Article

Detection of Suicide Ideation in Social Media Forums Using Deep Learning

Michael Mesfin Tadesse [1] [iD], Hongfei Lin [1,*], Bo Xu [1,2] and Liang Yang [1]

[1] School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China; michaelmesfinted82@gmail.com (M.M.T.); xubo@dlut.edu.cn (B.X.); liang@dlut.edu.cn (L.Y.)

[2] State Key Laboratory of Cognitive Intelligence, iFLYTEK, Shenzhen 518000, China

*   Correspondence: hflin@dlut.edu.cn

check for updates

Abstract: Suicide ideation expressed in social media has an impact on language usage. Many at-risk individuals use social forum platforms to discuss their problems or get access to information on similar tasks. The key objective of our study is to present ongoing work on automatic recognition of suicidal posts. We address the early detection of suicide ideation through deep learning and machine learning-based classification approaches applied to Reddit social media. For such purpose, we employ an LSTM-CNN combined model to evaluate and compare to other classification models. Our experiment shows the combined neural network architecture with word embedding techniques can achieve the best relevance classification results. Additionally, our results support the strength and ability of deep learning architectures to build an effective model for a suicide risk assessment in various text classification tasks.

Keywords: suicide ideation; early suicide detection; linguistic metadata; word embedding; machine learning; deep learning; Reddit social media

---

1. Introduction

Every year, almost 800,000 people commit suicide. Suicide remains the second leading cause of death among a young generation with an overall suicide rate of 10.5 per 100,000 people. It is predicted that by 2020, the death rate will increase to one every 20 s [1]. Almost 79% of the suicides occur in low- and middle-income countries where the resources for the identification and management is often scarce and insufficient.

Suicide ideation is viewed as a tendency to end ones' life ranging from depression, through a plan for a suicide attempt, to an intense preoccupation with self-destruction [2]. At-risk individuals can be recognized as suicide ideators (or planners) and suicide attempters (or completers) [3]. The relationship between these two categories is often a subject of discussion in research communities. According to some studies, most of the individuals with suicide ideation do not make suicide attempts. For instance, Klonsky et al. [4] believes that most of the oft-cited risk factors (depression, hopelessness, frustration) connected with suicide are the predictors of suicide ideation, not the progression from the ideation to attempt. However, Pompili et al. [5] reveals that a suicide ideator and suicide attempter can be quite similar to "several variables assumed to be risk factors for suicidal behavior". In WHO countries, early detection of suicide ideation has been developed and implemented as a national suicide prevention strategy to work towards the global market with the common aim to reduce the suicide rates by 10% by 2020 [1].

Over recent years, social media has become a powerful "window" into the mental health and well-being of its users, mostly young individuals. It offers anonymous participation in different cyber communities to provide a space for a public discussion about socially stigmatized topics. Generally,

more than 20% of suicide attempters and 50% of suicide completers leave suicide notes [6]. Thus, any written suicidal sign is viewed as a worrying sign, and an individual should be questioned on the existence of individual thoughts. According to Choudhury et al. [7], social media text, such as blog posts, forum messages, tweets, and other online notes, is usually recorded in the present and is well preserved. In comparison to an offline text, it can minimize any misleading text interpretations produced by a retrospective analysis.

Social media with its mental health-related forums has become an emerging study area in computational linguistics. It provides a valuable research platform for the development of new technological approaches and improvements which can bring a novelty in suicide detection and further suicide risk prevention [8]. It can serve as a good intervention point. Kumar et al. [9] studied the posting activities of Reddit SuicideWatch users who follow news about celebrity suicides. He introduced a method that can be efficient in preventing high profile suicides. Choudhury et al. [7] studied the shift from a mental health discourse to suicide ideation in Reddit social media. He developed a propensity score matching-based statistical approach to derive the distinct markers of this shift. Recently, Ji et al. [10] has developed a novel data protecting the solution and advanced optimization strategy (AvgDiffLDP) for early detection of suicide ideation.

Apart from traditional text classification approaches, deep learning methods have already made an impressive advance in the field of computer vision and pattern recognition. While traditional machine learning approaches liaise heavily on time-consuming and often incomplete handcrafted features, neural networks based on dense vector representations can produce superior results on various Natural language processing (NLP) tasks [11]. The growing success of word embedding [12,13] and deep neural networks are reflected in outperforming more traditional machine learning systems for suicide risk assessments.

The primary objective of our study is to share the knowledge of suicide ideation in Reddit social media forums from a data analysis perspective using effective deep learning architectures. Our main task is to explore the potential of Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN) and their combined model applied in multiple classification tasks for suicide ideation struggles. We try to test if an implementation of CNN and LSTM classifiers into one model can improve the language modeling and text classification performance. We will try to demonstrate that LSTM-CNN model can outperform the performance of its individual CNN and LSTM classifiers as well as more traditional machine learning systems for suicide-related topics. Potentially, it can be embedded on any online forum's and blog's data sets.

In our experiment, we first choose the data source, define our proposed model and analyze the baseline characteristics. Then, we compute the frequency of n-grams, such as unigrams and bigrams, in the dataset to detect the presence of suicidal thoughts. We evaluate the experimental approach based on the baseline and our proposed model. Finally, we train our LSTM-CNN model using 10-fold cross-validation to identify our best hyper-parameter selection for suicide ideation detection. For our dataset, we apply the data collected from Reddit social media which allow its users to create longer posts.

Our study has specific three-fold contributions:

- N-gram analysis: we evaluate the n-gram analysis to show that the expressions of suicidal tendencies and reduced social engagements are often discussed in suicide-related forums. We identify the transition towards the social ideation associated with different psychological stages such as heightened self-focused attention, a manifestation of hopelessness, frustration, anxiety or loneliness.
- Classical features analysis: using CNN, LSTM and LSTM-CNN combined model analysis, we evaluate bag of words, TF-IDF and statistical features performance over word embedding.
- Comparative evaluation: we explore the performance of LSTM-CNN combined class of deep neural networks as our proposed model for detection of suicide ideation tasks to improve the state-of-the-art method. In terms of evaluation metrics, we compare its strength and potential

with CNN and LSTM deep learning techniques and four traditional machine learning classifiers including SVM, NB, RF and XGBoost) on the real-world dataset.

The structure of our paper is as follows: Section 2 describes related work on suicide and suicide ideation detected in social media. Section 3 analyzes a data collection method. Section 4 introduces a proposed methodology. For combined neural network classification approaches, it conducts the data pre-processing followed by a word embedding process. Section 5 focuses on an experimental set up concerning the baseline, model architectures, parameters and evaluation metrics. In section 5, we examine the results and the most powerful machine learning techniques for the detection of suicide ideation. In Section 6, we conclude our study and discuss the main limitations of our work. Finally, we define the main directions for future work.

2. Background and Related Work

In recent years, a considerable number of experiments has been developed to emphasize an influencing power of social media on suicide ideation. Choudhury et al. [7] developed a statistical approach based on a score matching model to derive some distinct markers detecting the transition from a mental health discourse to suicide ideation. According to the authors, this transition can be accompanied by three specific psychological stages: thinking, ambivalence and decision-making. The first stage includes thoughts of anxiety, hopelessness and distress. The second stage is related to lowered self-esteem and reduced social cohesion. The third stage is accompanied by aggression and a suicide commitment plan. Similarly, Coppersmith et al. [14] examined the behavioral shifts of the users who identified a significant growth of tweets with feelings of sadness expressed in the weeks prior to a suicide attempt. Furthermore, a significant increase in tweets with anger emotions were detected the weeks following the suicide attempt.

