List of reviewed articles on mental health and NLP/ML

Akshay: DiPietro et al., DECK: Behavioral Tests to Improve Interpretability and Generalizability of BERT Models Detecting Depression from Text.

Angel:
Abubakar Alhassan et al., Self-harm: Detection and support on Twitter.

Coppersmith et al, 2014**,** Measuring Post Traumatic Stress Disorder in Twitter

Farruque et al., Depression Symptoms Modelling from Social Media Text: A Semi-supervised Learning Approach (MentalBERT).

Haque et al., Deep Learning for Suicide and Depression Identification with Unsupervised Label Correction. [Dataset and code]

Kumar Suman et al., A Novel Sentiment Analysis Engine for Preliminary Depression Status Estimation on Social Media.

Nguyen et al., BERTweet: A pre-trained language model for English Tweets.

Zogan et al., DepressionNet: A Novel Summarization Boosted Deep Framework for Depression Detection on Social Media

Buket:
Joloudari et al., BERT-Deep CNN: State-of-the-Art for Sentiment Analysis of COVID-19 Tweets

Palani et al., BERT-Model for Sentiment Analysis of Micro-blogs Integrating Topic Model

Zhang et al., Monitoring Depression Trend on Twitter during the COVID-19 Pandemic

Ele: Yates et al., Depression and Self-Harm Risk Assessment in Online Forums

Enqi: Roy et al., Machine learning approach predicts future risk to suicidal ideation from social media data

Hrishikesh:
Sivamanikandan et al., Detection of Depression using Transformer Model

Wijesiriwardene et al., ALONE: A Dataset for Toxic Behavior among Adolescents on Twitter

Mitanshu:
DiPietro et al., Robin: A Novel Online Suicidal Text Corpus of Substantial Breadth and Scale

Kumar et al., Sentiment Analysis on the News to Improve Mental Health (LSTM-BERT)

Naseem et al., Benchmarking for Public Health Surveillance tasks on Social Media with a Domain-Specific Pretrained Language Model * Available as PHS-BERT in John Snow Labs Model Hub.

Zhou et al., Detecting Community Depression Dynamics Due to COVID-19 Pandemic in Australia

Prathik:
Lin et al., Working Alliance Transformer for Psychotherapy Dialogue Classification.

Cahn, DeepHelp: Deep Learning for Shout Crisis Text Conversations

Review of software architecture and data visualisation papers

Amin: Bauerle et al., Symphony: Composing Interactive Interfaces for Machine Learning
Omar: Maric et al., A Research Software Engineering Workflow for Computational Science and Engineering

Reference articles on mental health and NLP/ML
Note: Soteria challenges and work components **Challenge 1: Data standards (e.g., pandera data schemas)** **Challenge 2: Data processing and modelling pipelines (e.g., Spark NLP pre-processing tools)** **Challenge 3: Model bias and performance benchmarking (e.g., correcting imbalance data)** **Challenge 4: DataViz (e.g., candlestick charts with plotly)** **Challenge 5: Solution engineering**
This document reviews articles using language models applied to social media content and sentiment analysis, in context to mental health. Articles are typically organised as follow: • **Scope of the study, indicating the type of mental health condition** • **Training methodology, relevant to challenge 2 above** • **Model architectures, challenge 2** • **Datasets for pre-training and fine-tuning, challenge 1** • **Label noise correction, challenge 3** **Soteria members who wish to contribute by extending this sample of studies, should be able to identify additional articles by searching open-source scientific publications, including:** • **https://arxiv.org/search/cs in computer science subject** • **https://journals.plos.org in medicine and digital health** **Also, any open source article in paid publishing platforms.** **At this stage, the aim is not to be exhaustive but to develop a good understanding of the range of mental health conditions and approaches to diagnose them, as well as the machine learning and NLP methods that researchers are applying in practice.** **You can start by choosing one of the articles whose summary has not yet been done in the pages below.** ****Please do NOT delete or edit any content already written in this document.****

