Screening for Suicidality in Reddit Users

Dylan Fox
dylfox21@gmail.com

Emily Gong
egong@umd.edu

Vincent Hsaio
vhsiao@umd.edu

Hadi Vafaei
vafaii@umd.edu

I. Introduction

In this paper, we explore the task of suicidality screening of Reddit users. Without taking into account explicit posts in the subreddit r/SuicideWatch or any other mental health related subreddits, we hope to identify meaningful semantic patterns which indicate that an individual may be at an increased risk of suicide. This was formalized as a modified shared task for the 2019 Workshop on Computational Linguistics and Clinical Psychology (CLPsych '19) as *Task C* (Zirikly et al. 2019). Our group is comprised of Dylan Fox, Emily Gong, Vincent Hsiao, and Hadi Vafaei. Although many of the tasks in this project overlapped significantly, Dylan focused on configuring a full pipeline, experimenting with various classifiers for post-level suicide risk classification, and solving data imbalances. Dylan also laid the groundwork for some of the report. Emily worked on user-level aggregation, explored temporal features, and rewrote the Jupyter Notebook for a clean experience. Vincent focused on LDA feature extraction, data-preprocessing and filtering, and some background research via literature review. Hadi experimented with neural features for post level classification.

For Task C, we decided to address this problem via a two-step process. We first classify each Reddit post individually, then proceed to aggregate the results of this post classification to determine the suicide risk for a single user. This is further discussed in the methods section. This method contrasts with the method present in (Shing et al. 2018), in which feature vectors were aggregated across posts and averaged to obtain a single feature vector for each user. This feature is then used to classify this user. Our decision to classify posts individually and aggregate results has practical advantages in a realistic screening scenario in which someone might want to trace back to specific posts that were flagged as indicating suicidal thoughts. Although Task C is undoubtedly the most difficult task present in (Zirikly, 2019), we believe that using posts outside of r/SuicideWatch and other mental health subreddits to be extremely interesting and impactful. We hope to gain some insight into how these difficult problems are solved in real-world scenarios, and hope to have some impact in the work of screening for suicide risk among social media posts.

II. Data and Methods

A. Dataset

The dataset we utilize for analysis is derived from (Shing et al. 2018). This dataset contains the Reddit posts from 2005 to 2015 of all users who had previously posted on the r/SuicideWatch subreddit: a forum which users could post about suicide related topics. These users are potentially at more risk of suicide than the average Reddit user, and this dataset was curated to help in identifying language in social media posts that could be indicative of suicidal thoughts or intent. Additionally, this dataset includes a set of control users who have never posted to r/SuicideWatch. Shing et al. obtained both expert and crowd-sourced annotations of suicide risk levels for these Reddit users on the following scale: (a) No Risk, (b) Low Risk, (c) Moderate Risk, (d) Severe Risk. A user assessed as No Risk may have posted to r/SuicideWatch in order to provide help to those in need or to ask for advice about a friend, without experiencing any suicidal thoughts themselves. A user with low risk expresses some factors which suggest suicide risk, but doesn't seem to be at immediate risk of suicide. A user with moderate risk shows indications that there could be a risk of this person attempting suicide. A user with severe risk shows numerous indications of suicidal intent, and they are at high risk of attempting suicide in the near future.

The dataset used in our project includes all posts of 934 users who have posted to r/SuicideWatch, in addition to the posts of an equal number of users who have not posted to r/SuicideWatch or a related mental health subreddit.

B. Data Processing

1) Data Filtering: For the purpose of Task C, we filter out all posts from the prespecified list of mental health subreddits in section 3.3 of the project description (though we did notice during error analysis that there are several mental health subreddits that do not seem to be included in this list i.e. r/OffMyChest, r/NeedAFriend, r/BipolarReddit, etc). Additionally, we acknowledge that an individual's suicidal thoughts are not an intrinsic part of their personality, but rather a symptom of severe depression that is occurring during a specific period in one's life. To reflect this, we further subset our data to only include posts that occur within 2 weeks of a user posting to r/SuicideWatch. We believe that including all of a user's posts during the 10 year span that data was collected would bias any classifiers attempting to distinguish suicidal risk, as posts created significantly far away from a difficult time in a user's life will often not indicate suicidal intent. We did not subset any posts from control users, and preserved them within our dataset regardless of timestamp. Additionally, through exploratory data analysis, we discovered that around 65% of all reddit posts in this dataset contained no post body, indicating that these posts contained

