Determining a Person's Suicide Risk by Voting on the Short-Term History of Tweets for the CLPsych 2021 Shared Task

Ulya Bayram
Department of Electronics Engineering
Çanakkale Onsekiz Mart University
Çanakkale, Turkey
ulya.bayram@comu.edu.tr

Lamia Benhiba
IAD Department, ENSIAS,
Mohammed V University in Rabat
Rabat, Morocco
lamia.benhiba@um5.ac.ma

Abstract

In this shared task, we accept the challenge of constructing models to identify Twitter users who attempted suicide based on their tweets 30 and 182 days before the adverse event's occurrence. We explore multiple machine learning and deep learning methods to identify a person's suicide risk based on the short-term history of their tweets. Taking the real-life applicability of the model into account, we make the design choice of classifying on the tweet level. By voting the tweet-level suicide risk scores through an ensemble of classifiers, we predict the suicidal users 30-days before the event with an 81.8% true-positives rate. Meanwhile, the tweet-level voting falls short on the six-month-long data as the number of tweets with weak suicidal ideation levels weakens the overall suicidal signals in the long term.

1 Introduction

Suicide is amongst the most pressing public health issues facing today's society, stressing the need for rapid and effective detection tools. As people are increasingly self-expressing their distress on social media, an unprecedented volume of data is currently available to detect a person's suicide risk (Roy et al., 2020; Tadesse et al., 2020; Luo et al., 2020). In this shared task, we aim to construct tools to identify suicidal Twitter users (who attempted suicide) based on their tweets collected from spans of 30-days (subtask 1) and six months (subtask 2) before the adverse event's occurrence date (Macavaney et al., 2021). The small number of users in the labeled collections of subtask 1 (57 suicidal/57 control) and subtask 2 (82 suicidal/82 control) and the scarcity of tweets for some users pose these tasks as small-dataset classification challenges. On that note, Coppersmith et al. (2018) reported high performance with deep learning (DL) methods on these collections after enriching them with additional data (418 suicidal/418 control).

When formulating the strategy to attack the challenge, we were motivated by the real-life applicability of the methods. Some social media domains already started implementing auto-detection tools to prevent suicide (Ji et al., 2020). These tools continuously monitor the presence of suicide risk in new posts. Therefore, we chose to train the models at the tweet level. Next, we develop a majority voting scheme over the classified tweets to report an overall suicide risk score for a user. We employ simple machine learning (ML) methods and create an ensemble. We also experiment with DL methods to assess whether complexity would improve the results. Since successful ML applications thrive on feature engineering (Domingos, 2012), we conduct feature selection to evaluate and determine the best feature sets for the models.

Our experiments suggest that majority voting (MV) over tweet-level classification scores is a viable approach for the short-term prediction of suicide risk. We observe that DL methods require plentiful resources despite the small size of the datasets. Simple ML methods with feature selection return satisfactory results, and the performance further improves by the ensemble classifier. We also observe that the MV approach falls short on the six-month-long data regardless of the applied model. Yet this limitation provides the invaluable insight that suicidal ideation signals are more significant when the date of the suicidal event is closer, which stresses the need for more complex, noise immune models for longer time-spanning data. In this context, we consider a noise-immune model as a suicidal ideation detection model that is not affected by tweets lacking suicidal ideation.

2 Methods

Pre-processing: We clean the tweets by removing user mentions, URLs, punctuation, and non-ASCII characters, then normalize hashtags into words using a probabilistic splitting tool based on English

Proceedings of the Seventh Workshop on Computational Linguistics and Clinical Psychology, pages 81–86
June 11, 2021. ©2021 Association for Computational Linguistics

Wikipedia unigram frequencies (Anderson, 2019). We maintain stopwords and emojis, as they might provide clues regarding the suicidal ideation of the users.

Experimentation Framework: Before designing the experiments, we face a critical choice: Should we merge all tweets per user, or should we perform the assessment per tweet and then aggregate the scores? To answer this, we consider a real-life risk assessment system. The system should provide a score every time someone posts a tweet. Some social media domains already implement these systems (Ji et al., 2020). Hence, we select to train the models to classify tweets, then apply majority voting (MV) per user to compute a risk score based on the tweet scores. Our framework is described in Figure 1.