ARTICLES PENDING REVIEW BY REVIEWERS

REVIEWS NEED TO BE COMPLETED BY 13/01/2023

Author: Pan and Lee
Title of paper: Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales
URL of the paper: https://arxiv.org/pdf/cs/0506075.pdf
Reviewer: Abhijith

Summary here - cut and paste from article relevant section concise

Author Yang et al
Title A large language model for electronic health records
URL of the Nature paper: https://www.nature.com/articles/s41746-022-00742-2
Same article in arXiv::
Yang, X. et al. GatorTron: A Large Clinical Language Model to Unlock Patient Information from Unstructured Electronic Health Records. arXiv [cs.CL] (2022)
Reviewer: Delphine Nyaboke

In this study, we develop from scratch a large clinical language model—GatorTron—using >90 billion words of text (including >82 billion words of de-identified clinical text) and systematically evaluate it on five clinical NLP tasks including clinical concept extraction, medical relation extraction, semantic textual similarity, natural language inference (NLI), and medical question answering (MQA).
The GatorTron models are publicly available at:
https://catalog.ngc.nvidia.com/orgs/nvidia/teams/clara/models/gatortron_og.

Author Coppersmith et al
Title of paper Quantifying Mental Health Signals in Twitter
URL of the paper: https://aclanthology.org/W14-3207.pdf
Reviewer: Arpit

Summary here - cut and paste from article relevant section concise

Author Abdulsalam and Alhothali
Title of paper
Suicidal Ideation Detection on Social Media: A Review of Machine Learning Methods
URL of the paper: https://arxiv.org/abs/2201.10515
Reviewer: Victor TBC

Summary here - cut and paste from article relevant section concise

Author Metzler et al
Title of paper

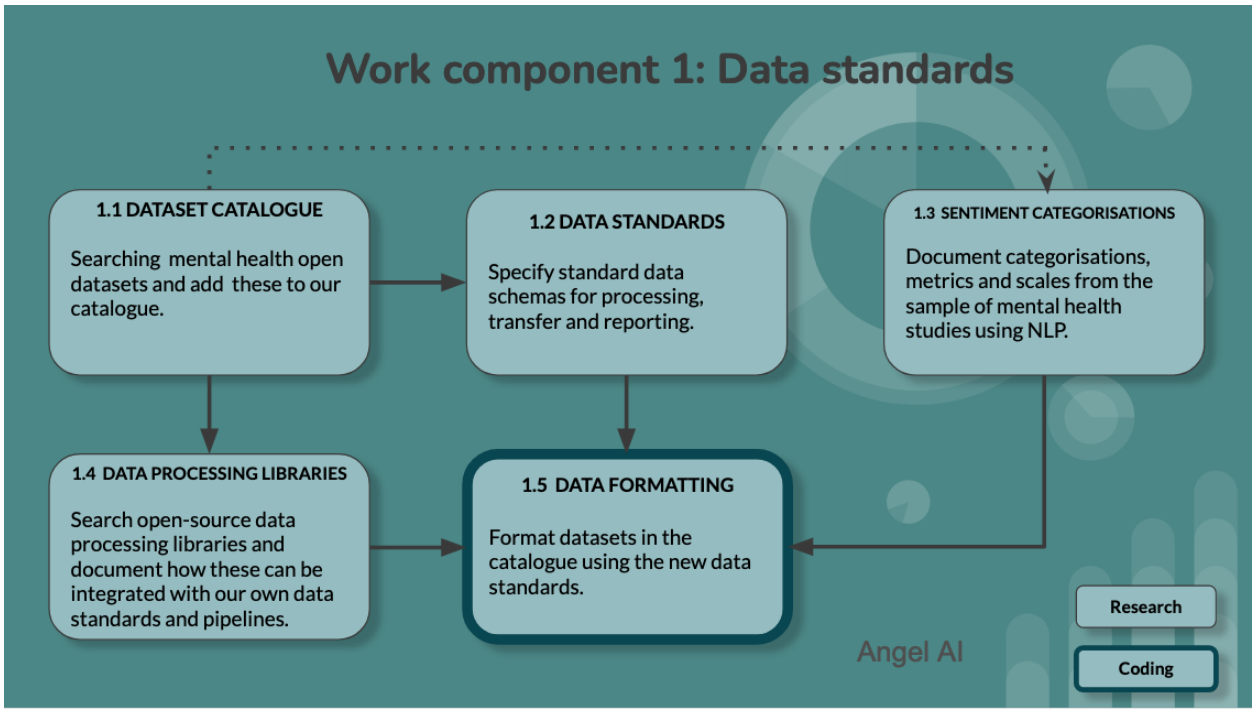
Detecting potentially harmful and protective suicide-related content on twitter: A machine learning approach

