ConvSent at CLPsych 2019 Task A: Using Post-level Sentiment Features for Suicide Risk Prediction on Reddit

Kristen Allen *
Dept of Engineering and Public Policy
Carnegie Mellon University
Pittsburgh, PA 15213
kcallen@cmu.edu

Shrey Bagroy *
Computer Science Dept
Carnegie Mellon University
Pittsburgh, PA 15213
sbagroy@cs.cmu.edu

Alex Davis
Dept of Engineering and Public Policy
Carnegie Mellon University
Pittsburgh, PA 15213
ald1@andrew.cmu.edu

Tamar Krishnamurti
Division of General Internal Medicine
University of Pittsburgh
Pittsburgh, PA 15213
tamark@pitt.edu

Abstract

This work aims to infer mental health status from public text for early detection of suicide risk. It contributes to Shared Task A in the 2019 CLPsych workshop by predicting users' suicide risk given posts in the Reddit subforum r/SuicideWatch. We use a convolutional neural network architecture to incorporate LIWC information at the Reddit post level about topics discussed, first-person focus, emotional experience, grammatical choices, and thematic style. In sorting users into one of four risk categories, our best system's macro-averaged F1 score was 0.50 on the withheld test set. The work demonstrates the predictive power of the Linguistic Inquiry and Word Count dictionary, in conjunction with a convolutional network and holistic consideration of each post and user.

1 Introduction

Psychological distress in the form of depression, anxiety, and other mental health issues can have serious consequences for individuals and society (WHO, 2017). Unfortunately, stigma surrounding poor mental health may prevent disclosure of suicidal ideation. For example, Oexle et al. (2017) found that perceived stigma and the associated secrecy around mental illness were positively linked with feelings of hopelessness and suicidal ideation. McHugh et al. (2019) found that the standard practice of clinicians asking people about suicidal thoughts fails in many cases, as 80% of patients who ultimately died of suicide reported no suicidal thoughts when prompted by their general practitioner.

There is a need to supplement traditional methods for evaluating suicidality that minimize the need for direct disclosure from the individual. Some of those suffering from mental health challenges have adopted social media outlets, such as Reddit's r/SuicideWatch, as a means to cope (Park et al., 2012; Robinson et al., 2016). Recent research finds promising links between an individual's mental well-being and the linguistic content they share on social media (Coppersmith et al., 2014; De Choudhury et al., 2016; Vioulès et al., 2018; Shing et al., 2018).

The Sixth Annual Workshop on Computational Linguistics and Clinical Psychology (CLPsych 2019) includes a shared task on predicting a Reddit user's degree of suicide risk based on their posts in the r/SuicideWatch forum (Zirikly et al., 2019). The task involves assigning a degree of risk (no, low, moderate, or severe) to a user on Reddit based on content they have posted on Reddit. For this task, researchers were given access to the University of Maryland Reddit Suicidality Dataset (Shing et al., 2018), made available with assistance by the American Association of Suicidology. This dataset consists of ∼1000 users annotated with the four-level scale, and a larger set of 20,000 unannotated users.

2 Prior work

The baseline deep learning model for classifying suicide risk on Reddit, by Shing et al. (2018), builds on the convolutional neural network (CNN) for language processing as laid out by Kim (2014). Shing et al.'s CNN makes use of unigram word embeddings, concatenated by post and then by user, then constructs an overall user score using

* These authors contributed equally

Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology, pages 182–187
Minneapolis, Minnesota, June 6, 2019.

Model	Precision	Recall	F1
CNN + GloVe vectors	0.55	0.43	0.42
Affect-only CNN + LIWC	0.53	0.47	0.49
Primary: CNN + all LIWC	0.65	0.55	0.56

Table 1: Average performance of our models in 10-fold cross-validation on the training set

Model	Full F1	Flagged F1	Urgent F1
Primary	0.37	0.88	0.77
Leave none out	**0.50**	**0.90**	**0.82**
Balanced classes	0.41	**0.90**	0.80

Table 2: Performance of our models by macro-averaged F1 on the test set. 'Full F1' indicates score across four classes, while 'flagged' and 'urgent' F1 reflect binary splits between no/some risk and non-severe/severe risk, respectively. All three submitted models use a convolutional network plus all LIWC features.

sliding windows over that sequence. In a separate approach, Shing et al. use an SVM to consider post-level features but make an overall risk assessment based on the most concerning individual post. Neither method incorporates distinct insights from individual posts—where, for instance, a long series of moderately concerning posts might indicate more serious risk. Our model incorporates information from multiple posts within the CNN framework.

