter. The kernel acts as a feature extractor in the case of CNN.
- *Dropout rate*: dropout is a useful regularization approach for neural networks that reduces the risk of overfitting. To preserve the stated likelihood, the approach simply drops units in deep neural networks.
- *Pool size*: the pooling layer is used to reduce the feature map by maintaining the features that are important for classification.
- *Optimizer*: different optimization algorithms are used for in the case of deep learning. It handles all the challenges that the model faces beforehand [48]. The three optimizers that are fed as grid search parameters are stochastic gradient descent, Adam, and RMSProp, as shown in Table 2, and it can be seen clearly that stochastic gradient descent is not selected after tuning the hyperparameters as it converges to a local minimum.

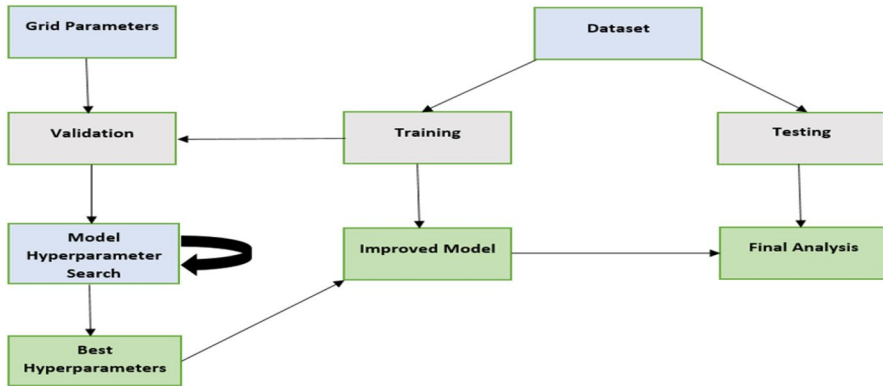


Fig. 8 Hyperparameter tuning using grid search

- *Batch size*: for ease and simplicity, a small cluster is frequently preferred in the process of learning. For checking, a range of 15–130 is considered to be a good choice. Note that the convent is reactive to the group size.

3.5 Word Embedding

Word embedding is a concept used to reduce vocabulary. In this case, words that have the same meaning will be treated alike and will be represented similarly. Each word in the vocabulary is represented by a vector. Each word is given a point in the vector space. The number of features then becomes considerably less compared to vocabulary [52]. Word embedding is superior when compared with the bag-of-words approach, as words with the same meaning are treated similarly. The two types of word embedding that are used in this paper are glove embedding and random embedding. Word embedding is implemented in Python 3.7 through Anaconda Navigator, using the Keras library.

3.6 Comparative Performance Evaluation

Different deep learning algorithms and their combinations, such as convolutional neural network, combination of convolutional neural network and long short-term memory, and a proposed hybrid model ACL are employed on the dataset to analyze the performance of these algorithms and to determine which model gives the highest efficiency.

The given deep learning algorithms were applied on Reddit data consisting of 20,000 posts taken from the SuicideWatch subreddit. The data were then cleaned and preprocessed, as explained in the section. The data was divided into training and testing in an 8:2 ratio, resulting in 16,000 training posts and 4000 testing posts.

