Deep Learning for Depression Detection of Twitter Users

Conference Paper - January 2018
DOI: 10.18653/v1/W18-0609

4 authors, including:

Ahmed Mohamed Husseini Orabi
University of Ottawa
15 PUBLICATIONS 53 CITATIONS
See Profile

Mahmoud Mohamed Husseini Orabi
12 PUBLICATIONS 39 CITATIONS
See Profile

Diana Inkpen
University of Ottawa
167 PUBLICATIONS 3,153 CITATIONS
See Profile

Some of the authors of this publication are also working on these related projects:

- Metaphor Detection in a Poetry Corpus View project
- Deep learning for Information Extraction View project

All content following this page was uploaded by Ahmed Mohamed Husseini Orabi on 23 March 2019.
The user has requested enhancement of the downloaded file.
Deep Learning for Depression Detection of Twitter Users

Ahmed Husseini Orabi, Prasadith Buddhitha, Mahmoud Husseini Orabi, Diana Inkpen
School of Electrical Engineering and Computer Science
University of Ottawa, Ottawa, ON K1N 6N5 Canada
{ahuss045, pkiri056, mhuss092, diana.inkpen}@uottawa.ca

Abstract

Mental illness detection in social media can be considered a complex task, mainly due to the complicated nature of mental disorders. In recent years, this research area has started to evolve with the continuous increase in popularity of social media platforms that became an integral part of people’s life. This close relationship between social media platforms and their users has made these platforms to reflect the users’ personal life on many levels. In such an environment, researchers are presented with a wealth of information regarding one’s life. In addition to the level of complexity in identifying mental illnesses through social media platforms, adopting supervised machine learning approaches such as deep neural networks have not been widely accepted due to the difficulties in obtaining sufficient amounts of annotated training data. Due to these reasons, we try to identify the most effective deep neural network architecture among a few of selected architectures that were successfully used in natural language processing tasks. The chosen architectures are used to detect users with signs of mental illnesses (depression in our case) given limited unstructured text data extracted from the Twitter social media platform.

1 Introduction

Mental disorder is defined as a "syndrome characterized by a clinically significant disturbance in an individual’s cognition, emotion regulation, or behavior that reflects a dysfunction in the psychological, biological, or developmental processes underlying mental functioning" (American Psychiatric Association, 2013). According to Canadian Mental Health Association (2016), 20% of Canadians belonging to different demographics have experienced mental illnesses during their lifetime, and around 8% of adults have gone through a major depression. According to World Health Organization (2014) statistics, nearly 20% of children and adolescents have experienced mental illnesses and half of these mental illnesses start before the age of 14. In addition, around 23% of deaths in the world were caused due to mental and substance use disorders. The broad implication of mental illness can be identified from the level of suicide in Canada where nearly 4,000 Canadians have died from suicide and 90% of them were identified as having some form of a mental disorder (Mental Health Commission of Canada, 2016). Apart from the severity of mental disorders and their influence on one’s mental and physical health, the social stigma (e.g., "mental disorders cannot be cured") or discrimination has made the individuals to be neglected by the community as well as to avoid taking the necessary treatments.

The inherent complexity of detecting mental disorders using social media platforms can be seen in the literature, where many researchers have tried to identify key indicators utilizing different natural language processing approaches. To extract the most prominent features to develop an accurate predictive model, one must acquire a sufficient amount of knowledge related to the particular area of research. Even if such features were extracted, this does not assure that those features are the key contributors to obtaining improved accuracies. Due to these reasons, we investigate the possibility of using deep neural architectures because the features are learned within the architecture itself.

Here, we explore a few selected deep neural network architectures to detect mental disorders, specifically depression. We used the data released for the Computational Linguistics and Clinical Psychology (CLPsych) 2015 shared task (Coppersmith et al., 2015b). Even though the task is comprised of three subtasks: detecting Post-Traumatic
Stress Disorder (PTSD) vs. control, depression vs. control and PTSD vs. depression, our primary objective was to detect depression using the most effective deep neural architecture from two of the most popular deep learning approaches in the field of natural language processing: Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), given the limited amount (i.e., in comparison to most of the deep neural network architectures) of unstructured data.