Several studies advocate the impact of social network reciprocal connectivity on users' suicide ideation. Hsiung [15] observed the users' behavior changes in reaction to a suicide case which happened within the social media group. Jashinsky et al. [16] emphatically highlighted the geographic correlation between the suicide mortality rates and the occurrence of risk factors in tweets. Colombo et al. [17] studied the tweets containing suicide ideation based on the users' behavior in social network interactions resulting in a high degree of reciprocal connectivity and strengthening the bonds between the users.

Another interesting observation is the impact of celebrity suicides on suicide ideation development among the members of online communities. Kumar et al. [9] examined the attributes of suicidal interests of Reddit users related to the copycat or Werther effect [18]. His work indicates a notable increase of users' posting frequency and the shifts in their linguistic behavior after the reports of celebrity suicides. This shift was observed in a direction towards more negative and self-focused posts with lower social integration. Similarly, Ueda et al. [19] conducted profound research on one million Twitter posts following the suicide of 26 prominent celebrities in Japan between the years 2010 and 2014.

Identification of regular language patterns in social media text leads to a more effective recognition of suicidal tendencies. It is often supported by applying various machine learning approaches on different NLP techniques. Desmet et al. [20] built a suicide note analysis method to detect suicide ideation using binary Support Vector Machine (SVM) classifiers. Huang et al. [21] created a psychological lexicon based on a Chinese sentiment dictionary (Hownet). He applied the SVM approach to identify a classification for developing a real-time suicide ideation detection system deployed in Chinese Weibo. Braithwaite et al. [22] demonstrated that machine learning algorithms are efficient in differentiating people to those who are and who are not at suicidal risk. Sueki et al. [23] studied a suicidal intent of Japanese Twitter users in their 20s, where he stated that a language framing is important for identifying suicidal markers in the text. For instance, "want to suicide" expression is more frequently associated with a lifetime suicidal intent than "want to die" expression. O' Dea et al. [24] proved that it is possible to distinguish the level of concern among suicide-related

posts using both human codes and an automatic machine learning classifiers (LR, SVM) on TF-IDF features. Wood et al. [25] identified 125 Twitter users and followed their tweets preceding the data available prior to their suicide attempt. Using simple and linear classifiers, they found 70% of the users with a suicide attempt and identified their gender with 91.9% accuracy. Okhapkina et al. [26] studied the adaptation of information retrieval methods for identifying a destructive informational influence in social networks. He built a dictionary of terms pertaining to a suicidal content. He introduced TF-IDF matrices and singular vector decompositions for them. Sawhney et al. [27] improved the performance of Random Forest (RF) classifier for identification of suicide ideation in tweets. Logistic regression classification algorithms applied in Aladag et al. [28] showed promising results in detecting suicidal content with 80–92% accuracy rate.

With recent advances of neural network models in natural language processing, a new contribution on detection of suicide ideation has emerged from the implementations of more sophisticated deep learning architectures to outperform more traditional machine learning systems. Recurrent neural network (RNN) is well designed for sequence modeling [29]. In particular, long short-term memory (LSTM) is considered to be one of the effective models able to keep useful information from long-range dependency. Sawhney et al. [30] work revealed the strength and ability of C-LSTM-based models as compared to other deep learning and machine learning classifiers for suicide ideation recognition. Ji et al. [31] compared the LSTM classifier with five other machine learning models and demonstrated the feasibility and practicability of the approaches. His study provides one of the major benchmarks for the detection of suicide ideation on Reddit SuicideWatch and Twitter.

Over the recent past, CNN neural networks with convolutional, nonlinear and pooling layers has been successfully applied to a wide range of NLP tasks and has proven to gain better performance than traditional NLP methods [29]. It, however, emphasizes the local n-gram features and prevents capturing long-range interactions. Kalchbrenner et al. [32] advocated the strength of CNN on n-gram features from various sentence positions. Yin and Schutze [33] introduced a multichannel word embedding and unsupervised pre-training model to improve the classification accuracy. Gehrmann et al. [34] used cTAKES and LR approaches with n-gram features to compare the CNN model to more traditional rule-based entity extraction systems. His findings show CNN to outperform other phenol-typing algorithms on the prediction of 10 phenotypes. Morales et al. [35] showed the strength of CNN and LSTM models for a suicide risk assessment presenting the results for a novelly tested personality and tone features. Bhat et al. [36] and [37] highlighted CNN's performance over other approaches to identify the presence of suicidal tendencies among adolescents. Du et al. [38] applied deep learning methods to detect psychiatric stressors for suicide recognition in social media. Using CNN networks, he built a binary classifier to separate suicidal tweets from non-suicidal tweets. Other recent studies [39] revealed positive results of CNN implementations on SuicideWatch forum which serves as a dataset in our research paper.

Fundamentally, single recurrent and convolutional neural networks applied as vectors to encode an entire sequence tend to be insufficient to capture all the important information sequence [40,41]. As a result, there have been several experiments to develop a hybrid framework for coherent combinations of CNNs and RNNs to apply the merits for both. For instance, He et al. [42] introduced a novel neural network model based on a hybrid of ConvNet and BI-LSTMs to solve the measurement problem of a semantic textual similarity. Matsumoto et al. [43] proposed an efficient hybrid model which combines a fast deep model with an initial information retrieval model to effectively and efficiently handle AS. In our study, we propose a framework based on the ensemble of LSTM and CNN combined model to recognize suicide ideation in social media.

3. Datasets

To detect suicide ideation, we train our classification models on a Reddit social media dataset where users can express their opinion via text posts, links or voting mechanism posts. They engage

with each other via comment threads attached to each post [9]. The dataset used in our experiment was built by Ji et al. [31] and consists of a list of suicide-indicative and non-suicidal posts. To preserve the users' privacy, their personal information is replaced with a unique ID. Since the users tend to get engaged in different kinds of subreddits, each group is formed by a corresponding random number of messages derived from various topics. Our dataset is created by 3549 suicide-indicative posts and 3652 non-suicidal posts from relatively large subreddits devoted to support potentially at-risk individuals. Non-suicidal posts originate from subreddits topically related to a family and friends. Table 1 shows the examples from both posts' categories which are topically specific.

Table 1. Examples of Human Annotations of Reddit Posts.

Suicidal Posts	Non-Suicidal Posts
Want to die, end it now, I wanna die with a blunt. I'm going to kill myself this weekend.	There is no one "correct" way to talk to someone struggling with suicidal thoughts, Children of suicide parents.
I want to slit my wrists tonight, I tried to commit suicide.	I think you should tell people how you feel, I think suicide is a permanent option that most of the time results out of a temporary issue.
I wish guns for suicide, nobody cares if I die. I want to die Where can i go to commit Suicide??	Will you ever get over the news that one of your parents committed suicide?
Where can i go to commit suicide?? I don't know what else to do.	Friend has given up, seriously considering suicide.
Just over life, die alone, sleep forever. I'm writing my suicide note right now. I plan to kill myself soon.	National Suicide Prevention online chat. I think suicide is a permanent option that most of the time results out of a temporary issue.
What's the point in living when I will always be alone.	Method used in chris cornell and chester bennington's suicides.

4. Methodology

The purpose of the present study is to implement a combined deep learning classifier to improve a performance of a language modeling and text classification for detecting suicide ideation in Reddit social media. In our experiment, we incorporate a technical description of approaches using various NLP and text to classify techniques.