We additionally leverage prior social media work (Braithwaite et al., 2016; Coppersmith et al., 2015) that finds suicidality can be predicted from a particular feature set, the Linguistic Inquiry and Word Count (LIWC) dictionary, as distributed by Tausczik and Pennebaker (2010).

3 Methods

All modeling methods were applied to the de-identified Reddit data as part of Shared Task A. Approval from CMU IRB was obtained on March 11 2019, and we adhered to the ethical review criteria laid out by Zirikly et al. (2019).

3.1 Modeling with word embeddings

Convolutional neural networks form the basic architecture for our models. Following Shing et al. (2018) and Kim (2014), we concatenate word embeddings for each word in a post, then concatenate these embedding sequences for all posts in order of occurrence. Our implementation uses pre-trained GloVe word embeddings by Pennington et al. (2014) and code snippets from Neubig et al. (2019).

In both of these experiments, we transform all posts by a user into a two-dimensional array of dimension $num\_total\_words \times embedding\_size$. For the CNN, filter parameters that must be trained are then $window\_size \times embedding\_size \times num\_filters$. Given the small size of the expert-annotated dataset, we next explore ways to reduce the number of features a network needs to train.

3.2 Modeling with post-level features

We next consider post-level features. In this dataset the post body field is often empty, presumably when the post comprises only an image or other embedded media, so features must be robust to this variation. In all subsequent models, each post component (title or body) is represented as a one-dimensional vector of size $num\_post\_features$. Calling each such 1-D vector $\mathbf{x}_{ij}$, we chronologically concatenate these vectors for each post title and non-empty body for user i into a longer 1-D vector:

$$\mathbf{x}_i = \mathbf{x}_{i1} \oplus \mathbf{x}_{i2} \oplus \ldots \oplus \mathbf{x}_{in}.$$

Thus we represent each user with the concatenated vector of all post features from posts $1:n$, where n is their total number of post titles and non-empty post bodies. The resulting vector for user i has shape $1 \times (n * num\_post\_features)$. Users are then batched for quicker training. Each user vector is padded to the length of the longest one, resulting in a batch of k user vectors having shape $k \times (n_{max} * num\_post\_features)$. Masking prevents back-propagation of weights to padding vectors.

Others' prior work successfully incorporated LIWC features into suicidality detection (e.g. Lightman et al. (2007)). Thus, we experiment with sets of LIWC features as the summary of each post by a user, then concatenate these features from all of a user's posts. In order to maintain cross-post context while reducing the number of features, the first model considers only features from the 'affect' category. Using just these sentiments appeared likely to predict self-destructive mental state (Kumar et al., 2015). Subsequent models use all 45 features provided in the LIWC dictionary.

We next apply a convolutional neural network to this 1-D sequence of LIWC features. Our network uses the `keras` implementation of a

one-dimensional CNN (Chollet et al., 2015), setting both stride length and window size equal to $num\_post\_features$ and using $num\_filters = 10$ filters. This structure means that each window looks at LIWC features from a single post title or body, and extracts relationships between these features into 10 filter representations. The model forgoes pooling (following Springenberg et al. (2014)) in favor of maintaining independent information about each post. Thus, after convolution, the batch of k users with max number of posts n_{max} has shape $k \times (n_{max} * num\_filters)$.

Convolution is followed by a dropout layer setting 30% of input units to 0 at any given timestep, intended to reduce overfitting. The next two layers are fully connected, with 250 and 100 nodes, respectively, and rectified linear activation functions; thus, after passing through the second linear layer, the data has shape $k \times 100$. Finally, labels are generated by a softmax output layer. Training seeks to minimize cross entropy, and uses 10-fold cross-validation (CV) on the training set.

'Affect-only' model

This model uses the four affect categories relating to negative sentiment: 'negative affect,' 'anger,' 'anxiety,' and 'sadness'. We selected this subset as a reasonable approximation of negative valence, and to test its predictive performance without broader information.

'Primary' model

The best-performing model on a set-aside development set serves as our primary model. This model differes from the affect-only model in incorporating all 45 LIWC categories as post features.

'Balanced classes' model

Next, we provide our model with custom weights corresponding to the penalty incurred while misclassifying each class. We provide larger weights for the underrepresented 'low risk' and 'moderate risk' classes to force the model to pay more attention to these categories while training.