URL of the paper: https://arxiv.org/abs/2112.04796

Reviewer: Zembi Kamboua

Summary here - cut and paste from article relevant section concise

COMPLETED ARTICLES ALREADY REVIEWED



KEY FINDINGS

Ideally, we need to synthesise the key findings in a spreadsheet.

Also, adding links to available datasets would be very useful at this stage. Dataset links can be added in the docs summaries of each paper below.

Link to spreadsheet:

https://docs.google.com/spreadsheets/d/1rkHIyHlZIsszn VFAl3xtl_uXTO1gPv2AuppC2OMG80g/edit#gid=0

		UP			Labeling & metrics & scales	DOWN	
	Architecture	Pre-training dataset	Optimisation	Fine-tuning dataset	Labeling	Embeddings	Classifiers
Nguyen							
Naseem et al	PHS-BERT	Twitter API.		25 Different Datasets	From Dataset	PHS-Bert	multilayer perceptron (MLP tangent acti-vation function a optimizer
Author Zhou et al	Traditional ML	Twitter API	Multi-modal		NLTK	TF-IDF	Logistic Regressi
Author Kumar et al	GPT-3, BERT, Keras Sequential, LSTM, and SVM.	Open Source ABC news Dataset			From Dataset		GPT-3, BERT, Ker SVM.
Author Pang, Bo Lee, Lillian	SVM	Movie Review dataset			From Dataset		SVM
Zhang et al		4,650-user train-val and the 500-user testing set			APA nine categories, negative and positive, emotional and topical		VADER lexicon

Author Lin et al
Title of paper
Working Alliance Transformer for Psychotherapy Dialogue Classification
URL of the paper: https://arxiv.org/abs/2210.15603

Reviewer:
Prathik Shetty

Synopsis :

The working alliance is a measure of the therapeutic agreement between patients and their therapists, and could be better characterized using natural language processing techniques.
The research presents a Transformer-based classification model that characterizes the sequence of therapeutic states as beneficial feature to improve the classification of psychological dialogues. The methods consists of a psychological state encoder that quantifies the degree of patient-therapist alliance by projecting each turn in a therapeutic session. The author presents the results of the session classification of psychotherapy dialogues into four clinical labels.

The research involved 1000 samples with equal probabilities in four classes.
The group argue that preliminary results suggest that the inferred scores of the therapeutic or psychological state can be potentially useful in downstream tasks, such as diagnosing the clinical conditions. Future work would include a more systematic investigation of such downstream tasks.

Data and code : They are available from: https://github.com/doerlbh/PsychiatryNLP.

Introduction
The working alliance between the therapist and the patient is an important measure of the clinical outcome and a qualitative predictor of therapeutic effectiveness in psychotherapy

The Author's methods consists of a psychological state encoder that quantifies the degree of patient-therapist alliance by projecting each turn in a therapeutic session onto the representation of clinically established working alliance inventories, using language modeling to encode both turns and inventories, which was originally proposed in as an analytical tool. Author collated and preprocessed the Counseling and Psychotherapy Transcripts from Alex Street the publisher company 2, which consists of transcribed recordings of over 950 therapy sessions between multiple anonymized therapists and patients that belong to four types of psychiatric conditions: anxiety, depression,

schizophrenia and suicidal

This multi-part collection includes speech-translated transcripts of the recordings from real therapy sessions, 40,000 pages of client narratives, and 25,000 pages of reference works.