Figure 1 shows a general overview of our proposed framework. It consists of two directions for text data mining methods. The first one consists of data pre-processing, features extraction with NLP techniques (TF-IDF, BOW and Statistical Features) employed to encode the words to be further proceeded by traditional machine learning systems for the baseline methods. The second framework is created by data pre-processing, features extraction using word embedding, followed by deep learning classifiers, one for the baseline method and one for the proposed model.
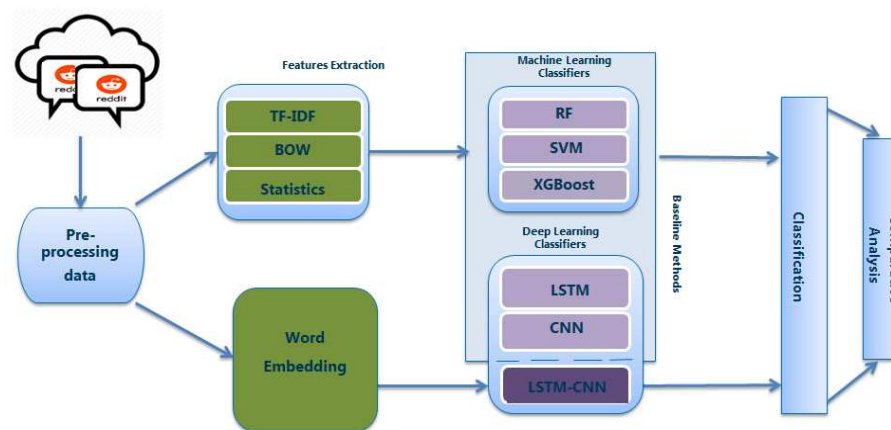
Figure 1. Suicide Ideation Detection Framework.

4.1. Pre-Processing

Pre-processing involves filtering of an input text to improve the accuracy of a proposed method by eliminating redundant features to process raw posts prior to the learning of word embedding. It is achieved by applying a series of filters on Reddit posts to transform raw data into a format understood by learning models. In our study, we employ the Natural language toolkit (NLTK) [44] to pre-process the dataset before it proceeds to its training stage. First, we start with a concatenation of the post titles and bodies. We remove duplicated sentences from the original dataset. Next, we use tokenization as a part of the data filtering and a converting process to divide the Reddit posts into individual tokens. Then, we replace all the URL addresses, contractions and redundant white spaces with a single whitespace. We remove brackets, dashes, colons, stop words and all newline symbols, which could lead to erratic results if stayed ignored. In this way, the posts become lowercased and saved as separate text files. We apply lemmatization to ensure that the word endings will not be roughly dropped, which could lead to creating senseless word pieces such as stemming. We rather transform them into word lemmas related to the dictionary. Finally, the cleaned data is ready for word embedding.

4.2. Proposed Network Model

To detect the presence of suicide ideation in Reddit social media, we combine the strengths of CNN and LSTM neural network architectures and apply a unified LSTM-CNN model for the classification of our chosen text data. The proposed model takes the output vector of the LSTM as the input vector of the CNN. Then it builds a new CNN model on the LSTM to extract the features of the input text sentences and improve the results of the classification accuracy. In our experiment, we follow the Hybrid framework for Text modeling using LSTM-CNN method applied in previous works [45–47].

Figure 2 shows the proposed LSTM-CNN combined model architecture for classifying the sentences with suicidal and non-suicidal content. It is created by the following layers. The first layer is a word embedding layer in which each word in a sentence is assigned a unique index to form a fixed-length vector. It is followed by a dropout layer applied to avoid over-fitting. Next, LSTM layer is added to catch a long-distance dependency across the text with a convolutional layer for performing features extraction. Pooling layer aggregates the information to pool a feature dimension which is later converted into a column vector by a flatten layer. Finally, the neural network process is accomplished with the classification done by a SoftMax function.
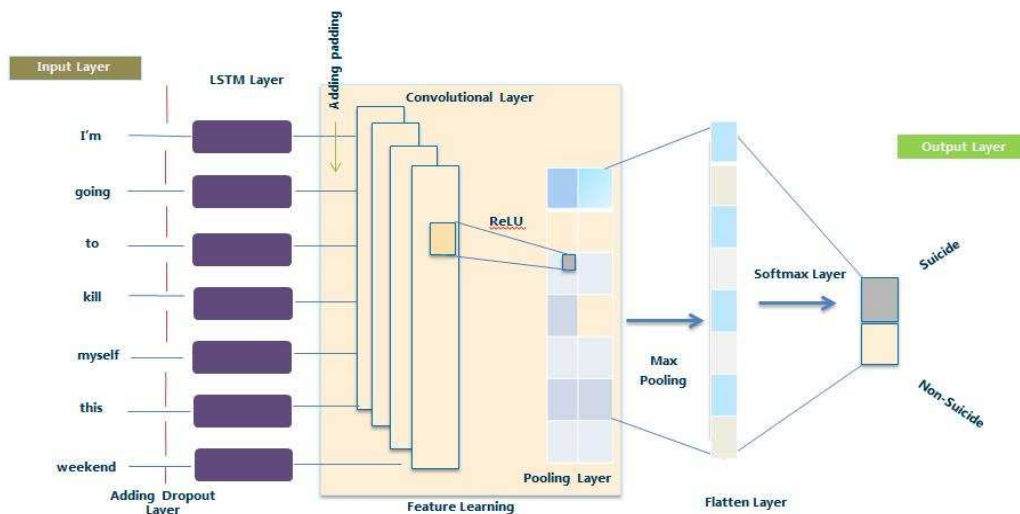
Figure 2. LSTM-CNN Architecture for Suicide Ideation Detection.

Word Embedding Layer

Word embedding is a set of language modeling and feature learning techniques in NLP. It is an input layer of LSTM-CNN combined model which changes the words into a real-valued vector representation. When using the word embedding techniques, the words from the vocabulary tend to map into a particular vector space of real numbers in a low-dimensional space [13]. The models are fundamentally based on an unsupervised training of distributed representations applied for solving supervised tasks [48]. In this section, we employ Word2vec [13] which belongs to a category of shallow models in which two neural layers are trained to reconstruct a word context or current words from their surrounding window of words. When a text is a sequence of words $x_1; x_2; x_3; ...; x_T$, which is converted to low-dimensional word vectors that are characterized by index numbers of embedding layers that transform such indices into d-dimensional of the embedding vector $X_t \in R^d$ through pre-training Word2Vec [13]. In this expression, d stands for the dimension of the word vector with an input text as mentioned in Equation (1):

$$X = [x_1; x_2; x_3; ...; x_T]^{T \times d} \tag{1}$$

At this point, the tth word in the text is expressed by $X_t \in R^d$, where d is a word embedding vector and T length of the text.

Dropout Layer

Dropout layer is used to avoid over-fitting and prevents co-adaptation of hidden units by randomly dropping out the noise in the training data [49]. A rate of 0.5 is employed on the layer to represent the layer's rate parameter, which can balance between 0 and 1 [49]. One of the main dropout layer's characteristics is that it randomly removes or turns off the activation of neurons in embedding layers as the dropout is applied on the layer, whereas each neuron in the embedding layer depicts a dense representation of a word in a sentence [30].

Long Short-Term Memory

Long Short-term Memory (LSTM) belongs to a group of RNN architectures applied in deep learning to classify process and predict time series in sentences. In comparison to RNN, it is not only more robust and able to capture long-term dependencies. It, however, consists of a memory cell that controls the flow to and from each gate. This way, it makes LSTM an excellent choice for identification of suicidal ideation in a social media text. One of LSTM strengths is to prevent vanishing or explosion gradient are often seen in RNN models [30].