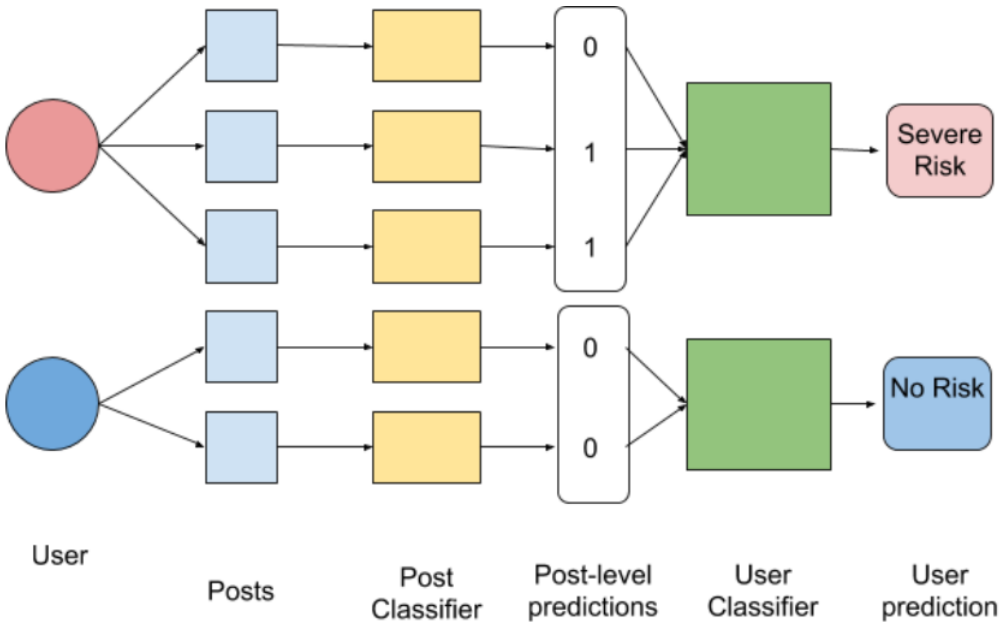

Fig. 1. Two-Step Hierarchical User Classification System

simply a title and an associated image or link. We made the decision to remove these posts from the dataset as well, which significantly reduced the number of posts with which to train our classifiers, but also significantly reduced noise within our dataset.

Total Posts	9326
Control Posts	7462
No Risk (a)	258
Low Risk (b)	138
Moderate Risk (c)	342
Severe Risk (d)	1126

TABLE I
NUMBER OF POSTS IN CROWD-SOURCED TRAINING SET AFTER FILTERING

2) Text Preprocessing: Reddit serves a predominantly English language community. Unlike some social media platforms such as Twitter, constraints on post length are not significant enough to make posts deviate from standard English. As a result, we decided that the standard English sentence and word tokenizer in the nltk library would be sufficient for parsing text in Reddit posts. Like similar systems using the nltk parser, we also remove English stopwords from each post before transforming it into a list of word tokens (https://www.nltk.org/nltk_data/).

C. Methods

Once the dataset has been filtered, we train a classifier on the binary classification task of determining if a user is at Severe Risk of attempting suicide in the near future. In doing so, we attempt to find a decision boundary between individuals in suicide risk classes (a), (b), (c) and users in class (d) based on their Reddit posts to non-mental health subreddits around the time of their post to r/SuicideWatch. All posts drawn from users in the control set are likewise grouped into the No Risk - Moderate risk category, as these individuals were chosen to not have any posts in mental health related subreddits.

As described in the introduction, we view this user classification as an aggregate task, as suicide risk as been annotated on the user level, but text data is provided at the post level. We propose a classification method which first trains a primary classifier on features extracted from individual posts, with ground truth labels adopted from the risk level of the user which authored the post. The individual post classification can thus be interpreted as a weakly supervised task, as posts have not been directly annotated for suicidality risk, and not all posts from a user with severe suicidal risk will indicate this risk. As these labels are noisy, we attempt to find the best possible binary decision boundary that classifies each post as exhibiting suicidal risk.

Once a weakly supervised classifier has been trained on individual posts, we are tasked with aggregating the classifications of a individual's posts to classify the risk level of the user. If each post was classified correctly (which is extremely unlikely due to the circumstances described above), then all posts from a given user would indicate that user as either severe risk or not severe risk.

D. Feature Extraction

In order to train the post classifier, natural language features must be extracted from each post which accurately capture user suicidal risk within a document. We hope to capture

semantic features that are indicative of suicidal thoughts or tendencies, such as the use of words which correlate to topics relating to depression, mental illness, or grief. As we do not append the titles to the post text, all features are extracted from only the post body. We experimented with several possible features, including:

1) **Bag-of-Words**: A vocabulary over the training set was generated, and bag-of-words features were collected for each individual post. Due to system memory constraints, it became infeasible to use these large feature vectors in base form, especially for offline classifiers (which require viewing all the data at once). Therefore, we perform dimensionality reduction using incremental PCA (Artac et al. 2002) as implemented in scikit-learn. This allowed us to reduce the these sparse bag-of-words features to dense embeddings of size 40 in an online fashion.

2) **Empath**: We use the frequency vector of 194 default Empath lexical categories (Fast et al. 2016), as implemented by the python library *empath*. We additionally ensure each vector is normalized.

3) **LDA and sLDA Topic Model Posteriors**: To obtain posterior topic distributions, we utilize both Latent Dirichlet Allocation (Blei et al. 2003) and Supervised Latent Dirichlet Allocation (Mcauliffe et al. 2008) to obtain 40-topic models over the training set. These implementations are found in the python library tomotopy (https://bab2min.github.io/tomotopy/v0.7.0/en/). We establish a corpus by treating each post as a separate document, and utilize topic posteriors for our feature vectors. In the case of sLDA features, our posterior was inferred with help of a binary response variable associated with each post that denoted if the post's author was classified as severe risk (+1) or not severe risk (0). The model is then used to generate feature vectors for each post.

Top 10 Words Within Topic	Response Variable Regression Coefficient
url, us, min, survey, gt95, person, short, study, answer, minutes	-5.501
person, amazon, power, atx, video, build, cpu, x, cooler, motherboard	-2.781
person, man, life, god, story, us, words, church, world, save	-1.012
gt, person, need, c, 2, end, find, list, file, model	-0.259
im, dont, know, like, want, get, feel, really, cant, ive	1.589

TABLE II
EXAMPLE OF SLDA TOPICS DRAWN FROM 40-TOPIC MODEL

To determine how many topics to use, we plot the held-out word log likelihood of the trained sLDA for different numbers of topics. This can be seen in Figure 2.