Table 2 Optimized hyperparameters obtained using grid search

| Model | Variation | Grid of parameters | Finest parameters |
|---|------------------|--|---|
| 1 CNN | Random embedding | Dropout_rate = [0.1, 0.2, 0.3, 0.4, 0.5] Optimizer = [Adam, RMSprop, SGD] Weight_constraint = [1–5] | Dropout_rate = '0.1' Optimizer = 'Adam' Weight_constraint = 4 |
| | Glove embedding | Neurons = [128, 256] Optimizer = [Adam, RMSprop, SGD] Batch_size = [32, 64, 128, 256] weight_constraint = [1–5] kernel_size = [3, 5] nb_filters = [128, 256] | Neurons = '128' Optimizer = 'RMSprop' Batch_size = 32 Weight_constraint = 5 Kernel_size = 3 nb_filters = 256 |
| 2 CNN + bidirectional LSTM | Random embedding | Dropout_rate = [0.1, 0.2, 0.3, 0.4, 0.5] Optimizer = [Adam, RMSprop, SGD] Units = [10, 20, 29, 39] Kernel_size = [3, 5] nb_filters = [128, 256] Batch_size = [32, 64, 128, 256] | Dropout_rate = '0.1' Optimizer = 'Adam' Units = '10' Kernel_size = '3' nb_filters = '256' Batch_size = '64' |
| | Glove embedding | Optimizer = [Adam, RMSprop, SGD] Units = [10, 20, 64, 128] Kernel_size = [3, 5] Batch_size = [32, 64, 128, 256] Dropout_rate = [0.1, 0.2, 0.3, 0.4, 0.5] | Optimizer = 'Adam' Units = '128' Kernel_size = '3' Batch_size = '128' Dropout_rate = '0.5' |
| 3 CNN + bidirectional LSTM with attention | Random embedding | nb_filters = [128, 256] units = [10, 20, 64, 128] Kernel_size = [3, 5] Batch_size = [32, 64, 128, 256] Neurons = [10, 20, 64, 128] Pool_size = [2, 4] Dropout_rate = [0.1, 0.2, 0.3, 0.4, 0.5] | nb_filters = '128' Units = '128' Kernel_size = 3 Batch_size = '128' Neurons = '64' Pool_size = 4 Dropout_rate = '0.4' |
| | Glove embedding | Pool size = [2, 4] Kernel size = [3, 5] Dropout rate = [0.1, 0.2, 0.3, 0.4, 0.5] units = [10, 20, 64, 128] nb_filters = [128, 256] Neurons = [10, 20, 64, 128] | Pool_size = '4' Kernel_size = '3' Dropout_rate = 0.2 Units = '128' nb_filters = '128' Neurons = '128' |

Deep learning performs admirably in terms of assessing and resolving the semantic meaning and syntactic structure of texts. The authors employed a long short-term memory (LSTM) network that used word embedding as inputs to record the title and text body of the postings, and memory cells to keep the state for a longer length of time. The researchers also used a convolutional neural network (CNN), which has been shown to be an effective tool for text classification and natural language processing.

After applying all the algorithms, it is shown that the proposed model, that is, ACL with Glove embedding after hyperparameter tuning, gives the highest accuracy, precision, specificity, F1 score, and ACL with random embedding, gives the highest recall. As precision is a degree of significance of the outcome, on the other hand, recall is a degree of how numerous outcomes are truly significant which are returned. While identifying suicidal ideation among individuals, it is required that no one with suicidal ideation should be treated as non-suicidal.

3.7 Comparison of the Proposed Work with Existing Methods

In this section, the superiority of the proposed model is proven by comparing its performance with some of the existing methods. An extensive literature survey has been done in Sect. 2, and it is found that most of the work done is on a very small size of data and in some cases the dataset was extremely unbalanced, it uses human annotation for labeling the data and the work done previously has mostly used some of the state-of-the-art Machine Learning algorithms as the dataset was very small and has used TFIDF and Bag of Words word embedding. The models were also not promising as the dataset was highly unbalanced which makes the model biased. It is also shown that most of the Deep Learning algorithms have been used with Bag of Words embedding, Word2Vec, Glove embedding but the size of the dataset was still a constraint as it was very small mostly in the range of 2000–6000. In the most recent work a combination of CNN-LSTM is proposed which gives a promising result [4]. In this proposed model, the author has used Word2Vec. In the proposed ACL model, the author has collected a large amount of recent posts from Reddit using SuicideWatch subreddit, the data collected was balanced and consists of 10,000 suicidal and 10,000 non-suicidal posts. The labeling of the data was performed using TextBlob and was verified with the help of a Psychiatric expert and after that Glove Embedding and Random Embedding were used. In our model, we have used a combination of CNN and LSTM and an Attention model is used to get the relevant data from the LSTM layer. Convolution layer would recognize the important patterns from the input layer, which might have lost if attention layer was inserted after that. So, the author has inserted the attention layer after the LSTM layer to get only the relevant information that was received so that the performance of the model can be increased. In our model, we have also applied hyperparameter tuning using grid search to find the best set of hyperparameters for configuring the model to obtain the best results.