In our LSTM layer model, we applied a single layer with 100 LSTM units. In each cell, four independent calculations were performed using four gates. The LSTM layer structure with

input sequences $X = (x_t)$ with a d-dimensional word embedding vector, while H represents the number of LSTM hidden layer nodes [46].

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{2}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{4}$$

$$u_t = tanh(W_u x_t + U_u h_{t-1} + b_u) \tag{5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot U_t \tag{6}$$

$$h_t = o_t \odot tanh(c_t) \tag{7}$$

In above mentioned equations, δ stands for a logistic sigmoid function, while $\odot$ represents element-wise multiplication. W_f and U_f, two weights matrices, and b_f a bias vector are applied for the forget gate f_t which are similar for the input gate i_t, memory cell c_t, tanh layer u_t, output gate o_t and hidden state h_t. Forget gate controls the information sent to the memory cell. This data selection is decided by sigmoid function. Input gate selects which new information will be kept in the memory cell. Memory cell stores the data at each step, and this way ensures long-distance correlations with new input. After the information is updated or ignored through sigmoid layer, tanh layer decides information's level of importance (-1 to 1). These two values are multiplied to update the memory of the new cell state. It is then added to old memory c_{t-1} resulting in c_t [50,51].

In the output gate, the amount of information from the internal memory cell is exposed based on the output cell state which is expressed by the hidden unit h_t at time t which later will be fed to the CNN layer.

Convolutional Layer

Convolutional layer is a part of CNN neural network initially designed for an image recognition with a strong performance ability [52,53]. In recent years, however, CNN has become an incredibly versatile model used for a wide range of multiple text classification tasks with considerable results [32,54,55]. When applying CNN on a well-structured and organized text, the model will discover and learn patterns that would otherwise be lost in a feed-forward network. For instance, a word "down" in the context of "down to earth" and "feeling down" has a different sentiment. In addition, CNN can extract features regardless of where they occur in a sentence [56]. CNN is similar to Feed-forward Neural Networks where the connections between the nodes do not form a cycle. Thus, a single neuron in CNN represents a region within an input sample such as a piece of image or text, in our convolution layer we follow the work by [46].

After each feature sequence is extracted by the LSTM model which is $H = [h_1, h_2, h_3, ..., h_T]^T$ where h_t stands for a m-dimensional feature vector of the tth word in the text sequence where T is the number of LSTM expansion steps equal to the text sequence length. $H \in R^{m \times T}$ is the CNN input matrix with fixed-length inputs; thus, every input length is standardized to T by trimming the longer sentences and padding the shorter sentences with zeros. The convolutional filter is $F \in R^{j \times k}$ where j is the number of the words in the window, k is the dimension of the word embedding vector. The convolutional filter $F = [F_0, F_2, ..., F_{m-1}]$ will generate one value as follows at time step t. Equation (8):

$$O_{F_l} = ReLU[(\sum_{i=0}^{m-1} h_{l+i}^T F_i) + b] \tag{8}$$

where b is a bias, and F and b are the parameters of this single filter. Finally, a feature map is generated on which ReLU activation function is applied to remove non-linearity. Its mathematical expression is as follows:

$$F(x) = max(0, x) \tag{9}$$

In our experiment, we use multiple convolutional filters with various parameter initializations to extract multiple maps from the text [46].

$$P\big(y^{(i)} = j \mid x^{(i)}; \theta\big) = \frac{e^{\theta_j^T x^{(i)}}}{\sum_{K=1}^{K} e^{\theta_j^T x^{(i)}}} \tag{10}$$

Pooling Layer

Pooling layer's function is to minimize a dimensionality of each rectified feature map and retain the most important information. Its characteristic feature is to make the input representations smaller and more manageable aggregating information. It reduces the number of parameters and computations in the network resulting in an ability to control over-fitting [57]. In our study, we use a max pooling operation, which represents the most important information in each feature map.

Flatten Layer

CNN flatten layer aims to transform a pooled feature map into a column vector which makes an input to the neural network of the classification task [47]. As the next step, the pooled feature maps are flattened through a reshape function to make the feature vector pulls concatenated.

$$Flattening = pooled.reshape \tag{11}$$

The above equation takes rows and appends them all to create a single column vector.

Output Layer

Main function of output or fully connected layer is to calculate a probability of suicide and non-suicide text. It uses a text feature vector from a convolutional and pooling layer's output which is followed by considerable activation functions for preventing gradient explosion or vanishing problems [58]. We can apply Sigmoid function [59], SoftMax function [60], Hyperbolic tangent function [61] or Rectified linear unit [62] widely used in classifying an input text into a binary classification based on the labeled training dataset [63]. In our experiment, we apply SoftMax activations on our output layer.

4.3. Baseline

To offer a fair comparative analysis to other competitive models, our experiment is conducted by a performance comparison of the proposed learning model against the baseline models. Handcrafted features (TF-IDF, Bag of Words, Statistical Features) are extracted from the text and fed into four traditional machine learning approaches (Support Vector Machine, Naive Bayes, Random Forest, Extreme Gradient Boosting) and two deep learning models (LSTM, CNN) with Word2vec embedding techniques. We implement machine learning approaches through Scikit-learn [64].

Support Vector Machine (SVM) is a supervised learning model that analyzes data and recognizes the patterns used for classification [65]. It is widely used in a text categorization [66] with good performance results for mental health tasks [67]. In our study, we apply the SVM algorithm to solve the problems that are linearly and non-linearly separable in a lower space by constructing a hyperplane in a high-dimensional space. To evaluate the efficacy of word embeddings, we employ the SVM technique that is proven to work well with concise and categorical data.

Naive Bayes (NB) classifier relies upon an underlying assumption that each feature is independent of another, which vastly simplifies the computational space [68,69]. Together with SVM, it is widely used in a text classification literature and is sufficient for solving practical text categorization problems [66,70]. In our study, we implement the NB algorithm as a probabilistic approach.

Random Forest (RF) is an ensemble technique which combines many weak classifiers into one strong classifier [71]. RF is widely used for binary class classification problems [72].

Extreme Gradient Boosting (XGBoost) is an implementation of gradient boosted decision trees designed for its speed and performance. It is a higher level of boosting algorithm which pushes the

limits of computation on a tree algorithm [73]. In comparison to other gradient boosting machines, XGBoost uses a more regulated model formalization to control over-fitting and provides a better performance [74,75]. To conduct our experiment, we employed a set of NLP representation techniques on our baseline methods. Text Frequency-Inverse Document

Frequency (TF-IDF) is a technique widely used in an information retrieval and text mining field. It measures a frequency of the word occurrence in a text; it selects important words and excludes the words with a low importance for the further text analysis [76,77]. Bag of Words (BOW) is an algorithm that lists the words paired with their word counts per document. The count of each word is used to create a feature vector for a further document summarization [78]. Statistical features [27] are extracted from the posts to encompass the number of tokens, words, sentences and their length.

To compare the proposed method with different variants of deep learning techniques, we use LSTM and CNN that were pre-trained with 300-dimensional word2vec techniques. The output dimension and time steps were set to 300. ADAM optimizer with learning rate 0.0001 was applied to minimize a binary cross-entropy loss and Sigmoid was the activation function for the final output layer. Finally, the model was trained over 20 epochs with a batch size 64 and 512, a dropout rate of 0.5 and ReLU activation function.

The network structure for CNN baseline applied for the text classification is similar to the CNN model proposed by Kim [54].

4.4. Model Architecture and Its Parameters

For the classification task, we train our LSTM-CNN combined model based on its previous implementation. Through the manual testing, we conduct a fine-tuning with 10-fold cross-validation. We apply a pre-trained word2vec model which was trained on 100 billion words from Google News for features classification. A one-dimensional convolutional neural network is initialized with a 300-dimensional pre-trained word2vec [13,79].