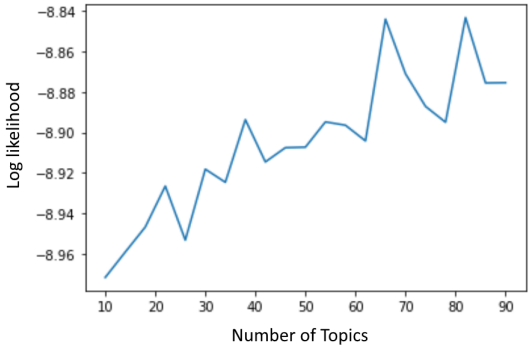

Fig. 2. sLDA Log Likelihood

We encountered that using a higher number of topics resulted in a better likelihood score even up to 70 or 80 topics, suggesting that the discourse on Reddit is highly varied. However, when incorporated into the post classification pipeline, sLDA models with a higher number of topics actually end up causing our post classifiers to begin over-fitting significantly. As a result, we struck a balance by choosing 40 topics.

E. Post Level Classifiers

To classify posts, we experiment with a series of classification models. After filtering the posts as described above, we observe a large class imbalance, with only around 20% of posts coming from individuals labeled as severe risk. To address this, we incorporate random oversampling of the minority class (d) to balance the classes. We implement this through the RandomOversampler module in the imbalanced-learn python library. This class balancing significantly improved our results on several classifiers. The following classification models were explored, each implemented in scikit-learn:

1) **Logistic Regression:** Logistic regression classifier with regularization value C = 0.2, chosen using a hyperparameter grid search over C = [0.1,0.2,0.5,1,2] and 5-fold cross validation on the training set.

2) **Random Forest**: Random Forest classifier with 200 random trees, max depth of 50 nodes, and $\sqrt{n}$ features considered when looking for best split. These hyperparameters were chosen using a grid search over max_depth = [10,20,50,100,200], num_trees = [20,50,100,200] and 5-fold cross validation on the training set. Default value was used for num_features.

3) **AdaBoost**: AdaBoost ensemble classifier with 1000 estimators. A decision stump (decision tree with depth=1) was used as the base estimator. Hyperparameters were chosen using a grid search over num_estimators = [100,200,500,1000] and 5-fold cross validation on the training set.

4) **Support Vector Machine**: SVMs with both linear and radial basis function (RBF) kernels were used as classifiers. Regularization parameter C was chosen to be 1 through hyperparameter grid search over C = [0.1,0.5,1,2] and 5-fold cross validation on the training set.

5) **Multi-Layer Perceptron**: MLP with 3 hidden layers of size 64, run over 500 epochs (or until convergence) with 'relu' activations. This classifier was optimized with the adam optimizer and a learning rate of 0.001. Model architecure (number of hidden layers and size of hidden layers) was determined through grid search and 5-fold cross validation over the training set.

F. User Level Aggregation

During inference, we collect all of an individual's posts that occur within 2 weeks of them post to r/SuicideWatch. If a user did not post to r/SuicideWatch, we do not filter any posts in this way. Additionally, for all users, we filter out any posts from mental health related subreddits, including r/SuicideWatch, as well as posts which contain no post body text (such as images, links, etc.) We then use our trained post classifier to classify each reddit post as Severe Risk or non-Severe risk. To retrieve a classification for the user, we perform argmax on these aggregated classifications, such that a risk classification is chosen if greater than 50% of an individual's posts fall into that category.

III. Evaluation

Prior to training, we establish a train-dev-test split within our data to reduce the bias of overfitting on our final results. We train our model on the crowd annotated training set for Task C as defined in Zirikly et al. We use the crowd annotated test set defined in the same paper as our development set, which allows us to select choice hyperparameters. Lastly, we utilize the expert annotated set as our test set in order to evaluate our final results. As described below, we do conduct some evaluation on the development set, as we believe it to be comparative on a high-level to the Task C results in Zirikly et al. While not many hyperparameters were chosen through use of this hold-out set, one must still note that selection of these parameters could cause slight overfitting on this dev set.

A. Results

To evaluate our post classifiers and feature choices, we recorded a set of 5 performance metrics (accuracy, recall, precision, f1-score, and ROC-AUC) on the down-stream task of user classification. We collected metrics on both the held-out (dev) crowd-sourced test set and the expert annotated set. The evaluation of our user-level classification on the crowd annotated test set can be found in Table V, while the evaluation of our user-level classification on the expert annotated set can be found in Table VI. As most of our classifier hyperparameters were chosen through cross-validation on the training set, we believe our results for the crowd-sourced annotations to be presentable for high-level comparison with Task C results from Zirikly et al. We must note that the task as defined in Zirikly at al. did not incorporate posts from control users, so these scores are not directly comparable.

Two of our user-level classifiers performed with an f1-score of over 0.4 on this set: the AdaBoost classifier and MLP using sLDA features. Furthermore, each classifier performed best on this set when utilizing either LDA or sLDA features. We can also observe how some classifiers, such as the SVMs and Logistic Regression, were able to maintain high performance regardless of the feature type, while other classifiers, such as the ensembling methods and multi-layer perceptron, show disparities between performance with each feature type.