4 Results and Discussion

The dataset consisted of 20,000 posts for training and testing. TextBlob was applied to the data to classify it as suicidal and non-suicidal. The data were first cleaned and preprocessed. Lemmatization was performed on the data, and all the symbols

Table 3 Results of the different learning algorithms used before tuning hyperparameters.

| S no. | Model | Variations techniques used | Accuracy | Precision | Recall | Specificity | F1 score |
|-------|--|----------------------------|----------|-----------|--------|-------------|----------|
| 1 | Convolutional neural network | Random embedding | 81.55 | 82.95 | 87.35 | 72.75 | 85.09 |
| 2 | Convolutional neural network + long short term memory | Glove embedding | 81.23 | 82.73 | 87.02 | 72.44 | 84.82 |
| 3 | Convolutional neural network + long short term memory with attention | Random embedding | 85.18 | 85.56 | 90.71 | 76.78 | 88.06 |
| | | Glove embedding | 85.03 | 85.78 | 90.09 | 77.34 | 87.88 |
| | | Random embedding | 87.50 | 86.76 | 93.53 | 78.35 | 90.02 |
| | | Glove embedding | 87.25 | 86.15 | 93.94 | 77.09 | 89.88 |

Table 4 Results of the different learning algorithms used after tuning hyperparameters

| S no. | Model | Variations techniques used | Accuracy | Precision | Recall | Specificity | F1 score |
|-------|---|-------------------------------------|----------------|----------------|----------------|----------------|----------------|
| 1 | Convolutional neural network | Random embedding Glove embedding | 81.73 81.80 | 83.07 83.12 | 87.52 87.60 | 72.94 73.00 | 85.24 85.27 |
| 2 | Convolutional neural network + long short term memory | Random embedding Glove embedding | 86.38 86.40 | 85.97 86.17 | 92.49 92.24 | 77.09 77.53 | 89.11 89.10 |
| 3 | Convolutional neural network + long short term memory with attention (ACL) (proposed model) | Random embedding Glove embedding | 88.33 88.48 | 86.90 87.36 | 94.94 94.57 | 78.29 79.23 | 90.74 90.82 |

| | | | |
|----------|----------|----------|----------|
| TP= 2106 | FN= 305 | TP= 2098 | FN= 313 |
| FP= 433 | TN= 1156 | FP= 438 | TN= 1151 |

Fig. 9 Confusion matrices of glove embedding and random embedding for CNN model

| | | | |
|----------|----------|----------|----------|
| TP= 2110 | FN= 301 | TP= 2112 | FN= 299 |
| FP= 430 | TN= 1159 | FP= 429 | TN= 1160 |

Fig. 10 Confusion matrices of glove embedding and random embedding for CNN model with hyperparameter tuning

| | | | |
|----------|----------|----------|----------|
| TP= 2187 | FN= 224 | TP= 2172 | FN= 239 |
| FP= 369 | TN= 1220 | FP= 360 | TN= 1229 |

Fig. 11 Confusion matrices of glove embedding and random embedding for CNN+LSTM model

| | | | |
|----------|----------|----------|----------|
| TP= 2230 | FN= 196 | TP= 2224 | FN= 202 |
| FP= 364 | TN= 1225 | FP= 357 | TN= 1232 |

Fig. 12 Confusion matrices of glove embedding and random embedding for CNN+LSTM model with hyperparameter tuning

and stopwords were removed. Dataframes of the preprocessed data are created, and all the NAN values, if any, are dropped. The tokens are created, and words are converted into vectors using genism, and the preprocessed data are saved in a file.

| | | | |
|----------|----------|----------|----------|
| TP= 2255 | FN= 171 | TP= 2265 | FN= 161 |
| FP= 344 | TN= 1245 | FP= 364 | TN= 1225 |

Fig. 13 Confusion matrices of glove embedding and random embedding for CNN+LSTM model with attention model

| | | | |
|----------|----------|----------|----------|
| TP= 2289 | FN= 147 | TP= 2280 | FN= 156 |
| FP= 345 | TN= 1244 | FP= 330 | TN= 1259 |

Fig. 14 Confusion matrices of glove embedding and random embedding for CNN+LSTM model using attention model with hyperparameter tuning

The authors compared the performance of the proposed model with deep learning models such as convolutional neural networks (CNNs) and a combination of convolutional neural networks (CNNs) and long short-term memory (LSTM) on online Reddit data. The data were divided into training and testing in an 8:2 ratio, resulting in 16,000 training posts and 4000 testing posts.

Table 3 displays the results of these methods before hyperparameter adjustment on various matrices.

Because accuracy alone cannot be used to compare the performance of different algorithms, the authors calculated the precision, recall, specificity, and F1 score as a gold standard for comparing their performance.

Table 4 shows the results of these algorithms after tuning the hyperparameters on different matrices.

The confusion matrix of these three algorithms with random embedding and glove embedding without using hyperparameter tuning and with random embedding and glove embedding using hyperparameter tuning is shown in Figs. 9, 10, 11, 12, 13, 14, to obtain the positive and negative results of the dataset produced. A confusion matrix was used to find the true negative, true positive, false negative, and false positive values among the 4000 posts.