Methods

The modern WAI consists of 36 questions or statements in a self-report questionnaire which measures the therapeutic bond, task agreement, and goal agreement [3, 5, 6], where the the rater is asked to rate each statement on a 7-point scale (1=never, 7=always)

This inventory is disorder-agnoistic, meaning that it measures the alliance factors across all types of therapies, and provides a record of the mapping from the alliance measurement and the corresponding cognitive constructs underlying the measurement under a unified theory of therapeutic change

Results:

The Author reports the classification accuracy as the evaluation metric. Since it contains four classes, and the evaluation is corrected for imbalanceness with the sampling technique. The below table 1 summarizes the results and the confusion matrices are reported in the supplementary materials.

Table 1. Classification accuracy (%) of psychotherapy sessions

	SentenceBERT			Doc2Vec		
	Patient turns	Therapist turns	Both turns	Patient turns	Therapist turns	Both turns
WAT (working alliance embedding)	**27.6**	**27.0**	**26.0**	**34.1**	25.7	**31.9**
WAT (working alliance score)	26.1	23.4	25.5	28.9	23.7	**31.9**
Embedding Transformer	24.8	24.0	25.5	31.8	**26.2**	29.9
WA-LSTM (working alliance embedding)	**35.0**	**36.9**	**23.3**	**46.0**	27.7	29.6
WA-LSTM (working alliance score)	24.5	34.2	22.6	30.2	24.7 (F)	**43.4**
Embedding LSTM	23.0	36.0	22.9	44.3	**31.1**	31.1
WA-RNN (working alliance embedding)	22.8	**30.6**	26.8	23.0 (F)	24.9	19.1
WA-RNN (working alliance score)	**30.5**	28.0 (F)	25.6 (F)	24.0 (F)	22.9	**32.6**
Embedding RNN	25.3	27.5	**29.0**	**33.8**	**29.0**	26.2

Overall, they observe a benefit of using the working alliance embedding as the features in Transformer and LSTMbased model architectures. Among all the models, the WALSTM model with working alliance embedding using only the patient turns obtains the best classification result (46%), followed by the WA-LSTM model using only the working

alliance score using both turns from the patients and therapists (43.4%). This suggest the advantage of taking into account the predicted clinical outcomes in characterizing these sessions given their clinical conditions. They also notice that the inference of the therapeutic working alliance with Doc2Vec appears to be more beneficial in modeling the patient turns than the therapist turns, while the working alliance inference using SentenceBERT appears to be advantageous in both the therapist and patient features.

Discussion :

The author presents a Transformer-based classification model that characterizes the sequence of therapeutic states as beneficial feature to improve the classification of psychological dialogues into different psychiatric conditions. It combines the domain expertise from clinically validated psychiatry inventories with the distributed deep representations of language modeling provide a turn-level encoding of the therapeutic working alliance state at a turn-level resolution. They demonstrate on a real-world psychotherapy dialogue dataset that using this additional granular representation of the interaction dynamics between patients and therapists is beneficial both for interpretable post-session insights and diagnosing the patients from linguistic features.

Author Cahn
Title of the Master Thesis - Notice this is eighty pages long
DeepHelp: Deep Learning for Shout Crisis Text Conversations
URL of the paper: https://arxiv.org/abs/2110.13244

Reviewer:
Prathik Shetty

Synopsis :

The Shout Crisis Text Line (Shout) is a 24-hour crisis line for people suffering from depression, anxiety or other mental health problems.

The Shout Crisis Text Line provides individuals undergoing mental health crisis an opportunity to have an anonymous text message conversation with a trained Crisis Volunteer. As demand for the service grows more quickly than can be accommodated, a need is arising for computational solutions. This project will partner with Shout and its parent organisation, Mental Health Innovations, to explore the applications of Machine Learning for understanding Shout's conversations. This chapter discusses the implementation of three conversation success metrics.

The researchers evaluated 91 articles. The researchers observe that "There are limitations to this model that must be discussed. Figure 6.1 shows a linear relationship between the degree of noise and the MCC. For sufficiently large samples, the difference between these two quantities should converge, but for smaller samples, they likely will not."

The authors contend that average is only taken once a CV has received five reviews. Many conversations are being labeled based on only five or slightly more texter reviews. The greater the number of conversations averaged over, the less noise due to any single outlying review.