We also evaluate our user-level classifiers on the expert annotated set and receive interesting results. The performance on this set is remarkably lower than the performance on the hold-out crowd annotated set. We attribute this to the significant differences in annotations between the crowdsourcers and the experts. This result was truly shocking, as we did initially did not believe the disparity between the distribution of crowd-sourced annotations and expert annotations to be this significant. Through extensive error analysis, hypothesize that non-expert annotators were more lenient with giving "Severe Risk" annotations than the highly trained experts, who would often put these individuals in classes (b) or (c). Thus, when our binary classifiers were trained on the crowd-sourced data and tested on expert annotated data, they often misclassified individuals who were not assessed by experts to have high risk, but rather assessed by crowdsourcers to be in this category. On this set, our best performing classifier, the Linear SVM using LDA features, receives an f1-score of 0.2134 - less than half the score of the best performing classifier on the crowd-sourced set. We again see that the each classifier performs best while using either sLDA or LDA topic features.

From the performance of our user-level classifiers on both the crowd annotated hold-out set and the expert annotated set, we can make several conclusions. In the context of this data, these classifiers can be used to screen for and flag users who may be experiencing suicidal thoughts. Therefore, we can conclude that we would like to prioritize false positives over false negatives, as missing someone who may attempt suicide in the near future is much worse than flagging someone who does not have any suicidal thoughts. With this in mind, it

is helpful to look at the recall of classifiers, which specifies what proportion of the population in question you correctly identified. We consistently observe that linear classifiers such as the Logistic Regression and Linear SVM classifiers perform better in terms of recall, compared to non-linear models such as the MLP or Random Forest (which often have higher precision). Therefore, linear models may unnecessarily classify individuals who are not suicidal, but they will also classify a larger proportion of those who truly are. One can argue that the Linear SVM, which had the best recall on the crowd-sourced hold-out set, may be a better model for this task of suicide risk screening than the AdaBoost classifier, which received the highest f1-score.

As stated before, we observe that the best performing classifiers on both sets used feature vectors based on LDA and sLDA topic posterior distributions over the test set. We were not surprised to see LDA and sLDA features perform better than the bag-of-words baseline, but were unsure abour how Empath features would compare. We believe that ability for Latent Dirichlet Allocation features to capture a wide variety of topics that are *learned from data* - compared to Empath's predefined set of lexical categories - allow it to generalize much better to unknown data from the distribution it was inferred on.

Additionally, we observe that due to the massive class imbalance present in the data, the accuracy metric as reported on both sets does not tell us much information about the performance of the classifier. If a classifier reported "Not Severe" for all users, we would still receive around 75% accuracy for either set. Thus, more descriptive metrics such as recall, precision, and f1-score can be used to determine the performannce of a classifier.

B. Error Analysis

During the development of our classification system, we perform error analysis on the formative evaluations that were run while tuning our model. One avenue that we pursued was to analyze the topic distributions of the posts of misclassified users in the downstream task.

Over the course of these iterative formative evaluations and subsequent error analyses, one phenomena we encountered was that the final post and user classification performance metrics for our hold-out (dev) set depended significantly on what topics were used to acquire a posterior topic distribution. Since sLDA relies on Gibbs sampling, the topics obtained during training are stochastic. Additionally, hyperparameters such as the number of topics can greatly alter the topics that are obtained. A key issue that we encountered due to this was that the presence of some topics actively decreased the observed performance of the post and user classifiers. Table III provides a short example of two topics that were obtained after training the sLDA model with 60 topics, each of which have a different (detrimental or helpful) effect on downstream classification performance.

Topic Number	Top 10 words	Response Variable Regression Coefficient
1 (Detrimental)	2, find, force, mass, x, person, solve, problem, ring, b	1.658
2 (Helpful)	im, dont, feel, want, know, life, cant, like, ive, get	3.143

TABLE III
EXAMPLE OF SLDA OVERFITTING WHEN TRAINED ON 60 TOPICS

The top words in topic 2 consist of many self-referring words as well as many words with negative sentiment. In contrast, topic 1 consists of words that are related to math problems. It would be expected that a topic such as topic 2 would have a high value for the suicidality response coefficient but it is surprising that topic 1 also has a high value. During development, it was noticed that if topic 1 was generated as one of the topics present in the model, classification performance would decrease on our validation set indicating a problem of generalization.

Further error analysis on this example revealed that one user in the training set has many of the posts with the highest inferred distributions in both topic 1 and 2. A quick analysis of their post history shows an interleaving of posts on various subreddits such as r/SuicideWatch and r/LearnMath. Due to their high post count, this adds additional bias to our sLDA model and subsequent post classifier. As a result, the math related topic is used as an indicator for suicidality in our post classifiers when it probably should not be. The presence of these topics that describe posting interests (unrelated to suicidality risk) of severe risk users exacerbates the disadvantages of our weakly supervised approach.

This problem can be seen as analogous to overfitting but it is not quite the same. Overfitting is commonly expressed as when models learn noise instead of a signal present in the data. However, our post classifiers end up learning a real pattern in the training data of the characteristics of severe risk users (ie: users posting for math help in the training set are more likely to be severe risk). Unfortunately, this learned pattern does not generalize well to other datasets. As mentioned above, we can reduce the frequency of these topics by reducing the total number of topics when training an sLDA model.