There is a risk of overfitting the test set while assessing alternative hyperparameters for the estimator because the values can be changed until the estimator performs optimally. The knowledge from the test set can then infiltrate into the model and assessment metrics, causing generalization scores to no longer report on the same thing. To establish the model's predictions' final validity, a test for attestation is essential. For authenticating the model's capability, K-fold cross-validation assures that each dataset observation has a chance of appearing in both the training

Table 5 Tenfold cross validation for model evaluation

| Model | Techniques | Folds | | | | | | | | | | Mean |
|--|--------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | |
| Convolutional neural network | Randomly initialized embedding | 96.54 | 97.55 | 91.33 | 92.58 | 95.66 | 97.25 | 94.88 | 96.99 | 93.33 | 92.56 | 95.06 |
| | Glove embedding | 96.52 | 97.55 | 91.22 | 89.52 | 93.74 | 98.99 | 92.54 | 93.88 | 98.45 | 94.12 | 94.65 |
| Convolutional neural network (optimized parameters) | Randomly initialized embedding | 96.44 | 91.78 | 92.73 | 95.22 | 97.44 | 91.54 | 92.88 | 97.52 | 94.69 | 91.49 | 94.77 |
| | Glove embedding | 97.49 | 98.22 | 99.45 | 91.57 | 97.29 | 95.47 | 97.33 | 96.54 | 98.99 | 94.78 | 96.71 |
| Convolutional neural network + long short term memory | Randomly initialized embedding | 98.22 | 99.82 | 94.52 | 97.45 | 98.69 | 91.56 | 92.44 | 97.48 | 97.56 | 98.36 | 96.61 |
| | Glove embedding | 91.22 | 97.56 | 96.52 | 89.25 | 98.22 | 96.74 | 98.99 | 96.58 | 99.99 | 97.79 | 96.28 |
| Convolutional neural network + long short term memory (optimized parameters) | Randomly initialized embedding | 96.58 | 97.88 | 97.58 | 99.52 | 93.54 | 98.11 | 97.52 | 96.78 | 96.58 | 97.99 | 97.20 |
| | Glove embedding | 97.52 | 98.69 | 97.22 | 99.41 | 94.52 | 97.69 | 98.22 | 98.65 | 97.33 | 94.25 | 97.25 |
| ACL (proposed model) | Randomly initialized embedding | 98.56 | 99.77 | 99.99 | 1.0 | 99.99 | 1.0 | 99.98 | 99.99 | 1.0 | 1.0 | 99.82 |
| | Glove embedding | 98.90 | 1.0 | 1.0 | 1.0 | 99.99 | 1.0 | 1.0 | 99.99 | 1.0 | 1.0 | 99.88 |
| ACL (proposed model) (optimized parameters) | Randomly initialized embedding | 98.58 | 1.0 | 1.0 | 1.0 | 99.99 | 99.99 | 99.96 | 99.99 | 1.0 | 1.0 | 99.85 |
| | Glove embedding | 98.89 | 1.0 | 1.0 | 99.99 | 99.99 | 1.0 | 99.99 | 99.99 | 1.0 | 1.0 | 99.88 |