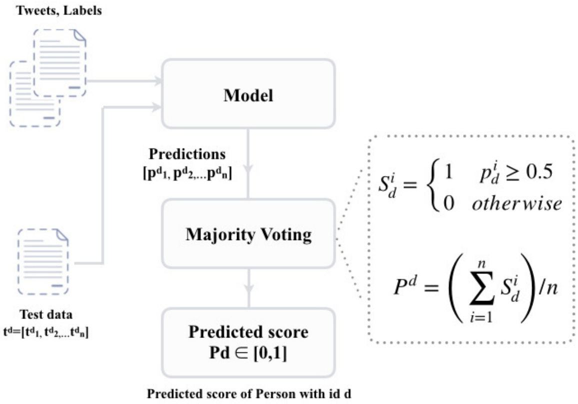

Tweets, Labels
Model
Predictions
$[p^d_1, p^d_2, ... p^d_n]$
Majority Voting
$S^i_d = \begin{cases} 1 & p^i_d \geq 0.5 \\ 0 & otherwise \end{cases}$
$P^d = \left(\sum_{i=1}^{n} S^i_d \right) / n$
Test data
$t^d = [t^d_1, t^d_2, ... t^d_n]$
Predicted score
Pd ∈ [0,1]
Predicted score of Person with id d

Figure 1: Classification framework used to compute person-level risk scores from the tweet-level scores.

Experiments with Standard ML methods: Before ML experiments, we initially explore a simple approach that constructs graphs from training sets and computes how well the given texts match the graphs (Bayram et al., 2018). However, tweets proved to be unfit for the method due to low word counts.

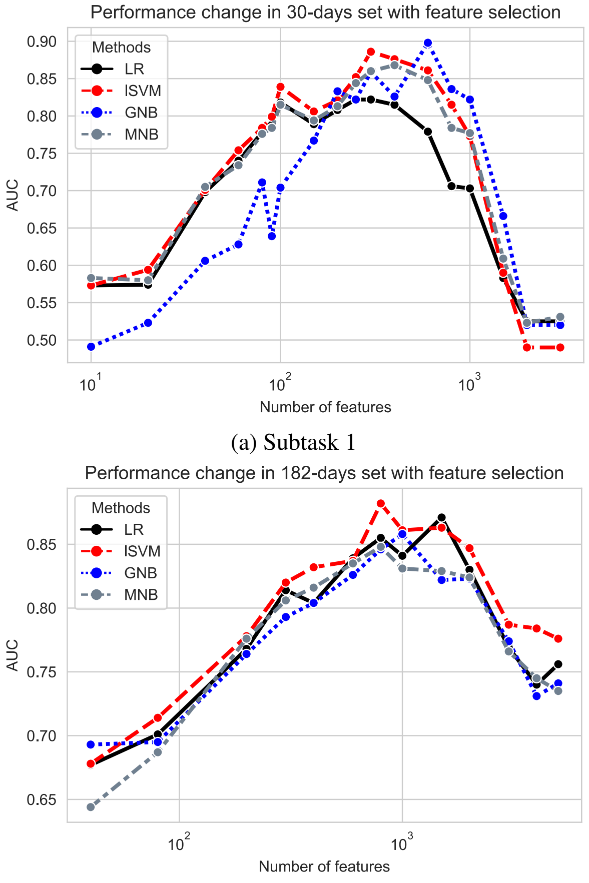
As most ML methods depend on learning from features, we select n-gram features where $n \leq 2$ for their popularity in suicide studies (O'Dea et al., 2015; De Choudhury et al., 2016; Pestian et al., 2020). For bigrams ($n = 2$), we apply a sliding window over concurrent words using the NLTK library (Bird et al., 2009). Next, we eliminate infrequent n-grams from the training set to reduce uninformative features (occurring in $\leq$3 tweets in 30-days, $\leq$10 tweets in 182-days training sets). Subsequently, we scale the features by row-normalizing them with the root of the sum of the square (i.e. variation) of the feature values.

Among the popular ML methods in suicide literature is logistic regression (LR) (Walsh et al., 2017; De Choudhury et al., 2016; O'Dea et al., 2015). We select the "liblinear" solver with default settings for being recommended for small datasets (Buitinck et al., 2013). To cover diverse mathematical frameworks and assumptions, we also include two naive Bayes methods (Gaussian (GNB) and Multinomial (MNB) with default settings) (Buitinck et al., 2013). We also experiment with K-Nearest Neighbors with different distance (uniform, weighted) and neighborhood (k ∈ {3, 5, 8}) settings, but we eliminate it for low within-dataset results. Similarly, ensemble-learning methods (Adaboost, XGBoost, Random Forest) also return underwhelming performance despite the parameter tuning, and thus, were eliminated. Additionally, we evaluate support vector machines (SVM) for their popularity in suicide research (Zhu et al., 2020; Pestian et al., 2020; O'Dea et al., 2015). SVM with rbf kernel proves to be successful but requires costly parameter tuning, while linear SVM (lSVM) shows success on within-dataset evaluations with less cost. Consequently, we select lSVM of sklearn (default settings) for the shared task (Buitinck et al., 2013), which returns only binary classification results. To convert them to probabilities, we apply probability calibration with logistic regression (CalibratedClassifierCV).