However, a model that is trained with fewer topics may not adequately represent discourse in a social media site such as Reddit where the topics of discourse are as widely varied as the number of subreddits. In the future it might be beneficial to do something like topic pruning where initially a sLDA model is trained with a large amount of topics and then topics that are actively harmful for generalization are pruned.

C. Ethical Issues

As mentioned in class, this project does not involve human subject research in the way that an IRB review is needed. However, it is still understood that there is a risk of user identification which is something to be conscious of as a researcher looking through the data. Since we are not interacting with humans directly, it is impossible to get informed consent - but it is possible to act with responsibility. There is current debate on whether it is ethical to use someone's data for a purpose they had not intended for. User Intervention is another concern of the IRB, but for the purpose of this project we do not make any attempt to contact users. The data of the users may be publicly available but it is well understood that this data is sensitive. All members of the group stored the data on a password protected device and will take measures to remove, refrain from sharing the data, or discuss the protocol of future use of the dataset with Philip Resnik outside the scope of this class. The dataset was preprocessed in relation to de-identification which reinforces the anonymity of users which could be dangerous in the wrong hands. We also ensure that the report only summarizes and/or paraphrases a user's post in our analysis and discussion making sure never to express the original post in verbatim so it is harder to trace back the post to a specific user. Lastly, we did not scrape any of the links posted and thus remained within the scope of the dataset curated for this sensitive task.

IV. Experiments with Deep Models

In addition to the methods described above, we also performed some preliminary experimentation with deep natural language models to see how they performed on this task. The choice of modeling approach depends on the end goal: deep models are usually more flexible and less interpretable than shallow, statistical models. Therefore, deep learning is often preferred when achieving higher performances is more important for a given application.

In this section, we will report the results for training a deep classifier using BERT, a transformer based pretrained language model.

A. Training

We used BERT (Devlin et al. 2019) to extract contextualized deep features from the data and trained a linear classifier on top of those features. We jointly trained BERT weights, as well as classifier weights for the best performance. However, we used a much smaller learning rate for BERT ($lr = 1e-5$) compared to the linear classifier ($lr = 1e-3$). This is to prevent catastrophic forgetting that sometimes happens when using transfer learning.

B. Results

The performance scores are summarized in table IV below[1].

Data	Accuracy	Precision	Recall	F1-score
Filtered / Crowd	0.7642	0.6485	0.7640	0.6932
Not Filtered / Crowd	0.7935	0.7559	0.7935	0.7195
Filtered / Expert	0.7918	0.7329	0.7920	0.7523
Not Filtered / Expert	0.8000	0.6700	0.7953	0.7293

TABLE IV
DEEP MODEL PERFORMANCE ON USER CLASSIFICATION TASK

C. Visualizing attention

BERT is a transformer based model. The attention weights are the core building blocks of transformers, and sometimes they attend to meaningful features in data. We explored this by pushing data through the model and visualized the attention weights using BertViz (Vig 2019). Please see Fig. 4 for an example of a normal user classified as no risk, and Fig. 5 for users classified as severe risk. Figures are in the Appendix below.

V. Discussion

A. Working with Real World Data

There are several notable aspects of this project which we can discuss. One main aspect being the significant noise of using real-world social media data. Given that we did not have ground truth labels for individual posts, we were forced to infer that all posts from a user expressed that individual's suicide risk. As not all posts from a user will indicate suicidal thoughts, this incorporated a large bias into our post classifier.

Additionally, during development we noticed that for the purpose of post classification, the control data was also significantly noisy. When displaying the top 10 documents with the highest suicide risk coefficient as inferred through sLDA, there are typically a few control posts that are also included. An example of such a post when paraphrased is:

> I need help. My family doesn't help me. Since I was born, they've never helped me. I think dad is going to make me marry someone who doesn't listen to me. I'm getting scared and aggressive. [help with exaggerated spelling] plz. My head really pains, I have stress too, I'm going mad too.

If this post was encountered during suicide risk screening scenario, a human could be led to flag this post due to the presence of an isolation from family suicide risk factor. However, since it is drawn from the control set, it is grouped with the low to moderate risk categories in the input to our post classifiers.

[1]Note that these performance scores are not comparable to results from tables V and VI since the data used for training deep models wasn't limited to a two week period centered around users posting to r/SuicideWatch. Due to limited resources, we were not able to train deep models on the same data used for other models in time.

B. Exploratory Data Analysis

The original project description included exploratory data analysis as an entirely separate task from predictive analysis. However, we believe EDA is a crucial part of understanding any domain specific task. We would have liked to explore more of the r/SuicideWatch posts in order to get a better understanding of the contrasting language use between individuals in different mental states. However, we were able to observe some interesting things through the data exploration that we did complete. Although we had assumed suicidality to be very time sensitive, we observed that some users may experience a lifelong suicide risk, based on an observed history of posts to mental health related subreddits.

C. Future Work

There are numerous other feature types and classifier models which we were not able to perform experiments with. In the future, it would be nice to experiment more with neural features, such as BERT, word2vec, or doc2vec, or with features that capture syntactic properties of the document. Additionally, we were not able to access certain types of features such as LIWC, as they were behind a proprietary paywall. Lastly, we were not able to experiment with combining features, and we believe this would result in a more robust classifier and better performance overall. However, we trained each post classifier on a single type of feature in order to form a comparative analysis between each type of classifier and feature type.