Table 2 presents a parameter setting for the proposed model (LSTM + CNN). The experiment is conducted using different parameters listed as follows: the parameter, namely number of filter, kernel size, padding, pooling size, optimizer, batch size, epochs and units. We use Python with NLTK natural language toolkit. The models are built by Tensorflow deep learning framework, and the experimental environment is trained on NVIDIA GTX 1080 in a 64-bit computer with Intel(R) Core(TM) i7- 6700 CPU @3.4GHz, 16 GB RAM and Ubuntu 16.04 operating system.

Table 2. Parameter Setting Regarding Proposed LSTM-CNN Model.

LSTM—CNN Model Layers	Parameters	Values
Convolutional layer	Number of filters	2, 4, 6, 8
	Kernel sizes	2, 3, 4
	Padding	'Same'
	Activation function	'ReLU'
Pooling layer	Pooling size	Max-Pooling
LSTM layer and other	Units	100
	Embedding dimension	300
	Batch size	8
	Number of epochs	10
	Dropout	0.5
	Fully connected layer	SoftMax

4.5. Evaluation Metrics

To evaluate the baseline with our proposed deep learning classification technique, we use evaluation metrics, such as accuracy of estimations (Acc.) Equation (12) and F-score (F1) Equation (15), consisting of precision (P) and recall (R) . It relies on a confusion matrix incorporating the information about each test sample prediction outcome. Accuracy is the rate of a correct classification;

F1 Equation (15) score is a harmonic average of precision and recall; precision estimates the number of positively identified samples; recall approximates the proportion of correctly identified positive samples. The closer the both values are, the higher the F1 score is. In the evaluation metrics, we find number of true positive predictions (TP), true negative predictions (TN), false-positive predictions (FP) and false-negative predictions (FN) [80]. The most straightforward classifying evaluation score is an accuracy defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

$$F_1 = 2.\frac{precision.recall}{precision + recall} \tag{15}$$

5. Experimental Results

We perform our results in two main phases. We begin by examining the data analysis results in the entire labeled corpus of Reddit posts. First, we analyze the most frequent n-grams in suicide-indicative posts linked with suicidal intents, and compare them with the n-grams in non-suicidal posts. Next, to measure the signs of suicidal thoughts, we use our proposed set of features and compare the performance of our proposed deep learning classifier with the baselines in terms of evaluation metrics.

5.1. Data Analysis Results

To compare dissimilarities in the lexicon, we examine the entire dataset to investigate the presence of suicidal thoughts. We compute the frequencies of all the unigrams and bigrams in both suicide-indicative posts and non-suicidal posts. We select the top 200 unigrams and bigrams from each category to examine their nature and connection with suicide ideation. We use a visual support of the word cloud. We support the analysis with top 20 most frequent n-grams in both dataset categories. Figures 3 and 4 present the top 200 unigrams and bigrams for both datasets generated in our experiment. Emerged words indicated by a high frequency are illustrated in both figures. Examining the posts from the SuicideWatch forum, we identify the features with a suicidal intent align with the findings supported in suicide literature [2]. Specifically, we observe evidence of manifestation of hopelessness and frustration ("fucking life", "tired living", "hate tired", "I'm tired"), anxiety ("I'm afraid"), sense of guilt ("I am sorry"), regret ("never again"), signs of loneliness ("no friend"). Concerning the mental focus of the users, we identify the self-oriented references and attention turned towards themselves ("I'm", "I'm not", "I've never"). This result is supported by [81]. After that, we detect the user's tendency for the preoccupation with their feelings ("feel like", "make feel"), strengthened by the words of negation ("no one", "anymore", "I've never", "would never"). In the suicide-indicative posts, we observe quite a high frequency of question marks ("Concerned but don't know what to say?", "Why is mankind afraid of death?"). It might originate in a frequent usage of rhetorical questions to emphasize the ideas consciously and intensify the sentiment [11].

Figure 3. Suicide Ideation Detection Framework.

Figure 4. Non-Suicide Ideation Detection Framework.

Another interesting observation is found in users' depiction of suicidal tendencies. It is mostly expressed by the words with death connotations ("suicide", "want die", "die fucking", "suicide wish", "wish die", "want kill", "want end", "want go"). Furthermore, sense of urgency and a manifestation of hopelessness is also visible ("hope help", "life hope", "help ended").

In contrast to the suicide-indicative posts, the unigrams and bigrams examined in the non-suicidal posts contain predominantly the words describing happy moments, positive attitude and feelings ("want joke", "want fun", "go out", "laugh thought", "want happy"). The users have a tendency to strive towards maintaining positive spirits ("get better"). They often mention social relations activities ("best friend", "high school") or express their feelings ("make feel", "feel like").

5.2. Classification Analysis Results

After the n-grams frequency analysis, we evaluate the experimental approach based on six baseline methods and a proposed model. Our main task is to detect suicide ideation from the chosen data. In our baseline, we use three single handcrafted features, such as TF-IDF, Bag of Words, Statistical Features and their combinations which are applied on SVM, NB, RF and XGBoost models. The main aim of combining the distinct NLP techniques is to examine which features best favor the performance accuracy for suicide ideation. Next, we apply a word2vec technique on LSTM and CNN model. We conduct the performance evaluation through three different approaches. First, we analyze the performance of machine learning and deep learning models in the baseline. Second, we compare all the classification methods within the baseline. Next, we compare the proposed LSTM-CNN model with the baseline. Finally, we make a brief comparative analysis of the handcrafted features in the classification models.

Table 3 shows the results of the baseline and proposed model on suicide ideation detection tasks in terms of evaluation metrics. The first six rows show the results for the baseline. The last row with LSTM-CNN demonstrates the proposed model. Each classified corpus contains an accuracy, F-measure, recall and precision result value.

Evaluating the performance of machine learning methods in the baseline, we observe the performance of XGBoost scoring higher than other traditional text classification approaches with both combined and single features, excluding the Statistics. Considering all the baseline classification methods, LSTM deep learning classifier outperforms other baseline approaches with its performance improvement reported in 91.7% accuracy and 92.6% F1 score.

Comparing our proposed model with the whole baseline, we can conclude that the best relevance classification in our experiment is achieved with LSTM-CNN combined model by using word embedding. Based on the optimized parameters, it significantly outperforms other algorithms reaching 93.8% accuracy and 92.8% F1 score. Our results show that the proposed combined neural network model performs better in comparison to single LSTM and CNN classifiers. Concerning the accuracy performance, our results outperform the accuracy of the experiment previously applied on the same dataset [31].

Considering the impact of single handcrafted features (Statistics, TF-IDF, BOW) on the performance of our three machine learning classifiers, we can observe that using Statistical features in SVM we achieve 79.6% accuracy which is the highest performance results. Next is TF-IDF in XGBoost with 85.6% accuracy and BOW with 83.1% accuracy. Combined handcrafted features (Statistics+TF-IDF+BOW) in XGBoost score the highest with 88.3% accuracy. This result is comparable with the word embedding neural network features in LSTM (91.7%) and CNN model (90.6%) respectively.

Table 3. Performance Results of the Classification Models.