'Leave none out' model

This final model used all available data for training. In the primary and balanced models, it was clear that while training set performance continues to improve, development set performance levels off somewhere around 150 epochs. That is, cross-validation results were optimized at epoch 235 for the primary model, and 67 for the balanced classes model. Taking the average, this system uses the model state after epoch 150 to predict test set results.

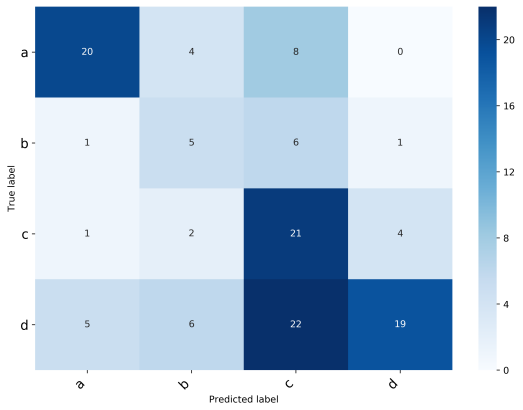


Figure 1: Confusion matrix on the test set from the best-performing model

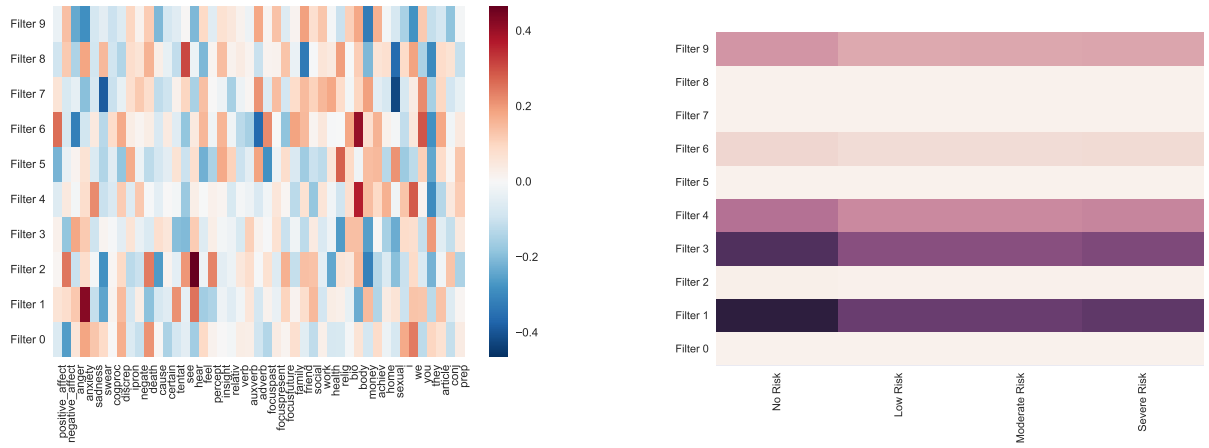
Our primary evaluation metric is the resulting macro-averaged F1 score of our models; we report averages on a set-aside development set (see Table 1). For three approaches, we also present macro-averaged F1 scores on an unseen test set.

4 Results

With our initial convolutional network model, using GloVe word embeddings in a convolutional neural net in the style of Kim (2014), we confirm similar performance to Shing et al. (2018) with a macro-averaged F1 score of 0.42. We also find that this model strongly overfits the data; it performs exceptionally well on the training data (F1=0.95) but fails to generalize well on development data (F1=0.42). This overfitting is expected, since the size of our dataset is not sufficient to successfully train large models.

The high overfitting and our model's inability to further learn from the dataset encourage us to focus on simpler models, and to thoughtfully select our features.

The best-performing models all use LIWC features at the post level, concatenated by user, and run through a one-dimensional CNN with stride length and window size equal to the number of features.



(a) Filter visualizations for each of the 10 filters

(b) Strength of average alignment between filters and the four classes.

Figure 2: Filters for the best-performing model indicate

4.1 'Affect-only' model

When representing each post as a vector of LIWC affect features, we find that the base model achieves an F1-score of 0.47 in cross-validation. We still find a significant discrepancy between our model's performance on seen/unseen data, indicating that it still suffers from overfitting. We experiment with hyperparameters like dropout and number of filters, finding that a model with 10 filters and 0.3 dropout probability outperforms all our previous models with a macro-averaged CV F1-score of 0.49.