In addition to the argmax aggregation scheme implemented in our current model, multiple other thresholds were tested for user aggregation. We attempted to find other thresholds which could provide better performance than simply taking the classification of the majority of posts. This method method was eventually discarded as results were too inconsistent between classifiers and feature vectors. Instead, a more interesting approach would be to retrieve confidence scores for each of the posts scaled by some temporal weight. Temporal weight would increase as posts were closer to an r/SuicideWatch post and thus incorporates the sense that older posts may not be as relevant when screening, since suicidality is time sensitive. In Figure 3, we display the temporal posting features of one individual. Orange bars represent the posts to r/SuicideWatch or other mental health subreddits that we filtered out, while blue bars represent all of a user's posts regardless of the subreddit (inclusive of r/SuicideWatch and other mental health subreddits).

Another interesting feature to consider involves the frequency of posts. One hypothesis to make is that users would post more frequently on social media as their suicidal risk level increased, as call for help. We explored the data visually to see if suicide watch posts coincided with frequent posting activity. At first glance, we could not find much correlation between frequency of posts and significant decline in mental health as shown by posting on certain subreddits.

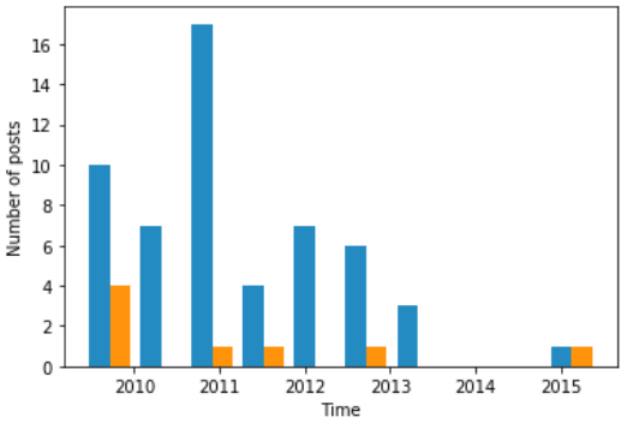

Fig. 3. Temporal Feature Exploration

However, there may be other ways that post frequency can be incorporated as a feature for classification tasks in the future.

Along these lines, it would be worth investigating patterns between the length of the post, time of day, and a measure of how self-oriented the post is. One could imagine that really late night posts might mean the user is exhausted, spilling out their words to express themselves and focused mainly on something that has happened to them. However, these are assumptions and certified psychologists may tell us otherwise. In addition, these assumptions may not be able to be captured well. There are plenty of late night owls using social media for distractions but who are not at severe risk.

D. Conclusion

In conclusion, we believe this project wass a significant learning experience for us as a group. This task involved dealing with real-world data that was not always clean or accurate, and required us to hypothesize various strategies of tackling a overarching task. We were surprised by many aspects of the data, notably the difference in annotation distributions between expert and crowd-sourced dataset which caused our classifiers to perform so much worse on the former set. However, we believe that we were able to do substantive comparative analysis between classification models and feature types in the context of this difficult task. Lastly, this task emphasized the importance of feature and model selection, especially when concerned with data such as this: where false negatives are so much more important than false positives.

VI. Final Statements and Signatures

1) We have read Benton et al. (2017).
2) We understand that privacy of the users and their data is critical, and absolutely no attempt can be made to de-anonymize or interact in any way with users.
3) We understand that this project is being done solely for educational purposes, and the results cannot be used directly in research papers. If we get promising results

and would like to develop the ideas into a research paper for publication, or to use what we have done further for another class, we will talk with Prof. Resnik about obtaining a suitable Institutional Review Board review. (It's not hard.)

4) We understand that we may not use these data for any purpose other than this specific class project. We will not show or share this data with anyone outside class, nor do any research or development on this dataset outside the scope of the class project. If there are things we are interested in doing with this dataset outside the scope of the class project, we will talk with Prof. Resnik.
5) We will store the dataset and any derivatives on computers that require password access. If we are working in an environment where other people can log in, e.g. a department server, we will set file permissions restrictively so that only you have access. You can also use group permissions limited to members of your group — but under no circumstances will data related to this project be world readable.
6) Any copies of the data or derivatives of it will be accompanied by a clear README.txt file identifying Prof. Resnik as the contact person and stating further redistribution is not to take place without contacting him first. If anyone we know is interested in the dataset, we will refer them to Prof. Resnik, rather than providing the data ourselves.
7) Once we have completed the project, we will delete any copy of the dataset we have made, including any derived files (e.g. tokenized versions of the documents).
8) We will not cut/paste any text content from this dataset into our project proposal, project writeup, onto the class discussion board, into e-mail, etc. If we want to identify a specific posting, e.g. in discussion on the class discussion board, we will use the ID from the dataset. If we want to give examples, we will create a paraphrase instead of the original text. For example, if a posting said What's this world come to? http://t.co/ XxI4QnMew we could change it to I wonder what this world has come to? http://t.co/ YYY. (Or just make up a post that demonstrates whatever it is you want to describe.)
9) We have deleted all copies of the project dataset.

Signed: Team H-DEV
Dylan Fox, Emily Gong, Vincent Hsiao, Hadi Vafaei

REFERENCES

[1] Artac, M., Jogan, M., & Leonardis, A. (2002, August). Incremental PCA for on-line visual learning and recognition. In Object recognition supported by user interaction for service robots (Vol. 3, pp. 781-784). IEEE.