Our approach and key contributions can be summarized as follows.

- Word embedding optimization: we propose a novel approach to optimize word-embedding for classification with a focus on identifying users suffering from depression based on their social posts such as tweets. We use our approach to improve the performance of two tasks: depression detection on the CLPsych2015 dataset and test generalization capability on the Bell Lets Talk dataset (Jamil et al., 2017).

- Comparative evaluation: we investigate and report the performance of several deep learning architectures commonly used in NLP tasks, in particular, to detect mental disorders. We also expand our investigation to include different word embeddings and hyperparameter tuning.

2 Mental illness detection

Identifying the treatment requirement for a mental disorder is a complicated clinical decision, which involves several factors such as the severity of symptoms, patients’ suffering associated with the symptoms, positive and negative outcomes of particular treatments, disabilities related to patients’ symptoms, and symptoms that could negatively impact other illnesses (American Psychiatric Association, 2013). It is also important to note that measuring the severity of the disorder is also a difficult task that could only be done by a highly trained professional with the use of different techniques such as text descriptions and clinical interviews, as well as their judgments (American Psychiatric Association, 2013). Considering the complexity of the procedures and level of skills involved in identifying mental disorder and the necessary treatments, detecting mental illness within social media using web mining and emotion analysis techniques could be considered a preliminary step that could be used to generate awareness.

It is of greater concern to respect ethical facets about the use of social media data and its privacy. The researchers working with such social media data must take the necessary precautions to protect the privacy of users and their ethical rights to avoid further psychological distress. Certain researchers have taken adequate steps in anonymizing the data to secure user privacy. Coppersmith et al. (2015b) have used a whitelist approach in anonymizing the data given to the CLPsych 2015 shared task participants. Even though screen names and URLs were anonymized using salted hash functions, the possibility of cross-referencing the hashed text against the Twitter archives still exists, and it could lead to breach of user privacy. Due to this reason, the researchers were asked to sign a confidentiality agreement to ensure the privacy of the data.

As social media interactions reside in a more naturalistic setting, it is important to identify to what extent an individual has disclosed their personal information, and whether the accurate and sufficient information is being published to determine whether a person has a mental disorder. The longitudinal data published on social media platforms have been identified as valuable (De Choudhury, 2013, 2014, 2015) with an extensive level of self-disclosure (Balani and De Choudhury, 2015; Park et al., 2012).

Most of the research conducted to detect mental illnesses in social media platforms has focused heavily on feature engineering. Throughout the literature, it could be identified that the most widely adopted feature engineering method is to extract lexical features using the Linguistic inquiry word count (LIWC) lexicon, which contains more than 32 categories of psychological constructs (Pennebaker et al., 2007). The lexicons have been used as one of the key feature extraction mechanisms in identifying insomnia (Jamison-Powell et al., 2012), distress (Lehrman et al., 2012), postpartum depression (De Choudhury et al., 2013), depression (Schwartz et al., 2014) and post-traumatic stress disorder (PTSD) (Coppersmith et al., 2014a). For each of these mental disorders to be identified, researchers had to extract features that overlap with each other, and are unique to a particular disorder. For example, the use of first-person pronouns (Lehrman et al., 2012) compared to the lesser use of second and third person pronouns (De Choudhury, 2013) are being used to detect users susceptible to dis-
tress and depression. To distinguish depression from PTSD, age is identified as a distinct feature (Preotiuc-Pietro et al., 2015a).