Methods	Feature Type	Acc.	F1-Score	Recall	Precision
RF	Statistics	77.2	75.1	73.9	76.3
	TF-IDF	81.8	80.9	83.4	80.5
	Bag of Words	81.1	78.6	77.9	81.1
	Statistics + TF IDF+ Bag of Words	85.6	84.1	84	85
SVM	Statistics	79.6	79	70	60
	TF-IDF	81.2	82.7	87.2	78.7
	Bag of Words	80.6	81.1	81.8	80.4
	Statistics + TF IDF+ Bag of Words	83.5	83.8	85.5	82.1
NB	Statistics	68.2	71.3	76.3	67.6
	TF-IDF	78.6	76.1	75.6	80.5
	Bag of Words	79.8	78.4	78.9	79.7
	Statistics + TF IDF+ Bag of Words	82.5	81.5	83.4	80.8
XGBOOST	Statistics	76.3	76.1	75.6	80.5
	TF-IDF	85.6	84.1	84.0	85.8
	Bag of Words	83.1	82.6	84.4	81.6
	Statistics + TF ID F+ Bag of Words	88.3	83.1	84.3	88.4
LSTM	Word2vec	91.7	92.6	90.5	94.8
CNN		90.6	92.8	93.8	91.8
LSTM-CNN		93.8	93.4	94.1	93.2

Acc. represents accuracy; RF = Random Forest, SVM = Support Vector Machine, NB = Naive Bayes, XGBoost = Extreme Gradient Boosting, LSTM = Long Short-term Memory, CNN = Convolutional Neural Networks.

For the models' parameters optimization, we believe it is advisable to perform a coarse line search over a single region to find the most suitable size for the dataset. After that, we are ready to do the exploration of the most proper combinations of different filter region sizes surrounding the optimal size. Based on our findings, the main effect of the filter region sizes is in keeping the number of feature maps of individual region size at a fixed value [82]. Additionally, considering the ReLU activation function, ReLU offers a faster and better performance as well as the generalization, which was similarly applied in the works of [83,84]. Since ReLU represents a nearly linear function, it preserves the properties of linear models that make it easier to optimize with gradient descent methods [60]. Our pooling strategy indicates that a max-pooling achieves a better performance than other alternative strategies for the classification tasks. It could be caused by a low importance of

the location of predictive contexts and higher prediction abilities of certain n-grams in a sentence in contrast to the entire sentence [85].

6. Conclusions and Future Work

The integration of deep learning methods into suicide care offers new directions for the improvement of detection of suicide ideation and the possibility for early suicide prevention. Our work takes part in this journey towards the technological improvement in convolutional linguistics to be shared within the research community and successfully implemented in mental health care.

In our study, we presented an approach to recognize the existence of suicide ideation signs in Reddit social media and focused on detecting the most effective performance improvement solutions. For such purpose, we built our system on subreddit data corpus created by suicide-indicative and non-suicidal posts. We used different data representation techniques to reformulate the text of the posts into the presentation that our system can recognize. In particular, we characterized a closer connection between the suicidal thoughts and language usage by applying various NLP and text classification techniques. We described the experiment with LSTM-CNN networks built on the top of word2vec features, and observed the potential of CNN in multiple texts classification tasks.

Based on our experiment, the proposed LSTM-CNN hybrid model considerably improves the accuracy of text classification. The main reason the model outperforms other machine learning classifiers is that it combines the strengths of both LSTM and CNN algorithms, and makes up their shortcomings. First, it takes advantage of the LSTM to maintain context information in a long text by keeping the previous tokens and resolves the problem of vanishing gradient. Second, it uses the CNN layer to extract the local pattern using the richer representation of the original input of the text and able to process the text considering not only single words but also their combinations of different predefined sizes trying to learn their best combinations and interpretations. Using this approach, we can ensure that the hybrid model can effectively improve the prediction results as we try to prove in our experiment.

Our aim was not to explore the detailed sensitivity of CNN hyper-parameters with respect to the designed decisions. However, we rather tried to improve the potential of CNN neural network classifier for suicide ideation tasks. During our data analysis, we identified the features with the depictions of suicidal tendencies. We observed a considerable shift in the language usage of at-risk individuals. The signs of frustration, hopelessness, negativity or loneliness were significantly detected accompanied by users' preoccupation with themselves.

According to our comparative evaluation, we specifically demonstrated the strength and potential of CNN. It resulted in the highest performance among other classification approaches chosen for our experiment, including LSTM as an artificial recurrent neural network. Through the hyper-parameter optimization, we were able to achieve an improvement based on the adaptive hyper-parameter tuning.

Although our research findings show that the performances of applied classification approaches are reasonably good, the absolute value of the metrics indicates that this is a challenging task worthy of further study. In our future work, we might try to access a larger dataset with suicide ideation content and a new dataset with related topics. We might examine the correlation between suicidal ideation and family environment, weather, etc. Both datasets will be collected from different social media sources for further demonstration and comparison with our proposed hybrid model. In addition, the performance of the datasets will be applied for further investigation with other deep learning classifiers, such as C-LSTM, RNN and their combined models, accompanied by various parameter optimization evaluations.

Limitation of our experiment can be found in its data deficiency and annotation bias. Data deficiency is one of the most critical issues of current research [86], where mainly supervised learning techniques are applied. They usually require a manual annotation. However, there are not enough annotated data to support further research. Another issue is the annotation bias caused by

manual labeling with some predefined annotation rules. In some cases, the annotation may lead to bias of labels resulting in misleading evidence to confirm the suicide action of the authors.

We believe that our study can contribute to future machine learning research for building an easily accessible and highly effective suicide detection and reporting system implemented in social media networks as an efficient intervention point between at-risk individuals and mental health services.

References

1. World Health Organization. *National Suicide Prevention Strategies: Progress, Examples and Indicators*; World Health Organization: Geneva, Switzerland, 2018.
2. Beck, A.T.; Kovacs, M.; Weissman, A. Hopelessness and suicidal behavior: An overview. *JAMA* **1975**, *234*, 1146–1149. [CrossRef] [PubMed]
3. Silver, M.A.; Bohnert, M.; Beck, A.T.; Marcus, D. Relation of depression of attempted suicide and seriousness of intent. *Arch. Gen. Psychiatry* **1971**, *25*, 573–576. [CrossRef] [PubMed]
4. Klonsky, E.D.; May, A.M. Differentiating suicide attempters from suicide ideators: A critical frontier for suicidology research. *Suicide Life-Threat. Behav.* **2014**, *44*, 1–5. [CrossRef] [PubMed]
5. Pompili, M.; Innamorati, M.; Di Vittorio, C.; Sher, L.; Girardi, P.; Amore, M. Sociodemographic and clinical differences between suicide ideators and attempters: A study of mood disordered patients 50 years and older. *Suicide Life-Threat. Behav.* **2014**, *44*, 34–45. [CrossRef] [PubMed]
6. DeJong, T.M.; Overholser, J.C.; Stockmeier, C.A. Apples to oranges?: A direct comparison between suicide attempters and suicide completers. *J. Affect. Disord.* **2010**, *124*, 90–97. [CrossRef] [PubMed]
7. De Choudhury, M.; Kiciman, E.; Dredze, M.; Coppersmith, G.; Kumar, M. Discovering shifts to suicidal ideation from mental health content in social media. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, San José, CA, USA, 9–12 December 2016; ACM: New York, NY, USA, 2016; pp. 2098–2110.
8. Marks, M. Artificial Intelligence Based Suicide Prediction. *Yale J. Health Policy Law Ethics* **2019**, Forthcoming.
9. Kumar, M.; Dredze, M.; Coppersmith, G.; De Choudhury, M. Detecting changes in suicide content manifested in social media following celebrity suicides. In Proceedings of the 26th ACM conference on Hypertext & Social Media, Prague, Czech Republic, 4–7 July 2015; ACM: New York, NY, USA, 2015; pp. 85–94.
10. Ji, S.; Long, G.; Pan, S.; Zhu, T.; Jiang, J.; Wang, S. Detecting Suicidal Ideation with Data Protection in Online Communities. In Proceedings of the International Conference on Database Systems for Advanced Applications, Chiang Mai, Thailand, 22–25 April 2019; Springer: Berlin, Germany, 2019; pp. 225–229.
11. Yang, Y.; Zheng, L.; Zhang, J.; Cui, Q.; Li, Z.; Yu, P.S. TI-CNN: Convolutional neural networks for fake news detection. *arXiv* **2018**, arXiv:1806.00749.
12. Mikolov, T.; Karafiát, M.; Burget, L.; Černockỳ, J.; Khudanpur, S. Recurrent neural network based language model. In Proceedings of the Eleventh Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, 26–30 September 2010.
13. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, CA, USA, 5–10 December 2013; pp. 3111–3119.