[2] Benton, A., Coppersmith, G., & Dredze, M. (2017, April). Ethical research protocols for social media health research. In Proceedings of the First ACL Workshop on Ethics in Natural Language Processing (pp. 94-102).

[3] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), 993-1022.

[4] Fast, E., Chen, B., & Bernstein, M. S. (2016, May). Empath: Understanding topic signals in large-scale text. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (pp. 4647-4657).

[5] Mcauliffe, J. D., & Blei, D. M. (2008). Supervised topic models. In Advances in neural information processing systems (pp. 121-128).

[6] Shing, H. C., Nair, S., Zirikly, A., Friedenberg, M., Daumé III, H., & Resnik, P. (2018, June). Expert, crowdsourced, and machine assessment of suicide risk via online postings. In Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic (pp. 25-36).

[7] Sawhney, R., Manchanda, P., Singh, R., & Aggarwal, S. (2018, July). A computational approach to feature extraction for identification of suicidal ideation in tweets. In Proceedings of ACL 2018, Student Research Workshop (pp. 91-98).

[8] Zirikly, A., Resnik, P., Uzuner, O., & Hollingshead, K. (2019, June). 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 (pp. 24-33).

[9] Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, & Toutanova, Kristina BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) 4171–4186

[10] Jesse Vig A Multiscale Visualization of Attention in the Transformer Model. arXiv preprint arXiv:1906.05714

Post Classifier	Feature Type	Accuracy	Recall	Precision	F1-Score	AUC
Logistic Regression	Bag-of-Words	0.6022	**0.6451**	0.2531	0.3636	0.6191
	Empath	**0.7386**	0.4516	**0.3256**	0.3783	0.6258
	LDA	0.6704	0.6129	0.2923	**0.3958**	**0.6478**
	sLDA	0.6477	0.5161	0.2539	0.3404	0.5960
Random Forest	Bag-of-Words	0.8238	0.0322	0.5000	0.0606	0.5127
	Empath	0.8068	0.1290	0.3636	0.1904	0.5403
	LDA	**0.8409**	**0.1935**	**0.6666**	**0.3000**	**0.5864**
	sLDA	0.8352	**0.1935**	0.6000	0.2927	0.5829
AdaBoost	Bag-of-Words	0.7670	0.3225	0.3333	0.3278	0.5923
	Empath	0.7556	0.1935	0.2500	0.2181	0.5347
	LDA	0.7159	0.4838	0.3061	0.3750	0.6270
	sLDA	**0.7841**	**0.4848**	**0.4054**	**0.4418**	**0.6660**
Linear SVM	Bag-of-Words	0.4261	**0.7419**	0.1982	0.3129	0.5502
	Empath	**0.7727**	0.3871	**0.3636**	0.3750	0.6211
	LDA	0.7102	0.5484	0.3148	**0.3999**	**0.6466**
	sLDA	0.6591	0.5161	0.2622	0.3478	0.6029
RBF SVM	Bag-of-Words	0.4943	0.6129	0.1980	0.2992	0.5409
	Empath	**0.8011**	0.2903	**0.4090**	0.3396	0.6003
	LDA	0.6136	**0.7096**	0.2716	**0.3929**	**0.6512**
	sLDA	0.5909	0.6774	0.2530	0.3684	0.6249
Multi-Layer Perceptron	Bag-of-Words	0.7614	0.1935	0.2608	0.2222	0.5381
	Empath	**0.8125**	0.2258	**0.4375**	0.2978	0.5818
	LDA	0.7670	0.3870	0.3529	0.3692	0.6177
	sLDA	0.7556	**0.4838**	0.3571	**0.4109**	**0.6488**

TABLE V
EVALUATION METRICS OF USER CLASSIFICATION ON CROWD-SOURCED ANNOTATED TEST SET

Post Classifier	Feature Type	Accuracy	Recall	Precision	F1-Score	AUC
Logistic Regression	Bag-of-Words	0.5130	0.5294	0.0957	0.1622	0.5204
	Empath	**0.7356**	0.3823	**0.1397**	0.2047	0.5762
	LDA	0.5863	**0.6176**	0.1265	**0.2099**	**0.6005**
	sLDA	0.6099	0.5294	0.1192	0.1946	0.5736
Random Forest	Bag-of-Words	**0.9031**	0.0294	0.2000	0.0513	0.5089
	Empath	0.8769	0.0294	0.0666	0.0408s	0.4945
	LDA	0.8874	0.0294	0.0909	0.0444	0.5003
	sLDA	0.8848	**0.0588**	**0.1428**	**0.0833**	**0.5121**
AdaBoost	Bag-of-Words	0.7643	0.0882	**0.1764**	0.1176	0.4991
	Empath	**0.7931**	0.2058	0.1186	0.1505	0.5282
	LDA	0.6963	0.2941	0.0980	0.1470	0.5148
	sLDA	0.6989	**0.3823**	0.1214	**0.1843**	**0.5561**
Linear SVM	Bag-of-Words	0.3612	**0.6470**	0.0866	0.1527	0.4901
	Empath	**0.7801**	0.3235	**0.1527**	0.2075	0.5741
	LDA	0.6333	0.5588	0.1319	**0.2134**	**0.5998**
	sLDA	0.6884	0.4705	0.1367	0.2119	0.5902
RBF SVM	Bag-of-Words	0.5104	0.4705	0.0864	0.1461	0.4924
	Empath	**0.8219**	0.1764	**0.1304**	0.1500	0.5307
	LDA	0.5183	**0.5588**	0.1010	**0.1711**	**0.5366**
	sLDA	0.5235	0.5000	0.0934	0.1574	0.5129
Multi-Layer Perceptron	Bag-of-Words	**0.8795**	0.1177	**0.2000**	0.1481	0.5358
	Empath	0.8743	0.0882	0.1500	0.1111	0.5196
	LDA	0.8062	0.2647	0.1551	**0.1956**	**0.5619**
	sLDA	0.7565	**0.2941**	0.1265	0.1770	0.5479