We found that working with the data extracted from the Twitter social media platform is challenging due to the unstructured nature of the text posted by users. The Twitter posts are introduced with new terms, misspelled words, syntactic errors, and character limitations when composing a message. Character n-gram models could be considered as an intuitive approach to overcome challenges imposed by unstructured data. Considering the effectiveness of such language models in classification tasks using Twitter data, Coppersmith et al. (2014a,b) has used unigram and character n-gram language models to extract features in the process of identifying users suspicious of having PTSD and several other mental illnesses such as bipolar disorder, depression, and seasonal affective disorder (SAD). Similarly, character n-grams can be identified as the key feature extraction mechanism in detecting mental illnesses such as attention deficit hyperactivity disorder (ADHD), generalized anxiety disorder, and eight other mental illnesses (Coppersmith et al., 2015a) as well as in detecting rare mental health conditions such as schizophrenia (Mitchell et al., 2015). Even though topic modelling techniques such as latent Dirichlet allocation (LDA) are being used to enhance the classifier predictability (Mitchell et al., 2015), researchers have identified supervised topic modeling methods (Resnik et al., 2015) and topics derived from clustering methods such as Word2Vec and GloVe Word Clusters (Preotiuc-Pietro et al., 2015b) to be more reliable in identifying users susceptible to having a mental illness. Further advancements in detecting mental health conditions were identified in the Computational Linguistics and Clinical Psychology (CLPsych) 2016 shared task (Milne et al., 2016) where post embedding’s (Kim et al., 2016) were used to determine the category of severity (i.e., crisis, red, amber and green) of forum posts published by users. In addition to lexical (e.g., character n-grams, word n-grams, lemma n-grams) and syntactic features (e.g., POS n-grams, dependencies), social behavioural patterns such as posting frequency and retweet rate, as well as the demographic details such as age, gender, and personality (Preotiuc-Pietro et al., 2015a) were also considered strong indicators in identifying mental illnesses. In general, the research in mental illness detection has evolved from the use of lexicon-based approaches to language models and topic models. The most recent research has tried to enhance models’ performance with the use of vector space representations and recurrent neural network layers with attention (Kshirsagar et al., 2017) to detect and explain posts depicting crisis.

In our research, we implement a model that produces competitive results for detecting depression of Twitter users (i.e., at user level not at post level) with limited data and without any exhaustive feature engineering.

### 3 Data

The training data consists of 1,145 Twitter users labeled as Control, Depressed, and PTSD (Coppersmith et al., 2015b). Also, each user of the dataset is labeled according to their gender and age. Table 1 represents detailed statistics of the dataset.

<table>
<thead>
<tr>
<th>Number of users</th>
<th>Control</th>
<th>Depressed</th>
<th>PTSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of tweets in each category</td>
<td>1,250,606</td>
<td>742,793</td>
<td>544,815</td>
</tr>
<tr>
<td>Average age</td>
<td>24.4</td>
<td>21.7</td>
<td>27.9</td>
</tr>
<tr>
<td>Gender (female) distribution per class</td>
<td>74%</td>
<td>80%</td>
<td>67%</td>
</tr>
</tbody>
</table>

Table 1: CLPSych 2015 shared task dataset statistics
AUC scores that we have obtained, we can conclude that the trained model is well generalized.

4 Methodology

The overall design of our approach is shown in Figure 1. We present a system that identifies users at the risk of depression from their social media posts.

Toward this, we present an efficient neural network architecture that improves and optimizes word embeddings. We evaluate the optimized embeddings produced by our architecture along with three commonly used word embeddings (Figure 1), random trainable, skip-gram, and CBOW, on the CLPsych 2015 shared task and the Bell Let’s Talk datasets. We perform a comparison on some selected CNN-based and RNN-based models to determine the best models and parameters across different settings for depression detection.

4.1 Preprocessing

We removed all the retweets, URL’s, @mentions, and all the non-alphanumeric characters. Also, all the stop words except for first, second, and third person pronouns were removed. From previous research, we identified that individuals susceptible to depression more regularly use first-person singular pronouns compared to the use of other pronouns (Pennebaker, 2011). The NLTK Tweet tokenizer is used to tokenize the messages. After tokenizing, we build a vocabulary (242,657 unique tokens) from the training dataset, which is used to encode text as a sequence of indices.