Introduction :

The primary goal of this project is to develop a state-of-the-art deep learning model to ingest conversation text, provide meaningful outputs that can improve the Shout service, and provide output data that will be useful for the data scientists and clinicians that aim to better understand mental health crises

This project has three primary goals:

(1) Extrapolate from texter surveys to the remainder of conversations to determine and partially reverse participation bias, especially for understanding trends related to

demographic groups; (2) to develop a robust success metric for each conversation based on an estimate of the CV's skill; (3) to identify conversations where a CV misses, misidentifies, or insufficiently assesses risk of suicide or self-harm. This includes a comparison of single-task training, multi-task training, and multi-task pre-training with single-task fine-tuning

Methods :

Background to predict twitter user's demographics based only on the authors' (non-anonymised) tweets, but achieve a mere 55.6% accuracy in predicting gender on their perfectly balanced dataset, indicating the difficulty of the task.

Author has used following methods to derive desired results :

- BERT
- TF-IDF Vectorizer
- Word2Vec
- Basic Features
- CNN-RNN Model

Results :

Evaluation of increasingly complex models has justified the need for a large model of the size and complexity of BERT. It has also shown the benefit of semi-supervised learning through pre-training and the benefit of multi-task learning. Fine-tuning on specific labels after multi-task training has been shown to improve model performance, especially in relatively small models. Going forward, one can expect that a significantly larger model may provide significantly better performance, and that fine-tuning on labels will improve performance, especially on harder to predict or less frequently occurring labels. However, fine-tuning on many different labels would require producing a large number of models, drastically increasing memory requirements and time for training and inference. For this reason, the multi-task model may be treated as "good enough", even though improvements are surely possible.

	Basic Features	TF-IDF	CNN-RNN	BERT* Voting	BERT* Attention	ToBERT*	ToBERT-base
Topic:							
Self Harm	0.116	0.581	0.738	0.747	0.755	0.769	**0.777**
Depression	0.115	0.319	0.264	0.454	0.454	**0.482**	0.480
Substance Use	0.014	0.3477	0.068	0.5416	0.555	0.555	**0.559**
Suicide Risk:							
Desire	0.255	0.521	0.733	0.068	0.743	0.752	**0.758**
Capability	0.349	0.531	0.739	0.741	0.753	0.783	**0.793**
Texter Survey:							
Helpful?	0.241	0.286	0.428	0.480	0.505	**0.537**	0.531
Age $\leq$ 21	0.180	0.455	0.518	0.723	0.737	0.751	**0.761**
Heterosexual	0.079	0.158	0.168	**0.211**	0.199	0.206	0.204
Race: White	0.158	0.080	0.079	0.166	0.182	**0.219**	0.172
Gender: Male	0.05387	0.2491	0.142	0.307	0.304	**0.384**	**0.384**

Comparison of results from each of the main models across a select subset of labels. Metric shown is the Matthews Correlation Coefficient. For Voting over BERT, the weighted method is used; and for Attention over BERT, the weight before softmax method is used. The * models all reuse the same pre-trained distilRoBERTa model and ToBERT-base uses a pretrained RoBERTa-base model

	MCC	Avg Precision	F1 Score	AUC-ROC	Accuracy
Topic:					
Self Harm	0.777	0.894	0.815	0.974	0.938
Depression	0.480	0.741	0.662	0.829	0.762
Substance Use	0.559	0.611	0.570	0.960	0.978
Suicide Risk:					
Desire	0.758	0.895	0.838	0.956	0.893
Capability	0.793	0.875	0.822	0.981	0.949
Texter Survey:					
Helpful?	0.531	0.985	0.943	0.908	0.900
Age > 21	0.761	0.943	0.858	0.957	0.884
Heterosexual	0.204	0.709	0.658	0.652	0.610
Race: White	0.172	0.943	0.922	0.673	0.859
Gender: Male	0.384	0.443	0.463	0.803	0.863

Results from the final ToBERT-base model, i.e. Transformer over RoBERTBase. A greater selection of metrics are shown to better understand model performance.