Feature selection: Following the ML method selections, we evaluate the effect of feature selection on ML performance. To compute feature importance scores, we also use the LR. For each selected number of features, we gather top suicidal and control features. Next, we train and evaluate the ML methods in a leave-one-out (LOO) framework using those features. The feature selection results of the selected ML methods for two subtasks are in Figure 2. We select the best ML models from these plots.

Experiments with Ensemble: Ensemble classifiers previously showed success in ML challenges (Niculescu-Mizil et al., 2009). Since every classifier renders predicted probabilities for every data point, we build an ensemble classifier to optimize the results of four selected ML methods (LR, GNB, MNB, lSVM). We adopt a weighting ensemble method where the weight of each classifier is set proportional to its performance (Rokach, 2010). We call this method weighted Ensemble (wEns).

Experiments with DL: To measure whether re-



(a) Subtask 1

(b) Subtask 2

Figure 2: Feature selection evaluations on the labeled datasets of two subtasks.

sults would improve with complexity, we also evaluate shallow DL methods. We use the pre-trained transformer model Bert-base-uncased (Devlin et al., 2018) to catch the linguistics features of the tweets. The embeddings are then fed to a DL Recurrent Units-based architecture to learn text sequence orders. We experiment with two types of recurrent neural networks (RNNs): Long Short Term Memory (LSTM) (Gers et al., 1999), and Gated Recurrent Unit (GRU) known for overcoming vanishing and exploding gradient problems faced by vanilla RNNs during training (Cho et al., 2014). After assessing various configurations of both architectures, we settle on a multi-layer bi-directional GRU with the following characteristics: embedding dimension=256, number of layers=2, batch size=32. We call this model GRU-Bert. We include a drop-out to regularise learning and a fully connected layer with a Sigmoid activation to produce the classification for each tweet. Finally, we include the same majority voting framework to infer the classification on the user level. We use Pytorch (Paszke et al., 2019) and scikit-learn (Buitinck et al., 2013) libraries for implementation.

3 Results

Before training each classifier, we employ the best performing top features from the Figure 2, where every classifier has its most fitting top features for each subtask. Next, we construct a LOO cross-validation framework for within-dataset evaluations.[1] It is important to note that, in each step of the LOO, we choose new user ids for evaluation and completely exclude all of their tweets from the training sets to evade ML methods potentially learning the way a person drafts tweets. That means the within-dataset LOO results of a subtask are reported for all users of the labeled set. Moreover, the labeled datasets have more users than the unlabeled test sets per subtask (e.g. 57 vs. 11 suicidal users in subtask1). Ergo, we expect a high magnitudinal difference between the within-dataset and the test results.

Table 1: Within-dataset evaluation results.

	F1	F2	TPR	FPR	AUC
Subtask 1: (30 days)					
LR	78.0	81.6	84.2	31.6	80.8
GNB	81.2	88.8	**94.7**	38.6	89.3
MNB	83.1	84.8	86.0	**21.0**	86.8
lSVM	81.9	87.2	91.2	31.6	88.6
wEns	**85.0**	**90.6**	**94.7**	28.1	**93.2**
GRU-Bert	81.2	82.2	83.1	**21.7**	84.0
Subtask 2: (6 months)					
LR	**81.9**	83.9	85.4	**23.2**	85.5
GNB	69.6	83.0	**95.1**	78.0	81.5
MNB	75.7	77.1	78.0	28.0	82.8
lSVM	78.6	87.1	93.9	45.1	84.6
wEns	**81.7**	**88.0**	92.7	34.1	**88.5**
GRU-Bert	74.5	75.4	76.0	28.6	77.5

The within-dataset evaluation results of the selected methods are in Table 1. For subtask 1, we obtain the best LOO cross-validation score from the wEns method that combines the results of four ML methods (LR, MNB, GNB, lSVM) in a way that improves the results obtained from each of them. Meanwhile, GRU-Bert and MNB return the lowest false positive rates (FPR) for this subtask,

[1] Within-dataset evaluation results of the selected ML and weighted ensemble methods are obtained from LOO cross-validation. While for GRU-Bert, collections were split into training-validation-test sets in 70:10:20 ratios.

which might be a critical rate to consider in real-life applications in social media domains. LOO results of subtask 2 in Table 1 show that wEns returns the best scores for the longer-spanning dataset as well, where LR returns the best FPR, and GBN returns the highest true positives rate (TPR).