4.2 Word encoding

A network input is a sequence of tokens, such as words, where $S = [s_1, s_2, \ldots, s_t]$ and $t$ denotes the timestep. $S_i$ is the one-hot encoding of input tokens that have a fixed length ($T$), such that a sequence that exceeds this length is truncated. A word dictionary of fixed terms $W$ is used to encode a sequence. It contains three constants that determine the start and end of this sequence, in addition to the out of vocabulary (OOV) words. We normalize the variable text length using padding for short sequences and truncation for long sequences. We set the minimum occurrences of a word to 2 and the size of context window to 5, which produce 242,657 words. Then, we select the most frequent 100,000 words of them without stopwords.

4.3 Word Embedding Models

Word embedding models are fundamentally based on the unsupervised training of distributed representations, which can be used to solve supervised tasks. They are used to project words into a low-dimensional vector representation $x_i$, where $x_i \epsilon R^W$ and $W$ is the word weight embedding matrix. We pre-train two different Word2Vec (Mikolov et al., 2013) word embeddings, using Skip-gram and Continuous Bag-Of-Words (CBOW) distributed representation, in addition to a random (Rand) word embedding that has a uniform distribution scheme of a range (-0.5 to +0.5).

Word2Vec is a shallow model, in which neural layers, typically two, are trained to reconstruct a word context or the current word from their surrounding window of words. Skip-gram infers the nearby contextual words, as opposed to other distributed representations, such as CBOW, that focus on predicting current words. CBOW is a continuous skip-gram, in which the order of context words does not affect prediction or projection. CBOW is typically faster than skip-gram, which is slower but able to identify rare words (Mikolov et al., 2013).

Our embedding models are pre-trained on the CLPsych 2015 Shared task data. We also have an additional hyperparameter that is used to either freeze the embedding weight matrix or allow for further training.

4.4 Word Embedding Optimization

We implement an optimized approach for building an efficient word embedding to learn a better feature representation of health-specific tasks. Recently, there has been an increased use of *embeddings average* to compute word embedding, which
provides an improved feature representation that can be used across multiple tasks (Faruqui et al., 2015).

A word embedding is typically trained in an unsupervised manner using unlabeled data since it is not task-specific. We do the same by training our embedding on a large unlabeled training corpus (Collobert and Weston, 2008; Mikolov et al., 2013). Word2Vec trains word embeddings in a supervised manner, as it defines a training criterion that enables using unlabeled data (e.g., predicting the current word as in CBOW or context as in skip-gram). We do the same at the sentence level by predicting the surrounding sentences (Hill et al., 2016), as well as their possible sense (i.e., depressed, PTSD, neither, or unknown). W \( P \) denotes the words occurring next to a word \( w_i \) in the training data. \( W^N \) is a set of \( n \) words that are randomly selected and not occurring next to the word \( w_i \) in the training data. We use an analogizer activation as it enables all-positive outputs without losing any value. Then, we use cosine distance function to compute similarities among word representations, and to produce word probability representations.

5 Models

We describe four selected neural network models, which are used to evaluate the performance of depression detection. The first three models use CNN and the last one uses RNN. We build these model on the top of the word-embeddings described in the previous section. A drop-out of a probability 0.2 follows the word embedding layer. Each model is followed by a vanilla layer that is fully-connected, has 250 hidden units, and uses a Rectified Linear Unit (ReLU) activation. Then, we apply dropout with a probability of 0.2. The output layer is a fully-connected layer with one hidden unit, and it uses a sigmoid activation to produce an output.

5.1 Convolutional Neural Network (CNN)

A convolution operation is a representation of learning from sliding w-grams for an input sequence of \( d \) entries, \( e_1, e_2, \ldots, e_t \). A vector \( c_t \in R^{ed} \) is the concatenated embedding of \( f \) entries, such that \( x_{i-f+1}, \ldots, x_i \) where \( f \) is the filter length. For w-gram, we generate a representation \( p_i \in R^{ed} \) using convolution weights \( We \in R^{d \times wd} \) where a bias \( be \) and \( p_i = \text{tanh}(W_{x_i}+b) \).