TABLE VI
EVALUATION METRICS OF USER CLASSIFICATION ON EXPERT ANNOTATED TEST SET

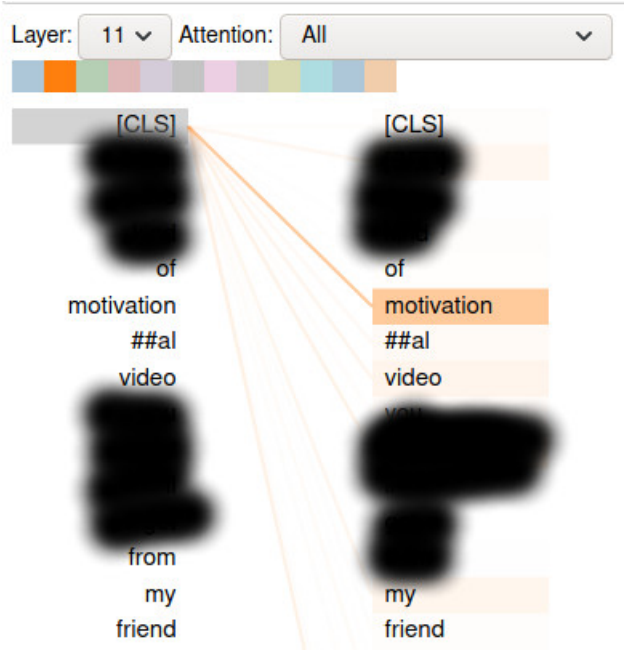

Fig. 4. Attention weights from BERT layer 11. The classifier classified this user as no risk

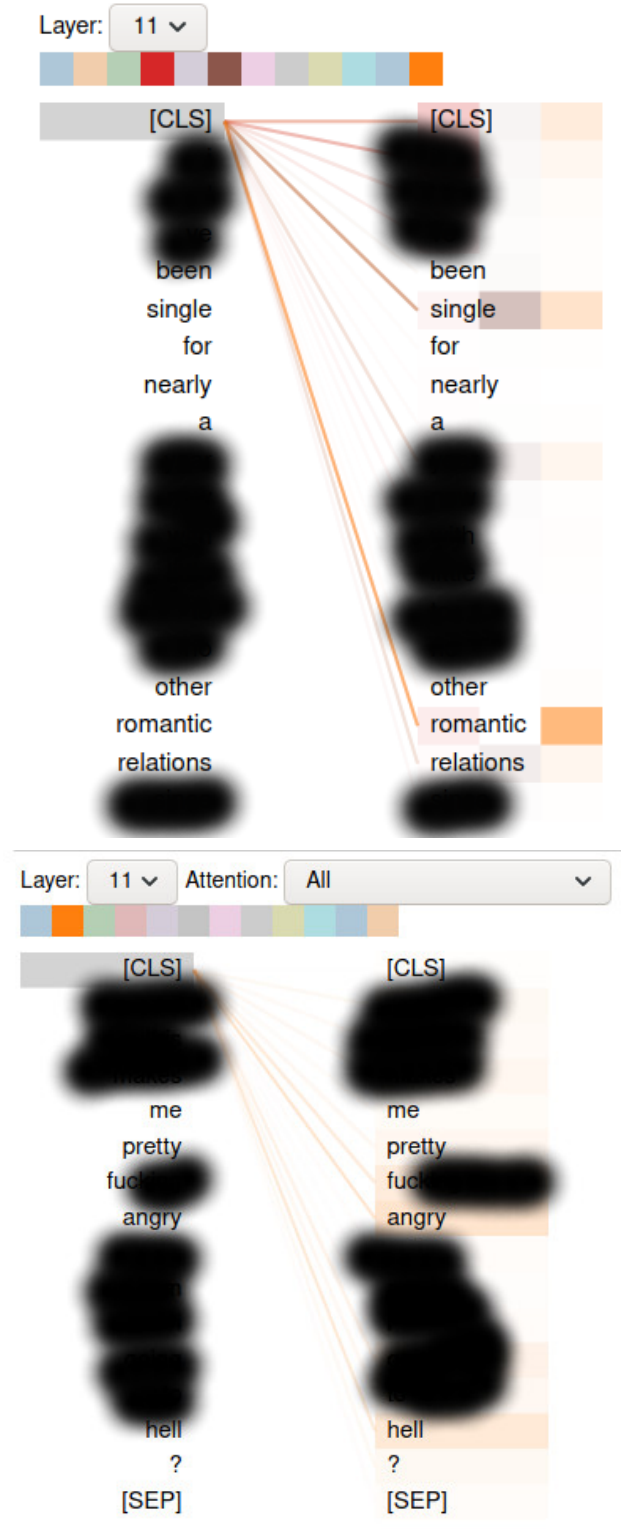

Fig. 5. Attention weights for users classified as severe risk