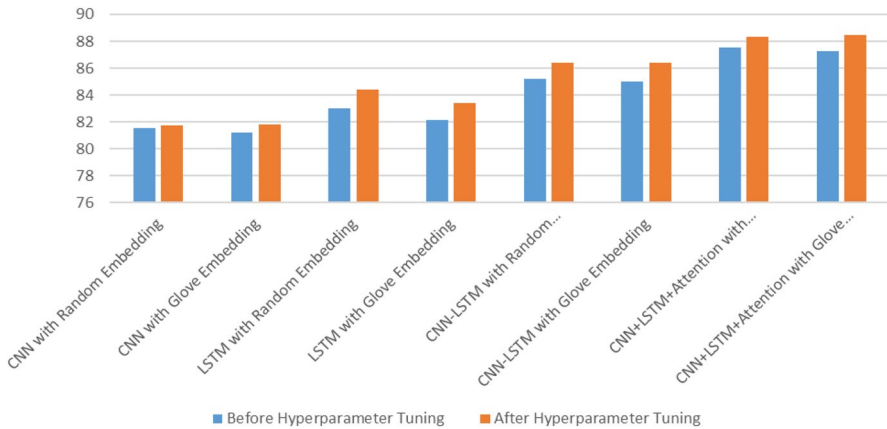


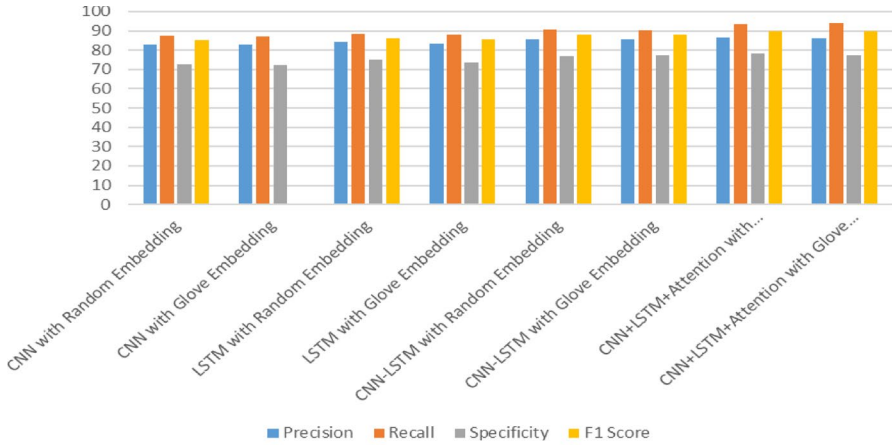
Fig. 15 Evaluation for CNN, fusion of CNN-LSTM and fusion along ATTENTION using two embedding technique based on testing accuracy before and after hyperparameter tuning

and testing datasets. As shown in Table 5, the performance metric for all three models is the average of the ten tenfold cross-validation data.

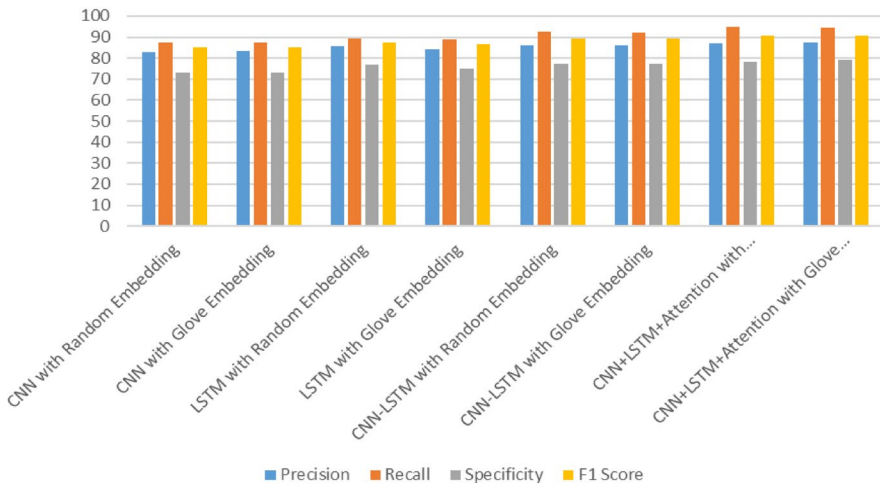
After applying all the algorithms, it is shown that the model proposed i.e. ACL with Glove embedding after hyperparameter tuning gives the highest Accuracy of 88.48%, Precision of 87.36%, F1 score of 90.82% and specificity of 79.23% and ACL with Random embedding gives the highest Recall of 94.94%. A comparison of CNN, CNN-LSTM, and CNN-LSTM with attention using two embedding techniques based on testing accuracy before and after hyperparameter tuning is shown in Fig. 15. The evaluation based on specificity, recall, precision, and F1 score for CNN, CNN-LSTM, and fusion of CNN-LSTM along attention and making use of two embedding techniques before and after parameter tuning is shown in Fig. 16.

5 Conclusion

The authors of this article looked at different posts based on their content from SuicideWatch subreddit to see if they were suicidal or not. The data collection procedure was completed with uniformly distributed dataset of 20,000 posts i.e. 10,000 posts were suicidal and 10,000 were non-suicidal which makes the dataset balanced. Data are read first. TextBlob is applied to this data and it is classified into Suicidal (1) and Non-Suicidal (0) as shown in Fig. 4. The data are preprocessed through the semantics of the recognized features. Lemmatization was performed on the data, and all the symbols and stopwords were removed. Dataframes of the preprocessed data are created, and all the NAN values, if any, are dropped. The tokens are created, and words are converted into vectors using genism, and the preprocessed data are saved in a file. After the preprocessing phase, two different types of word embedding were applied to the dataset: glove embedding and random embedding. Subsequently, hyperparameter tuning was performed using



(a) Specificity, Precision, Recall and F1 score before Hyperparameter Tuning



(b) Specificity, Precision, Recall and F1 score after Hyperparameter Tuning

Fig. 16 Evaluation based on specificity, recall, precision, and F1 score for CNN, fusion of CNN-LSTM and CNN-LSTM fusion along attention using two embedding technique before and after parameter tuning

a grid search to find the best hyperparameters that can be used in different deep learning algorithms.