On studying the performance of our model, we find that its behaviour is not uniform across all classes: it does well in labeling 'no risk' and 'severe risk,' but performs poorly in trying to label the intermediate risk categories.

4.2 'Primary' model

We next use variations to improve features provided while still minimizing parameters trained. For our 'primary' model, we provide all 45 LIWC category features to a CNN of the same structure.

In macro-averaging pairwise AUC scores on the development set, this model scores 0.76. On the test set, the model's macro-averaged F1 is 0.37. A random guessing strategy weighted by label frequency would yield F1=0.25.

4.3 'Balanced classes' model

We find that this change boosts the model's CV performance on our development set to an F1 score of 0.57, with a macro-averaged AUC score on the development set of 0.78. We also find that this model performs more uniformly across the four classes than we see in the previous model, resulting in a slightly better score on the unseen test set, F1=0.40.

4.4 'Leave none out' model

With this final model and feature architecture, we train our model on the entire training dataset available for Task A, stopping after 150 epochs. This model achieves our highest score on the test set, a macro-averaged F1-score of 0.50 on this task–comparing favorably with the best-scoring system, whose F1-score is 0.53. We also note that our model achieves high F1-scores (0.90 and 0.82 respectively) for the 'flagged' and 'urgent' tasks.

This model's final confusion matrix is shown in Figure 1. We find that our model is best at identifying the 'no risk' and 'moderate risk' users, while it miscategorizes 42% of 'severe risk' cases as 'moderate risk' as well. There are fewer 'low risk' users, and about half of these are miscategorized as 'moderate risk' as well.

5 Discussion

5.1 'Affect-only' model

We can attribute this model's difficulty with intermediate labels to our usage of only the negative 'affect' category from LIWC. This category extracts counts for words associated with 'negative_affect,' 'anger,' 'anxiety,' and 'sadness', i.e, words one would typically associate with severe suicidality conditions; presence of (a large number of) these words may be common in Severe risk

users, whereas their absence might be a strong indicator of No risk users. Poorer performance in the intermediate categories may indicate inconsistent use of emotion terms by those users, or may suggest a smaller range of variation between those categories as opposed to variation within the extremes.

5.2 'Primary' and 'balanced classes' models

The 'primary' and 'balanced classes' models perform similarly, with a difference in F1 scores of about 0.03. We believe that the latter model is slightly more effective because its higher weights for the intermediate categories counteracted those labels' lower representation in the training set. This is borne out in the model's slightly better performance on those classes: it categorizes $\frac{1}{13}$ of 'low risk' and $\frac{10}{28}$ 'moderate risk' users correctly, whereas the 'primary' model is right about $\frac{0}{13}$ and $\frac{8}{28}$ of such users, respectively. Macro-averaged F1 as the primary metric means that even this slight improvement is significant when comparing the two models.

It seems plausible that, because it was trained for longer, the 'primary' model was more overfitted to the training data. Because we use 10-fold cross-validation to train these models, we also note that both these models are trained using 90% of the training data; we hypothesize this missing 10% of data to be the primary reason that our leave-none-out model outperforms both of these models. A larger training dataset allows the model to "observe" more data, which helps both with getting more training data for under-represented classes (e.g. low and moderate risk) and with generalizing better on all unseen data.

5.3 'Leave none out' model

Difficulty identifying 'low risk' users may be partially explained by the fact that fewer users from the training set were in that class than any other– just 10% of examples were labeled low risk, so there was less opportunity to learn these features.

In Figure 2a, we plot the learned convolutional layer weights from our final model with respect to the input LIWC feature categories, finding that each filter is activated (or deactivated) by a subset of LIWC features. We hypothesize that each filter focuses on learning presence or absence of a particular character trait (or 'sentiment') from each post. For instance, filter 9 is inversely associated with money, anxiety, and 'we,' indicating that someone describing their stress around money would have a negative activation for Filter 9. Seeing a stronger association between Filter 9 and 'no risk,' we can extrapolate that users who are not at risk are less likely to be preoccupied with their financial troubles on r/SW.

While not all subsets are clear, we can observe some patterns. For instance, Filter 2 has the highest positive weights for 'hear,' 'negative_affect,' 'death,' 'percept,' and 'see.' We could hypothesize that a user activating this filter is preoccupied with how they are perceived, and is also considering death (whether their own or that of a loved one). This filter may indicate both a feeling of being observed, perhaps stigmatized, and an experience of suicidal ideation, as discussed by Oexle et al. (2017).