CNNWithMax: We apply a one-dimensional con-
volution operation with 250 filters and a kernel of size 3, where \( w^i_t = \text{conv1d}(s_i) \) and \( f \) is the filter length. After that, a global max-pooling layer is applied on the feature map to extract global abstract information, such that \( \hat{w}^j = \text{globalmax}(w^j_t) \), which results in an abstract feature representation of length 250.

**MultiChannelCNN:** We apply 3 convolutions, each of which has 128 features and filters of the lengths 3, 4, and 5. A one-dimensional operation is used, where \( w^i_t = \text{Conv1d}(S_i) \), and \( f \) is the filter length. Then, a max-pooling layer is applied on the feature map to extract abstract information, \( \hat{w}^j_t = \text{max}(C^j_t) \). Finally, we concatenate feature representations into a single output. Conversely to recurrent layers, convolutional operations are helpful with max-pooling to extract word features without considering the sequence order (Kalchbrenner et al., 2014). Such features can be used with recurrent features in order to improve the model performance.

**MultiChannelPoolingCNN:** We extend the previous model to apply two different max-pooling sizes, 2 and 5.

### 5.2 Recurrent Neural Network (RNN)

It is commonly used in NLP as it allows for remembering values over different time durations. In RNN, each element of an input embedding \( x_i \) is processed sequentially. \( h_t = \text{tanh}(Wx_i + Wh_{i-1}) \) and \( W \) represent the weight matrix between an input and hidden states \( (h_t) \) of the recurrent connection at timestep \( (t) \). RNN allows for variable length processing while maintaining the sequence order. However, it is limited when it comes to long sentences due to the exponentially growing or decaying gradients. Long short term memory (LSTM) is a common way to handle such a limitation using gating mechanisms.

**Bidirectional LSTM with attention:** We use bidirectional LSTM layers with 100 units, which receive a sequence of tokens as inputs. Then, the LSTM projects word information \( H = (h_1, h_2, \ldots, h_T) \), in which \( h_t \) denotes the hidden state of LSTM at a timestep \( (t) \). LSTM captures the temporal and abstract information of sequences forwardly \( (h^f) \) or backwardly \( (h^b) \). Then, we concatenate both forward and backward representations, where \( h_t = h^f_t || h^b_t \). Finally, we use the last output in the sequence.

**Context-aware Attention:** Words have different weight values, as they are generally not equal. Thus, we use an attention mechanism (Bahdanau et al., 2014; Vaswani et al., 2017) to focus on the important words. We use a context-aware attention mechanism (Yang et al., 2016), which is the weighted summation of all words in a given sequence \( (r = \Sigma_{i=1}^T a_i h_i) \). We use this representation as a classification feature vector.

### 6 Models Training

For training, we minimize the validation loss error between the actual and predicted classes in order to learn the network parameters. A mini-batch gradient descent with a batch size 32 is applied to improve the network loss function through backpropagation. Adam optimizer (Kingma & Ba, 2014) with a learning rate of 0.007 is used to train our models. The gradient norm (Pascaru, Mikolov, & Bengio, 2012) is clipped at 7, which protects our model from the exploding gradient.

**Regularization:** we randomly drop neurons off a network using dropout in order to prevent co-adaptation of those neurons (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). Dropout is also used on the recurrent connection of our LSTM layers. We additionally, apply weight decay using L2 regularization penalty (Cortes, Mohri, & Rostamizadeh, 2012).

**Hyperparameters:** we use an embedding layer of the size 300, and an LSTM layer of size 50, which increases to be 100 for the bidirectional LSTM. We apply a dropout of 0.4 and 0.2 on the recurrent connections. Finally, an L2 regularization of 0.0001 is applied at the loss function.

### 7 Experiments

We evaluate our approach using two experiments, 1) depression detection on the CLPsych2015 dataset (Section 3) and 2) test generalization ability on the Bell Let’s Talk dataset (Section 3). We use different word embeddings for our experiments with the deep neural network models. In the first experiment, we perform a comparison on the selected models for depression detection.