14. Coppersmith, G.; Ngo, K.; Leary, R.; Wood, A. Exploratory analysis of social media prior to a suicide attempt. In Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology, San Diego, CA, USA, 16 June 2016; pp. 106–117.

15. Hsiung, R.C. A suicide in an online mental health support group: Reactions of the group members, administrative responses, and recommendations. *CyberPsychol. Behav.* **2007**, *10*, 495–500. [CrossRef]

16. Jashinsky, J.; Burton, S.H.; Hanson, C.L.; West, J.; Giraud-Carrier, C.; Barnes, M.D.; Argyle, T. Tracking suicide risk factors through Twitter in the US. *Crisis* **2014**, *35*, 51–59. [CrossRef]

17. Colombo, G.B.; Burnap, P.; Hodorog, A.; Scourfield, J. Analysing the connectivity and communication of suicidal users on twitter. *Comput. Commun.* **2016**, *73*, 291–300. [CrossRef]

18. Niederkrotenthaler, T.; Till, B.; Kapusta, N.D.; Voracek, M.; Dervic, K.; Sonneck, G. Copycat effects after media reports on suicide: A population-based ecologic study. *Soc. Sci. Med.* **2009**, *69*, 1085–1090. [CrossRef] [PubMed]

19. Ueda, M.; Mori, K.; Matsubayashi, T.; Sawada, Y. Tweeting celebrity suicides: Users' reaction to prominent suicide deaths on Twitter and subsequent increases in actual suicides. *Soc. Sci. Med.* **2017**, *189*, 158–166. [CrossRef] [PubMed]

20. Desmet, B.; Hoste, V. Emotion detection in suicide notes. *Expert Syst. Appl.* **2013**, *40*, 6351–6358. [CrossRef]

21. Huang, X.; Zhang, L.; Chiu, D.; Liu, T.; Li, X.; Zhu, T. Detecting suicidal ideation in Chinese microblogs with psychological lexicons. In Proceedings of the 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, Bali, Indonesia, 9–12 December 2014; pp. 844–849.

22. Braithwaite, S.R.; Giraud-Carrier, C.; West, J.; Barnes, M.D.; Hanson, C.L. Validating machine learning algorithms for Twitter data against established measures of suicidality. *JMIR Ment. Health* **2016**, *3*, e21. [CrossRef]

23. Sueki, H. The association of suicide-related Twitter use with suicidal behaviour: A cross-sectional study of young internet users in Japan. *J. Affect. Disord.* **2015**, *170*, 155–160. [CrossRef]

24. O'Dea, B.; Wan, S.; Batterham, P.J.; Calear, A.L.; Paris, C.; Christensen, H. Detecting suicidality on Twitter. *Internet Interv.* **2015**, *2*, 183–188. [CrossRef]

25. Wood, A.; Shiffman, J.; Leary, R.; Coppersmith, G. Language signals preceding suicide attempts. In Proceedings of the CHI 2016 Computing and Mental Health Workshop, San Jose, CA, USA, 7–12 May 2016.

26. Okhapkina, E.; Okhapkin, V.; Kazarin, O. Adaptation of information retrieval methods for identifying of destructive informational influence in social networks. In Proceedings of the 2017 IEEE 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA), Taipei, Taiwan, 27–29 March 2017; pp. 87–92.

27. Sawhney, R.; Manchanda, P.; Singh, R.; Aggarwal, S. A computational approach to feature extraction for identification of suicidal ideation in tweets. In Proceedings of the ACL 2018, Student Research Workshop, Melbourne, Australia, 15–20 July 2018; pp. 91–98.

28. Aladağ, A.E.; Muderrisoglu, S.; Akbas, N.B.; Zahmacioglu, O.; Bingol, H.O. Detecting suicidal ideation on forums: Proof-of-Concept study. *J. Med. Internet Res.* **2018**, *20*, e215. [CrossRef]

29. Wang, C.; Jiang, F.; Yang, H. A hybrid framework for text modeling with convolutional rnn. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; ACM: New York, NY, USA, 2017; pp. 2061–2069.

30. 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, Brussels, Belgium, 31 October–1 November 2018; pp. 167–175.

31. Ji, S.; Yu, C.P.; Fung, S.f.; Pan, S.; Long, G. Supervised learning for suicidal ideation detection in online user content. *Complexity* **2018**, *2018*, 6157249. [CrossRef]

32. Kalchbrenner, N.; Grefenstette, E.; Blunsom, P. A convolutional neural network for modelling sentences. *arXiv* **2014**, arXiv:1404.2188.

33. Yin, W.; Schütze, H. Multichannel variable-size convolution for sentence classification. *arXiv* **2016**, arXiv:1603.04513.

34. Gehrmann, S.; Dernoncourt, F.; Li, Y.; Carlson, E.T.; Wu, J.T.; Welt, J.; Foote, J., Jr.; Moseley, E.T.; Grant, D.W.; Tyler, P.D.; et al. Comparing deep learning and concept extraction based methods for patient phenotyping from clinical narratives. *PLoS ONE* **2018**, *13*, e0192360. [CrossRef] [PubMed]

35. Morales, M.; Dey, P.; Theisen, T.; Belitz, D.; Chernova, N. An investigation of deep learning systems for suicide risk assessment. In Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology, Minneapolis, MN, USA, 6 June 2019; pp. 177–181.

36. Bhat, H.S.; Goldman-Mellor, S.J. Predicting Adolescent Suicide Attempts with Neural Networks. *arXiv* **2017**, arXiv:1711.10057.

37. Gaur, M.; Alambo, A.; Sain, J.P.; Kursuncu, U.; Thirunarayan, K.; Kavuluru, R.; Sheth, A.; Welton, R.; Pathak, J. Knowledge-aware assessment of severity of suicide risk for early intervention. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; ACM: New York, NY, USA, 2019; pp. 514–525.

38. Du, J.; Zhang, Y.; Luo, J.; Jia, Y.; Wei, Q.; Tao, C.; Xu, H. Extracting psychiatric stressors for suicide from social media using deep learning. *BMC Med. Inform. Decis. Mak.* **2018**, *18*, 43. [CrossRef] [PubMed]

39. Yao, H.; Rosenthal, R.W.F. Detection of Suicidality among Opioid Users on Reddit: A Machine Learning Based Approach. *J. Med. Internet Res.* **2019**. [CrossRef]

40. Hermann, K.M.; Kocisky, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; Blunsom, P. Teaching machines to read and comprehend. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 1693–1701.

41. Hill, F.; Bordes, A.; Chopra, S.; Weston, J. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv* **2015**, arXiv:1511.02301.

42. He, H.; Lin, J. Pairwise word interaction modeling with deep neural networks for semantic similarity measurement. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 937–948.