Table 2: Test results over unlabeled data and the results from the baseline method of CLPsych2021.

	F1	F2	TPR	FPR	AUC
Subtask 1: (30 days)					
Baseline	63.6	63.6	63.6	**36.4**	66.1
LR	63.6	63.6	63.6	**36.4**	**74.0**
wEns	**69.2**	**76.3**	**81.8**	54.5	70.2
Subtask 2: (6 months)					
Baseline	**71.0**	**72.4**	**73.3**	**33.3**	**76.4**
LR	64.5	65.8	66.7	40.0	56.9
wEns	59.5	67.1	**73.3**	73.3	58.2

Based on the LOO results, we select three different methods we were allowed to submit for the evaluation of the test set: LR, wEns, and GRU-Bert. We choose LR and wEns for their high performance on LOO experiments, while we select GRU-Bert for measuring how a DL method would generalize over the test sets. The baseline classifier provided by the organizers is also a logistic regression. However, it performs the classification over merged tweets of users - therefore is different from our implementation of LR. In Table 2, wEns appears to provide the best F1, F2, and TPR scores over the test set of subtask 1, while our LR outperforms the AUC of the baseline method. While these methods show the success of generalizability on the 30-days test set, the results are not that successful for subtask 2. The wEns method performs the same as the baseline in terms of TPR, but the rest of the scores are lower than the baseline results.

4 Discussion

In subtask 1, the test set results show that feature selection can considerably enhance the performance of ML models compared to the baseline. We also find that the ensemble classifier is comparably better than the baseline in this subtask. Meanwhile, though the baseline of CLPsych2021 is the same as our LR, our additional MV and feature selection together enable LR to substantially outperform the baseline. These successes of simple ML methods indicate that a collection of tweets from within the 30-days of a suicidal event is good enough to capture the existence of suicidal ideation, which is an important finding for future real-life suicide prevention applications.

In contrast to the observations from subtask 1, our test results on subtask 2 are unsatisfactory. Yet, they provide the valuable insight that suicidal signals are more significant in the short-term, and older tweets lacking suicidal ideation generate noise. This insight suggests the need to account for a time-domain aspect. To investigate the viability of this claim, we experiment with a simple time-decay coefficient in the MV framework and evaluate it through LR on the test set. We multiply each vote by the coefficient $2^{\frac{-timeDiff}{halfLife}}$ where $timeDiff$ is the number of days between the current and last tweets, and $halfLife$ (=7 days) is a hyperparameter that reflects the weight of a vote in the final suicide risk score of a user. Initial experiments show that even this simple time-decay coefficient improves the test results significantly. This observation suggests that tweet dates are critical features for this subtask and should be included in future work.

Notwithstanding, on both subtasks, the shallow DL methods we experimented with perform poorly. These results could be attributed to overfitting on the small dataset and noise sensitivity for the larger time-spanning dataset. Additionally, regardless of the dataset size, these methods proved to be computationally expensive. As within-dataset experiments using simple ML methods outperformed these expensive shallow DL methods, we excluded the latter from the test set evaluation. Future work on DL will include deeper, more complex, and noise immune methods that could integrate Convolutional neural networks (CNN), deeper LSTM or GRU layers, and experiments with various word embedding models.

If we compare our findings with those in Coppersmith et al. (2018), we observe different results in terms of short-term versus long-term dataset classifications. We attribute these different outcomes to the fact that the original study optimizes the design for detecting trait-level (relevant to risk for any point in time) suicide risk when we endeavor to identify suicidal ideation at the state level (immediate risk presence). This design choice, along with tweet-level classification, enabled our model to recognize suicidal nuances in short-term tweets. Meanwhile, we were unable to detect any suicidal ideation through manual inspection (reading and interpreting the tweets) over most of these tweets due to their noisy and ambiguous nature.