In the second experiment, since the dataset is imbalanced, we perform 5-fold cross-validation with stratified sampling to report results. Data points are shuffled for each split while maintaining the class distribution. After that, we test the generalization ability of the models selected, for which we use 80% and 20% of the data for train-
<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>F1</th>
<th>AUC</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>77.480</td>
<td>77.472</td>
<td>0.844</td>
<td>77.601</td>
<td>77.480</td>
</tr>
<tr>
<td>CNNWithMax</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>87.957</td>
<td>86.967</td>
<td>0.951</td>
<td>87.435</td>
<td>87.029</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>79.813</td>
<td>78.460</td>
<td>0.879</td>
<td>79.707</td>
<td>78.979</td>
</tr>
<tr>
<td>CBOW</td>
<td>60.768</td>
<td>43.095</td>
<td>0.544</td>
<td>38.056</td>
<td>54.207</td>
</tr>
<tr>
<td>Trainable</td>
<td>80.820</td>
<td>80.173</td>
<td>0.909</td>
<td>80.440</td>
<td>82.099</td>
</tr>
<tr>
<td>MultiChannelPoolingCNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>87.510</td>
<td>86.491</td>
<td>0.950</td>
<td>87.266</td>
<td>86.768</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>78.818</td>
<td>76.073</td>
<td>0.883</td>
<td>80.514</td>
<td>75.691</td>
</tr>
<tr>
<td>CBOW</td>
<td>49.667</td>
<td>37.573</td>
<td>0.556</td>
<td>33.580</td>
<td>55.652</td>
</tr>
<tr>
<td>Trainable</td>
<td>73.691</td>
<td>72.021</td>
<td>0.824</td>
<td>72.762</td>
<td>72.799</td>
</tr>
<tr>
<td>MultiChannelCNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>85.617</td>
<td>84.153</td>
<td>0.935</td>
<td>85.817</td>
<td>84.064</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>61.688</td>
<td>53.757</td>
<td>0.700</td>
<td>53.757</td>
<td>53.757</td>
</tr>
<tr>
<td>CBOW</td>
<td>76.248</td>
<td>72.047</td>
<td>0.803</td>
<td>74.787</td>
<td>71.742</td>
</tr>
<tr>
<td>Trainable</td>
<td>82.268</td>
<td>80.347</td>
<td>0.870</td>
<td>82.770</td>
<td>79.983</td>
</tr>
<tr>
<td>BiLSTM (Context-aware attention)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>64.136</td>
<td>65.124</td>
<td>0.925</td>
<td>81.626</td>
<td>84.349</td>
</tr>
<tr>
<td>Trainable</td>
<td>77.589</td>
<td>75.193</td>
<td>0.832</td>
<td>76.687</td>
<td>74.925</td>
</tr>
</tbody>
</table>

Table 2: Performance of our models on the CLPsych 2015 dataset with 5-fold cross-validation. The rows are highlighted according to the highest AUC score.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>F1</th>
<th>AUC</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>77.460</td>
<td>73.460</td>
<td>0.718</td>
<td>73.322</td>
<td>74.025</td>
</tr>
<tr>
<td>CNNWithMax</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>64.935</td>
<td>67.878</td>
<td>0.751</td>
<td>68.376</td>
<td>69.681</td>
</tr>
<tr>
<td>CBOW</td>
<td>61.688</td>
<td>53.757</td>
<td>0.700</td>
<td>53.757</td>
<td>53.757</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>68.182</td>
<td>67.456</td>
<td>0.773</td>
<td>68.387</td>
<td>70.362</td>
</tr>
<tr>
<td>Trainable</td>
<td>83.117</td>
<td>82.252</td>
<td>0.925</td>
<td>81.626</td>
<td>84.349</td>
</tr>
<tr>
<td>MultiChannelCNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>62.338</td>
<td>57.491</td>
<td>0.586</td>
<td>57.687</td>
<td>57.388</td>
</tr>
<tr>
<td>CBOW</td>
<td>72.078</td>
<td>66.882</td>
<td>0.734</td>
<td>68.969</td>
<td>66.159</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>60.390</td>
<td>54.599</td>
<td>0.525</td>
<td>54.911</td>
<td>54.558</td>
</tr>
<tr>
<td>Trainable</td>
<td>82.468</td>
<td>81.513</td>
<td>0.888</td>
<td>80.871</td>
<td>83.495</td>
</tr>
<tr>
<td>MultiChannelPoolingCNN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimized</td>
<td>51.948</td>
<td>50.752</td>
<td>0.682</td>
<td>69.076</td>
<td>62.918</td>
</tr>
<tr>
<td>CBOW</td>
<td>64.266</td>
<td>64.248</td>
<td>0.752</td>
<td>69.307</td>
<td>70.082</td>
</tr>
<tr>
<td>Skip-gram</td>
<td>63.636</td>
<td>62.731</td>
<td>0.733</td>
<td>63.636</td>
<td>65.104</td>
</tr>
<tr>
<td>Trainable</td>
<td>80.519</td>
<td>80.035</td>
<td>0.914</td>
<td>80.519</td>
<td>83.803</td>
</tr>
</tbody>
</table>