5.4 Findings

Overall, this work demonstrates the power of combining human feature-engineering with deep learning in data-constrained situations. The Linguistic Inquiry and Word Count dictionary, in conjunction with a convolutional network, leads to a holistic consideration of each post and each user, all while reducing the overall number of parameters the network needs to learn. Within the constraints of a relatively small dataset, we find that our best model incorporates engineered features and all available data to outperform a 'baseline' re-implementation of Shing et al. (2018).

5.5 Acknowledgements

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE1252522. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Dr. Krishnamurti's time was supported by an Institutional K-award (NIH KL2 TR001856).

References

Scott R Braithwaite, Christophe Giraud-Carrier, Josh West, Michael D Barnes, and Carl Lee Hanson. 2016. Validating machine learning algorithms for Twitter data against established measures of suicidality. *JMIR mental health*, 3(2).

François Chollet et al. 2015. Keras. https://github.com/fchollet/keras.

Glen Coppersmith, Mark Dredze, and Craig Harman. 2014. Quantifying mental health signals in Twitter. In *Proceedings of the workshop on computational linguistics and clinical psychology: From linguistic signal to clinical reality*, pages 51–60.

Glen Coppersmith, Ryan Leary, Eric Whyne, and Tony Wood. 2015. Quantifying suicidal ideation via language usage on social media. In *Joint Statistics Meetings Proceedings, Statistical Computing Section, JSM*.

Munmun De Choudhury, Emre Kiciman, Mark Dredze, Glen Coppersmith, and Mrinal Kumar. 2016. 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*, pages 2098–2110. ACM.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Mrinal Kumar, Mark Dredze, Glen Coppersmith, and Munmun De Choudhury. 2015. Detecting changes in suicide content manifested in social media following celebrity suicides. In *Proceedings of the 26th ACM conference on Hypertext & Social Media*, pages 85–94. ACM.

Erin J Lightman, Philip M McCarthy, David F Dufty, and Danielle S McNamara. 2007. Using computational text analysis tools to compare the lyrics of suicidal and non-suicidal songwriters. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 29.

Catherine M. McHugh, Amy Corderoy, Christopher James Ryan, Ian B. Hickie, and Matthew Michael Large. 2019. Association between suicidal ideation and suicide: meta-analyses of odds ratios, sensitivity, specificity and positive predictive value. *BJPsych Open*, 5(2):e1.

Graham Neubig, Daniel Clothiaux, Vaibhav, Zhengzhong Liu, Danish Pruthi, and Zhiting Hu. 2019. Code samples from Neural Networks for NLP. https://github.com/neubig/nn4nlp-code.

N Oexle, V Ajdacic-Gross, R Kilian, M Müller, S Rodgers, Z Xu, W Rössler, and N Rüsch. 2017. Mental illness stigma, secrecy and suicidal ideation. *Epidemiology and psychiatric sciences*, 26(1):53–60.

Minsu Park, Chiyoung Cha, and Meeyoung Cha. 2012. Depressive moods of users portrayed in Twitter. In *Proceedings of the ACM SIGKDD Workshop on healthcare informatics (HI-KDD)*, volume 2012, pages 1–8. ACM New York, NY.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Jo Robinson, Georgina Cox, Eleanor Bailey, Sarah Hetrick, Maria Rodrigues, Steve Fisher, and Helen Herrman. 2016. Social media and suicide prevention: a systematic review. *Early intervention in psychiatry*, 10(2):103–121.

Han-Chin Shing, Suraj Nair, Ayah Zirikly, Meir Friedenberg, Hal Daumé III, and Philip Resnik. 2018. Expert, crowdsourced, and machine assessment of suicide risk via online postings. In *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*, pages 25–36.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54.

M Johnson Vioulès, Bilel Moulahi, Jérôme Azé, and Sandra Bringay. 2018. Detection of suicide-related posts in Twitter data streams. *IBM Journal of Research and Development*, 62(1):7–1.

WHO. 2017. Policy options on mental health: a WHO-Gulbenkian mental health platform collaboration. Technical report, World Health Organization.

Ayah Zirikly, Philip Resnik, Özlem Uzuner, and Kristy Hollingshead. 2019. CLPsych 2019 shared task: Predicting the degree of suicide risk in Reddit posts. In *Proceedings of the Sixth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*.