5 Conclusion

In this shared task, we investigate various models for identifying suicide risk based on user's tweets. Inspired by real-life applications, we focus on assessing suicide risk on the tweet level. Experimental results reveal that the ensemble classifier can identify suicidal users from 30-days tweets with a high performance rate, demonstrating the power of majority voting over tweet-level classifications for short-term suicide risk detection. Meanwhile, we construe from the underwhelming results on the six-month dataset that these models were more sensitive to the signals relevant to short term risk than those relevant to long term risk. In future work, we will incorporate a temporal aspect to improve the noise immunity of our models, and we will continue experimenting with more complex models.

Ethics Statement

Secure access to the shared task dataset was provided with IRB approval under University of Maryland, College Park protocol 1642625.

Acknowledgements

The organizers are particularly grateful to the users who donated data to the OurDataHelps project without whom this work would not be possible, to Qntfy for supporting the OurDataHelps project and making the data available, to NORC for creating and administering the secure infrastructure, and to Amazon for supporting this research with computational resources on AWS. The authors are thankful to the anonymous reviewers for their constructive comments and valuable suggestions.

References

Derek Anderson. 2019. wordninja Python library. https://github.com/keredson/wordninja. [Online; accessed 11-March-2021].

Ulya Bayram, Ali A Minai, and John Pestian. 2018. A lexical network approach for identifying suicidal ideation in clinical interview transcripts. In *International Conference on Complex Systems*, pages 165–172. Springer.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Glen Coppersmith, Ryan Leary, Patrick Crutchley, and Alex Fine. 2018. Natural language processing of social media as screening for suicide risk. *Biomedical informatics insights*, 10:1178222618792860.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Pedro Domingos. 2012. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87.

Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. 1999. Learning to forget: Continual prediction with lstm. In *9th International Conference on Artificial Neural Networks: ICANN '99*, pages 850–855. IET.

Shaoxiong Ji, Shirui Pan, Xue Li, Erik Cambria, Guodong Long, and Zi Huang. 2020. Suicidal ideation detection: A review of machine learning methods and applications. *IEEE Transactions on Computational Social Systems*.

Jianhong Luo, Jingcheng Du, Cui Tao, Hua Xu, and Yaoyun Zhang. 2020. Exploring temporal suicidal behavior patterns on social media: Insight from twitter analytics. *Health informatics journal*, 26(2):738–752.

Sean Macavaney, Anjali Mittu, Glen Coppersmith, Jeff Leintz, and Philip Resnik. 2021. Community-level research on suicidality prediction in a secure environment: Overview of the CLPsych 2021 shared task. In *Proceedings of the Seventh Workshop on Computational Linguistics and Clinical Psychology (CLPsych 2021)*. Association for Computational Linguistics.

Alexandru Niculescu-Mizil, Claudia Perlich, Grzegorz Swirszcz, Vikas Sindhwani, Yan Liu, Prem Melville, Dong Wang, Jing Xiao, Jianying Hu, Moninder Singh, et al. 2009. Winning the kdd cup orange challenge with ensemble selection. In *KDD-Cup 2009 Competition*, pages 23–34. PMLR.

Bridianne O'Dea, Stephen Wan, Philip J Batterham, Alison L Calear, Cecile Paris, and Helen Christensen. 2015. Detecting suicidality on twitter. *Internet Interventions*, 2(2):183–188.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

John Pestian, Daniel Santel, Michael Sorter, Ulya Bayram, Brian Connolly, Tracy Glauser, Melissa DelBello, Suzanne Tamang, and Kevin Cohen. 2020. A machine learning approach to identifying changes in suicidal language. *Suicide and Life-Threatening Behavior*, 50(5):939–947.

Lior Rokach. 2010. Ensemble-based classifiers. *Artificial intelligence review*, 33(1):1–39.

Arunima Roy, Katerina Nikolitch, Rachel McGinn, Safiya Jinah, William Klement, and Zachary A Kaminsky. 2020. A machine learning approach predicts future risk to suicidal ideation from social media data. *NPJ digital medicine*, 3(1):1–12.

Michael Mesfin Tadesse, Hongfei Lin, Bo Xu, and Liang Yang. 2020. Detection of suicide ideation in social media forums using deep learning. *Algorithms*, 13(1):7.

Colin G Walsh, Jessica D Ribeiro, and Joseph C Franklin. 2017. Predicting risk of suicide attempts over time through machine learning. *Clinical Psychological Science*, 5(3):457–469.

H Zhu, X Xia, J Yao, H Fan, Q Wang, and Q Gao. 2020. Comparisons of different classification algorithms while using text mining to screen psychiatric inpatients with suicidal behaviors. *Journal of psychiatric research*, 124:123–130.