Table 3: Performance of our models on the Bell Let’s Talk dataset. The rows are highlighted according to the highest AUC score.

The majority of the researchers have relied on support vector machine (SVM) classifiers to distinguish users with mental disorders from control groups and different mental disorder categories except when trying to identify the level of depression with the use of regression models (Schwartz et al., 2014). We used the SVM linear classifier with TF-IDF to initiate a baseline for the binary classification task. For evaluation, we used five-fold cross-validation, and the resulting best model was used on the Bell Let’s Talk dataset to predict users with depression. The results are reported both on the validation and test data.

Table 2 shows good standings results for depression detection, which indicates that regularization and hyperparameter tuning helped resolve the overfitting issues. CNN-based with max-pooling models reported better performance than RNN-
based models. The CNNWithMax models using our optimized embedding reported higher accuracy (87.957%), F1 (86.967%), AUC (0.951), precision (87.435%), and recall (87.029%), as compared to other models. Table 2 reports that CNN-based models’ results are close to each other, as opposed to RNN-based models, which at best reported 83.236% with trainable random embedding (trainable). Interestingly, CNN models performed better than RNN models for depression detection.

Table 3 reports the generalization ability of our approach on the unseen dataset (Section 3). The models trained using our optimized embedding managed to maintain their performance with generalization ability. Our embedding performs better because it is optimized using the CLPsych2015 dataset, which includes depression and PTSD labeled data. Table 3 shows that the results of the CNN models are competitive, as opposed to RNN models. The best performing RNN model reported 91.425%. CBOW embedding performed the least as compared to others, including the random embedding. In particular, pre-trained CBOW and skip-gram models do not perform as expected, mainly due to the size of the CLPsych2015 corpus, which is nearly around 22 million words. Furthermore, optimized and trainable random embeddings have an advantage for being able to update their weights during training. We conclude that user-level classification for depression detection performs well even with datasets that are small and/or imbalanced.

8 Comparison to Related Work

Resnik et al. (2015) and Preotiuc-Pietro et al. (2015b) reported high results for the CLPsych2015 shared task using topic models. However, their results are not comparable, as they are reported on the official testing set that was not available to us. Alternatively, we performed a five-fold cross-validation on the shared task training data (Tables 2 and 3). We report better performance when testing on the Bell Let’s Talk dataset as compared to Jamil et al. (2017).

9 Conclusion

In conclusion, we presented a novel approach to optimize word-embedding for classification tasks. We performed a comparative evaluation on some of the widely used deep learning models for depression detection from tweets on the user level. We performed our experiments on two publicly available datasets, CLPsych2015 and Bell Lets Talk. Our experiments showed that our CNN-based models perform better than RNN-based models. Models with optimized embeddings managed to maintain performance with the generalization ability.

For future work, we will evaluate against more RNN-based models, in particular with more focus on attention mechanisms. We will investigate other kinds of mental disorders, such as PTSD.

Acknowledgments

This research is funded by Natural Sciences and Engineering Research Council of Canada (NSERC) and the Ontario Centres of Excellence (OCE).

References