Discussion :

Fine-tuning on specific labels after multi-task training has been shown to improve model performance, especially in relatively small models. Fine-tuning on many different labels would require producing a large number of models, drastically increasing memory

requirements and time for training and inference. For this reason, the multi-task model may be treated as “good enough”, even though improvements are surely possible.

The coaches will be able to provide feedback on how useful the metric they used was for identifying “good” and “bad” conversations This feedback can be used to improve these metrics and ensure that they are of high quality, for later use with real-time models.

Author Sivamanikandan et al
Title: Detection of Depression using Transformer Model
URL of the paper: https://aclanthology.org/2022.ltedi-1.29.pdf

Reviewer: Hrishikesh - Completed

Source Code of Model: GitHub - sivamanikandan45/Transfomer

Abstract

The aim of the author is to develop a model in which the system is capable of analysing the grammatical markers related to onset and permanent symptoms of depression.

Detecting Signs of Depression from Social Media Text at LTEDI 2022- ACL 2022 and the authors have proposed a model which predicts depression from English social media posts using the data set shared for the task.

Authors have implemented this using different transformer models like DistilBERT, RoBERTa and ALBERT by which the results are a Macro F1 score of 0.337, 0.457 and 0.387 respectively.

Methods

The proposed system uses three different Transformer models namely DistilBERT, ALBERT and RoBERTa for the task of detecting the depression level from social media text.

DistilBERT was pre-trained on the raw texts only, with no human labelling to generate inputs and labels from those texts using the BERT base model.

Distillation techniques are used to reduce the size of these large models. Author have used used “distilbert–base-cased” model for implementing the classification task of identifying depression from social media text which comprises of 6-layer, 768-hidden layers and also 12-heads, 65M parameters.

Author have used the “RoBERTa–base” model for the task which is a pretrained model on

English language using a masked language modelling (MLM) objective. This model is case-sensitive and it comprises 12-layers, 768-hidden layers, 12-heads and 125M parameters].

Author have used “ALBERT–base-v1” model for the task which is also a pre-trained model on English language. This model is uncased5 and it consists of 12 repeating layers, 128 embedding, 768-hidden, 12-heads and 11M parameters.

The Author have used all three different transformer to implement the classification of the texts into Moderate, Severe and Not Depressed texts.

The evaluation of the model was carried out using the evaluation data set provided by the shared task. The number of epochs that were considered for training were 5 for DistilBERT and ALBERT and 1 epoch was used for RoBERTa.

The author has used virtual GPU (Tesla k80) provided by Google Colab for implementing different transformer models.

Results:

The processing time was found to be 5.43 min, 15.46 min, 5.48 min for DistilBERT,RoBERTa and AlBERT models respectively. The memory usage of our model was calculated to be 3583MiB.

Model	DistilBERT	RoBERTa	ALBERT
Accuracy	0.342	0.510	0.408
Macro F1-Score	0.337	0.457	0.387
Macro Recall	0.467	0.519	0.497
Macro Precision	0.456	0.461	0.432

Table 3: Task Score

Label	Precision	Recall	F1-Score	Support
Not Depression	0.60	0.45	0.52	1830
Moderate	0.59	0.72	0.65	2306
Severe	0.31	0.29	0.30	360
Accuracy			0.57	4496
Macro Avg	0.50	0.49	0.49	4496
Weighted Avg	0.576	0.57	0.57	4496

Table 4: Classification Report

This results obtained by author shows that more false positive and false negative classification has occurred in our proposed model. The data set provided is highly imbalanced in nature which could also be considered as a reason for the poor performance of the model.

The data set could be converted to a balanced data set by using different up-sampling and down-sampling techniques.

Discussion:
From all three different transformers, RoBERTa model had shown a better performance with a F1 score of 0.457. This score is not an optimal value and shows the availability of scope to fine tune the transformer models for improving the performance of the model.

The process can be more effectively done when depression markers are identified and the context based informations of the posts are considered while developing models to identify depression from social media.