Different deep learning algorithms and their combinations, such as convolutional neural network, combination of convolutional neural network and long short-term memory, and a proposed hybrid model ACL are implemented on the dataset to calculate the performance of these algorithms and to determine which model gives the highest efficiency.

The proposed model has the best features of the three different models: attention model, convolutional neural network, and long short-term memory. During the preprocessing phase, the attention model focuses on the details of the data that are required and specific words while performing the prediction, the convolutional neural network component displays a local pattern using the first solid color marking, which can clarify the text by considering not only one word but also a mixture of different ones with predefined size to test their intensive learning combination and interpretation, and the use of long short-term memory is to store sentence information in longer terms, keeping past tokens, and solving the problem of vanishing gradients.

After applying all the algorithms, it is shown that the model proposed i.e. ACL with Glove embedding after hyperparameter tuning gives the highest Accuracy of 88.48%, Precision of 87.36%, F1 score of 90.82% and specificity of 79.23% and ACL with Random embedding gives the highest Recall of 94.94%.

In the future, this work may be extended by including data such as images, videos, and blogs on other social media platforms. Furthermore, data from private social networking sites such as Facebook and Instagram can be used to do future work. The work will be extended by using a larger dataset, and we would also perform some new models such as BERT and some transfer learning models to develop an effective model for solving multiclass classification on the suicidal ideation dataset.

Acknowledgements Dr. Manu Arora, MD Psychiatry, Govt. Medical College (GMC), Jammu, Union Territory of Jammu & Kashmir (India), provided accurate instruction and support in diagnosing suicidal causes and features from the patient's record, and the authors are grateful.

Data Availability The data that underpins this article will be shared on request from the corresponding author.

References

1. Karmen, C., Hsiung, R.C., Wetter, T.: Screening internet forum participants for depression symptoms by assembling and enhancing multiple NLP methods. *Comput. Methods Programs Biomed.* **120**(1), 27–36 (2015)
2. Mental Health Data, https://www.who.int/mental_health/media/investing_mnh.pdf, 22/03/2021
3. Ji, S., Pan, S., Li, X., Cambria, E., Long, G., Huang, Z.: Suicidal ideation detection: A review of machine learning methods and applications. *IEEE Trans. Comput. Soc. Syst.* **8**(1), 214–226 (2020)
4. Tadesse, M.M., Lin, H., Xu, B., Yang, L.: Detection of suicide ideation in social media forums using deep learning. *Algorithms* **13**(1), 7 (2020)
5. Chadha, A., Kaushik, B.: Suicidal ideation from the perspective of social and opinion mining. In: *Proceedings of ICRIC 2019*. Springer, Cham, pp 659–670 (2020)
6. De Choudhury, M., Counts, S., Horvitz, E.: Social media as a measurement tool of depression in populations. In: *Proceedings of the 5th annual ACM web science conference*, pp. 47–56 (2013)
7. Sawhney, R., Manchanda, P., Mathur, P., Shah, R., Singh, R.: Exploring and learning suicidal ideation connotations on social media with deep learning. In: *Proceedings of the 9th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pp 167–175 (2018)
8. Gaur, M., Alambo, A., Sain, J. P., Kursuncu, U., Thirunarayan, K., Kavuluru, R., Pathak, J.: Knowledge-aware assessment of severity of suicide risk for early intervention. In: *The World Wide Web Conference*, pp 514–525 (2019)
9. Teismann, T., Brailovskaia, J., Margraf, J.: Positive mental health, positive affect and suicide ideation. *Int. J. Clin. Health Psychol.* **19**(2), 165–169 (2019)