43. Maziarz, M.; Piasecki, M.; Rudnicka, E.; Szpakowicz, S.; Kędzia, P. plWordNet 3.0—A Comprehensive Lexical-Semantic Resource. In Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics, Osaka, Japan, 11–16 December 2016; pp. 2259–2268.

44. Bird, S.; Klein, E.; Loper, E. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2009.

45. Sosa, P.M. Twitter Sentiment Analysis Using Combined LSTM-CNN Models. 2017. Available online: https://www.academia.edu/35947062/Twitter/_Sentiment/_Analysis/_using/_combined/_LSTM-CNN/_Models (accessed on 23 December 2019).

46. Zhang, J.; Li, Y.; Tian, J.; Li, T. LSTM-CNN Hybrid Model for Text Classification. In Proceedings of the 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 12–14 October 2018; pp. 1675–1680.

47. Ahmad, S.; Asghar, M.Z.; Alotaibi, F.M.; Awan, I. Detection and classification of social media-based extremist affiliations using sentiment analysis techniques. *Hum.-Centric Comput. Inf. Sci.* **2019**, *9*, 24. [CrossRef]

48. Orabi, A.H.; Buddhitha, P.; Orabi, M.H.; Inkpen, D. Deep learning for depression detection of twitter users. In Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic, New Orleans, LA, USA, 5 June 2018; pp. 88–97.

49. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

50. Yan, S. Understanding LSTM Networks. 2015. Available online: https://colah.github.io/posts/2015-08-Understanding-LSTMs/ (accessed on 16 July 2019).

51. Olah, C. Yan, S. Understanding LSTM and Its Diagrams. MLReview.com 2016. Available online: https://medium.com/mlreview/understanding-lstm-and-its-diagrams-37e2f46f1714 (accessed on 16 July 2019).

52. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [CrossRef]

53. Srinivas, S.; Sarvadevabhatla, R.K.; Mopuri, K.R.; Prabhu, N.; Kruthiventi, S.S.; Babu, R.V. A taxonomy of deep convolutional neural nets for computer vision. *Front. Robot. AI* **2016**, *2*, 36. [CrossRef]

54. Kim, Y. Convolutional neural networks for sentence classification. *arXiv* **2014**, arXiv:1408.5882.

55. Zhang, X.; Zhao, J.; LeCun, Y. Character-level convolutional networks for text classification. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada 7–12 December 2015; pp. 649–657.

56. Sosa, P.M.; Sadigh, S. Twitter Sentiment Analysis with Neural Networks. 2016. Available online: https://www.academia.edu/35947062/Twitter_Sentiment_Analysis_using_combined_LSTM-CNN_Models (accessed on 23 December 2019).

57. Xu, B.; Wang, N.; Chen, T.; Li, M. Empirical evaluation of rectified activations in convolutional network. *arXiv* **2015**, arXiv:1505.00853.

58. Hochreiter, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Uncertain. Fuzziness -Knowl.-Based Syst.* **1998**, *6*, 107–116. [CrossRef]

59. Norouzi, M.; Ranjbar, M.; Mori, G. Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 2735–2742.

60. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; The MIT Press: Cambridge, MA, USA, 2016.

61. Nguyen, D.; Widrow, B. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In Proceedings of the 1990 IJCNN IEEE International Joint Conference on Neural Networks, San Diego, CA, USA, 17–21 June 1990; pp. 21–26.

62. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning (ICML-10), Haifa, Israel, 21–24 June 2010; pp. 807–814.

63. Garcia, B.; Viesca, S.A. Real-time american sign language recognition with convolutional neural networks. In *Convolutional Neural Networks for Visual Recognition*; Stanford University: Stanford, CA, USA, 2016; pp. 225–232.

64. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

65. Schütze, H.; Manning, C.D.; Raghavan, P. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, UK, 2008; Volume 39, pp. 293–311.

66. Joachims, T. Text categorization with support vector machines: Learning with many relevant features. In *Lecture Notes in Computer Science, Proceedings of the European Conference on Machine Learning, Chemnitz, Germany, 21–23 April 1998*; Springer: Berlin, Germany, 1998; pp. 137–142.

67. De Choudhury, M.; Gamon, M.; Counts, S.; Horvitz, E. Predicting depression via social media. In Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media, Bosten, MA, USA, 8–11 July 2013.

68. Rish, I. An empirical study of the naive Bayes classifier. In Proceedings of the IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, New York, NY, USA, 4–6 August 2001; Volume 3, pp. 41–46.

69. McCallum, A.; Nigam, K. A comparison of event models for naive bayes text classification. In Proceedings of the AAAI-98 Workshop on Learning for Text Categorization, Madison, WI, USA, 26–27 July 1998; Volume 752, pp. 41–48.

70. Sebastiani, F. Machine learning in automated text categorization. *ACM Comput. Surv. (CSUR)* **2002**, *34*, 1–47. [CrossRef]

71. Freund, Y.; Schapire, R.; Abe, N. A short introduction to boosting. *J.-Jpn. Soc. Artif. Intell.* **1999**, *14*, 1612.

72. Schapire, R.E.; Singer, Y.; Singhal, A. Boosting and Rocchio applied to text filtering. In Proceedings of the SIGIR, Melbourne, Australia 24–28 August 1998; Volume 98, pp. 215–223.

73. Brownlee, J. *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python*; Machine Learning Mastery: New York, NY, USA, 2018.

74. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]

75. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; ACM: New York, NY, USA, 2016; pp. 785–794.

76. Ikonomakis, M.; Kotsiantis, S.; Tampakas, V. Text classification using machine learning techniques. *WSEAS Trans. Comput.* **2005**, *4*, 966–974.

77. Wang, Z.; Qian, X. Text categorization based on LDA and SVM. In Proceedings of the 2008 IEEE International Conference on Computer Science and Software Engineering, Wuhan, China, 12–14 December 2008; Volume 1, pp. 674–677.

78. Fiori, A. *Innovative Document Summarization Techniques: Revolutionizing Knowledge Understanding: Revolutionizing Knowledge Understanding*; IGI Global: Hershey, PA, USA, 2014.

79. Collobert, R.; Weston, J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5–9 July 2008; ACM: New York, NY, USA, 2008; pp. 160–167.

80. Basu, T.; Murthy, C. A feature selection method for improved document classification. In Proceedings of the International Conference on Advanced Data Mining and Applications, Nanjing, China 15–18 December 2012; Springer: Berlin, Germany, 2012; pp. 296–305.

81. Stirman, S.W.; Pennebaker, J.W. Word use in the poetry of suicidal and nonsuicidal poets. *Psychosom. Med.* **2001**, *63*, 517–522. [CrossRef]

82. Zhang, Y.; Wallace, B. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv* **2015**, arXiv:1510.03820.

83. Zeiler, M.D.; Ranzato, M.; Monga, R.; Mao, M.; Yang, K.; Le, Q.V.; Nguyen, P.; Senior, A.; Vanhoucke, V.; Dean, J.; et al. On rectified linear units for speech processing. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 3517–3521.

84. Dahl, G.E.; Sainath, T.N.; Hinton, G.E. Improving deep neural networks for LVCSR using rectified linear units and dropout. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 8609–8613.

85. Vu, N.T.; Adel, H.; Gupta, P.; Schütze, H. Combining recurrent and convolutional neural networks for relation classification. *arXiv* **2016**, arXiv:1605.07333.

86. Ji, S.; Pan, S.; Li, X.; Cambria, E.; Long, G.; Huang, Z. Suicidal Ideation Detection: A Review of Machine Learning Methods and Applications. *arXiv* **2019**, arXiv:1910.12611.