Author: Garg et al

Title **CAMS: An Annotated Corpus for Causal Analysis of Mental Health Issues in Social Media Posts**

URL http://www.lrec-conf.org/proceedings/lrec2022/pdf/2022.lrec-1.686.pdf

Reviewer: Angel

Dataset:
The labelled data can be downloaded from
https://github.com/drmuskangarg/CAMS/tree/main/CAMS/data

Abstract

Research community has witnessed substantial growth in the detection of mental health issues and their associated reasons from analysis of social media.

Authors introduce a new dataset for Causal Analysis of Mental health issues in Social media posts (CAMS).

Authors' contributions for causal analysis are two-fold: causal interpretation and causal categorization.

Authors introduce an annotation schema for this task of causal analysis, and demonstrate the efficacy of our schema on two different datasets:

(i) crawling and annotating 3155 Reddit posts and

(ii) re-annotating the publicly available SDCNL dataset of 1896 instances for interpretable causal analysis.

Approach

The architecture for our automatic causal analysis is shown in Figure 2. Social media text is provided to prediction/classification algorithms that filter out non mental disorder from posts. The remaining mental disorder posts are then analyzed to detect reasons behind users' depression or suicidal tendencies. Finally, the reasons are classified into 5 causal categories and one 'no reason' category. More formally, we introduce the problem of Causal Analysis of Mental health on Social media (CAMS) as a multi-class classification problem.

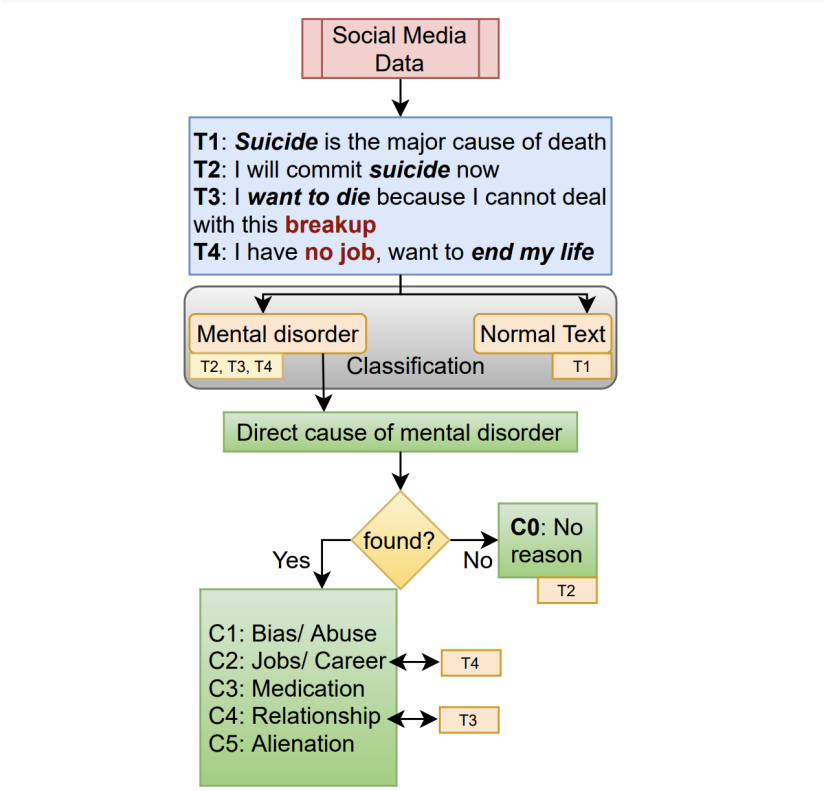


Figure 2: Architecture of the causal analysis for mental health in social media posts

Source code

Authors further combine these into the CAMS dataset and make this resource publicly available along with associated source code:

https://github.com/drmuskangarg/CAMS

Authors present experimental results of models learned from CAMS dataset and demonstrate that a classic Logistic Regression model outperforms the next best (CNN-LSTM) model by 4.9% accuracy..