10. Odea, B., Wan, S., Batterham, P.J., Calear, A.L., Paris, C., Christensen, H.: Detecting suicidality on Twitter. *Internet Interv.* **2**(2), 183–188 (2015)
11. Krishnamurthy, G., Majumder, N., Poria, S., Cambria, E.: A deep learning approach for multimodal deception detection. arXiv preprint <https://arXiv.org/1803.00344> (2018)
12. Abboute, A., Boudjeriou, Y., Entringer, G., Azé, J., Bringay, S., Poncelet, P.: Mining twitter for suicide prevention. In: *International Conference on Applications of Natural Language to Data Bases/Information Systems*. Springer, Cham, pp 250–253 (2014)
13. Mental Health Data, <https://www.who.int/data/gho/data/themes/mental-health>, 20/2/2021
14. Data of Suicide in India, https://en.wikipedia.org/wiki/Suicide_in_India, 24/2/2021
15. Suicide Definition, <https://en.wikipedia.org/wiki/Suicide>, 24/2/2021
16. Le, H., Khan, B.A., Murtaza, S., Shah, A.A.: The increase in suicide during the COVID-19 pandemic. *Psychiatr. Ann.* **50**(12), 526–530 (2020)
17. Wang, Y., Tang, J., Li, J., Li, B., Wan, Y., Mellina, C., Chang, Y.: Understanding and discovering deliberate self-harm content in social media. In: *Proceedings of the 26th International Conference on World Wide Web*, pp 93–102 (2017)
18. Cheng, Q., Li, T.M., Kwok, C.L., Zhu, T., Yip, P.S.: Assessing suicide risk and emotional distress in Chinese social media: a text mining and machine learning study. *J. Med. Internet Res.* **19**(7), e243 (2017)
19. Birjali, M., Beni-Hssane, A., Erritali, M.: Machine learning and semantic sentiment analysis based algorithms for suicide sentiment prediction in social networks. *Procedia Comput. Sci.* **113**, 65–72 (2017)
20. Luo, J., Du, J., Tao, C., Xu, H., Zhang, Y.: Exploring temporal suicidal behavior patterns on social media: insight from Twitter analytics. *Health Inform. J.* **26**(2), 738–752 (2020)
21. Ricciardelli, L.A., Nackerud, L., Quinn, A.E., Sewell, M., Casiano, B.: Social media use, attitudes, and knowledge among social work students: ethical implications for the social work profession. *Soc. Sci. Human. Open* **2**(1), 100008 (2020)
22. Cook, B.L., Progovac, A.M., Chen, P., Mullin, B., Hou, S., Baca-Garcia, E.: Novel use of natural language processing (NLP) to predict suicidal ideation and psychiatric symptoms in a text-based mental health intervention in Madrid. *Comput. Math. Methods Med.* **2016**, 1–8 (2016)
23. De Choudhury, M., Gamon, M., Counts, S., Horvitz, E.: Predicting depression via social media. In: *Proceedings of the International AAAI Conference on Web and Social Media* (Vol. 7, No. 1) (2013)
24. Castillo-Sánchez, G., Marques, G., Dorrnoro, E., Rivera-Romero, O., Franco-Martín, M., De la Torre-Díez, I.: Suicide risk assessment using machine learning and social networks: a scoping review. *J. Med. Syst.* **44**(12), 1–15 (2020)
25. Huang, X., Zhang, L., Chiu, D., Liu, T., Li, X., Zhu, T.: Detecting suicidal ideation in Chinese microblogs with psychological lexicons. In: *2014 IEEE 11th Intl Conf on Ubiquitous Intelligence and Computing and 2014 IEEE 11th Intl Conf on Autonomic and Trusted Computing and 2014 IEEE 14th Intl Conf on Scalable Computing and Communications and Its Associated Workshops*, pp 844–849. IEEE (2014)
26. Shing, H. C., Nair, S., Zirikly, A., Friedenber, M., Daumé III, H., Resnik, P.: Expert, crowdsourced, and machine assessment of suicide risk via online postings. In: *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*, pp 25–36 (2018)
27. Ji, S., Yu, C.P., Fung, S.F., Pan, S., Long, G.: Supervised learning for suicidal ideation detection in online user content. *Complexity* (2018). <https://doi.org/10.1155/2018/6157249>
28. Shah, F. M., Ahmed, F., Joy, S. K. S., Ahmed, S., Sadek, S., Shil, R., Kabir, M. H.: Early depression detection from social network using deep learning techniques. In: *2020 IEEE Region 10 Symposium (TENSYP)*, pp 823–826. IEEE. (2020)
29. Sanderson, M., Bulloch, A.G., Wang, J., Williams, K.G., Williamson, T., Patten, S.B.: Predicting death by suicide following an emergency department visit for parasuicide with administrative health care system data and machine learning. *EClinicalMedicine* **20**, 100281 (2020)
30. Rabani, S. T., Khan, Q. R., din Khanday, A. M. U.: A novel approach to predict the level of suicidal ideation on social networks using machine and ensemble learning (2021)
31. Ji, S., Li, X., Huang, Z., Cambria, E.: Suicidal ideation and mental disorder detection with attentive relation networks. *Neural Comput. Appl.* **34**, 1–11 (2021)
32. Chadha, A., Kaushik, B.: A survey on prediction of suicidal ideation using machine and ensemble learning. *Comput. J.* (2019). <https://doi.org/10.1093/comjnl/bxz120>

33. Larsen, M.E., Boonstra, T.W., Batterham, P.J., O’Dea, B., Paris, C., Christensen, H.: We feel: mapping emotion on Twitter. *IEEE J. Biomed. Health Inform.* **19**(4), 1246–1252 (2015)
34. McClellan, C., Ali, M.M., Mutter, R., Kroutil, L., Landwehr, J.: Using social media to monitor mental health discussions—evidence from Twitter. *J. Am. Med. Inform. Assoc.* **24**(3), 496–502 (2017)
35. Bentley, K.H., Franklin, J.C., Ribeiro, J.D., Kleiman, E.M., Fox, K.R., Nock, M.K.: Anxiety and its disorders as risk factors for suicidal thoughts and behaviors: a meta-analytic review. *Clin. Psychol. Rev.* **43**, 30–46 (2016)
36. Christensen, H., Batterham, P.J., O’Dea, B.: E-health interventions for suicide prevention. *Int. J. Environ. Res. Public Health* **11**(8), 8193–8212 (2014)
37. Lewis, S.P., Heath, N.L., Sornberger, M.J., Arbutnott, A.E.: Helpful or harmful? An examination of viewers’ responses to nonsuicidal self-injury videos on YouTube. *J. Adolesc. Health* **51**(4), 380–385 (2012)
38. Wang, P., Xu, J., Xu, B., Liu, C., Zhang, H., Wang, F., Hao, H.: Semantic clustering and convolutional neural network for short text categorization. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (volume 2: Short Papers)*. pp 352–357, (2015)
39. Ruiz, V., Shi, L., Quan, W., Ryan, N., Biernesser, C., Brent, D., Tsui, R.: CLPsych2019 Shared task: predicting suicide risk level from Reddit posts on multiple forums. In: *Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology*, pp. 162–166 (2019)
40. He, X., Deng, L.: Deep learning in natural language generation from images. In: *Deep learning in natural language processing*, pp. 289–307. Springer, Singapore (2018)
41. Long Short Term Memory, <https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/>, 29/03/2021
42. Beaini, D., Achiche, S., Duperré, A., Raison, M.: Deep green function convolution for improving saliency in convolutional neural networks. *Vis. Comput.* **37**, 1–18 (2020)
43. Scherer, D., Müller, A., & Behnke, S.: Evaluation of pooling operations in convolutional architectures for object recognition. In: *International conference on artificial neural networks*. Springer, Berlin, Heidelberg, pp 92–101 (2010)
44. Kim, M.G., Ko, H., Pan, S.B.: A study on user recognition using 2D ECG based on ensemble of deep convolutional neural networks. *J. Ambient. Intell. Humaniz. Comput.* **11**(5), 1859–1867 (2020)
45. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint* <https://arxiv.org/1409.1556> (2014)
46. Udelhoven, T., Schütt, B.: Capability of feed-forward neural networks for a chemical evaluation of sediments with diffuse reflectance spectroscopy. *Chemom. Intell. Lab. Syst.* **51**(1), 9–22 (2000)
47. Tran, K., Bisazza, A., & Monz, C.: Recurrent memory networks for language modeling. *arXiv preprint* <https://arxiv.org/1601.01272> (2016)
48. Hyperparameters, <https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a>, 23/02/2021
49. Convolutional Neural Network, <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>, 10/03/2021
50. Attention Model, <https://www.analyticsvidhya.com/blog/2019/11/comprehensive-guide-attention-mechanism-deep-learning/>, 24/03/2021
51. Yogatama, D., Mann, G.: Efficient transfer learning method for automatic hyperparameter tuning. In: *Artificial intelligence and statistics*, pp. 1077–1085. PMLR (2014)
52. Wang, P., Xu, B., Xu, J., Tian, G., Liu, C.L., Hao, H.: Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing* **174**, 806–814 (2016)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.