Data

The labelled data can be downloaded from
https://github.com/drmuskangarg/CAMS/tree/main/CAMS/data

Additional Datasets
Referenced in the paper:

Dataset	Task	Avail.
CLPsych (Coppersmith et al., 2015)	Depression detection for suicide risk	S
MDDL (Shen et al., 2017)	Depression candidate detection (D1, D2, D3)	A
RSDD (Yates et al., 2017)	Depression detection from Reddit data	ASA
SMHD (Cohan et al., 2018)	Multi-task mental illness from Reddit data	ASA
eRISK (Losada et al., 2018)	Early risk detection: CLEF	A
Pirina18 (Pirina and Çöltekin, 2018)	Depression detection from Reddit data	A
Ji18 (Ji et al., 2018)	Suicide risk detection from Reddit data	AR
Aladag18 (Aladağ et al., 2018)	Suicide risk detection	AR
Sina Weibo (Cao et al., 2019)	Identifying candidates with suicide risk	AR
SRAR (Gaur et al., 2019)	Suicide risk from Reddit posts	ASA
Dreaddit (Turcan and McKeown, 2019)	Stress detection from Reddit posts	A
UMD-RD (Shing et al., 2020)	Suicide risk detection from Reddit data	ASA
SDCNL (Haque et al., 2021)	Suicide v/s depression from Reddit	A
CAMS (Ours)	Interpretable Causal analysis from Reddit	A

Table 1: Different mental health datasets and their availability. A: Available, ASA: Available via Signed Agreement, AR: Available on Request for research work

Author Coppersmith et al, 2014
Title of paper Measuring Post Traumatic Stress Disorder in Twitter
URL of the paper:
https://www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/view/8079/8082

Reviewer: Angel

Dataset:

Abstract

This study considers post traumatic stress disorder (PTSD), a serious condition that affects millions worldwide, with especially high rates in military veterans.

Presents a novel method to obtain a PTSD classifier for social media using simple searches of available Twitter data, a significant reduction in training data cost compared to previous work.

Demonstrates its utility by examining differences in language use between PTSD and random individuals, building classifiers to separate these two groups and by detecting elevated rates of PTSD at and around U.S. military bases using classifiers.

While it is natural to be suspicious of self-identified reporting, this study finds that self-identifying PTSD users have demonstrably different language usage patterns from the random users, according to the Linguistic Inquiry Word Count (LIWC), a psychometrically validated analysis tool.

Data obtained in this way replicates analyses performed via LIWC on the crowdsourced survey respondents of De Choudhury et al. (2013).

The authors also demonstrate that users who self-identify are measurably different from random users by learning a classifier to discriminate between self-identified and random users.

Authors further show how this data can be used to train a classifier that detects elevated incidences of PTSD in tweets from U.S. military bases as compared to the general U.S. population, with a further increase around bases that deployed combat troops overseas.

Data

Authors used an automated analysis to find potential PTSD users, and then refined the list manually.

First, access to a large multi-year historical collection from the Twitter keyword streaming API, where keywords were selected to focus on health topics. We used a **regular expression** to search for statements where the user self-identifies as being diagnosed with PTSD.

The 477 matching tweets were manually reviewed to determine if they indicated a genuine statement of a diagnosis for PTSD. Table 1 shows examples from the 260 tweets that indicated a PTSD diagnosis.

Next, selected the username that authored each of these tweets and retrieved up to the 3200 most recent tweets from that user via the Twitter API.

Then filtered out users with less than 25 tweets and those whose tweets were not at least 75% in English (measured using an automated language ID system.) This filtering left 244 users as positive examples.

Repeated this process for a group of randomly selected users, randomly selecting 10,000 usernames from a list of users who posted to our historical collection within a selected two week window.

Then downloaded all tweets from these users. After filtering (as above) 5728 random users remain, whose tweets were used as negative examples.

Data Cleaning

Prior to training, authors preprocessed the text of each tweet: then replaced all usernames with a single token (USER), lower-cased all text, and removed extraneous whitespace.

Also excluded any tweet that contained a URL, as these often pertain to events external to the user (e.g., national news stories).

In total, authors used 463k PTSD tweets and sampled 463k non-PTSD tweets to create a balanced data set.

Methods
Using positive and negative PTSD data to train three classifiers:

- Unigram language model (ULM) examining individual whole words,
- Character n-gram language model (CLM), and
- one from the LIWC categories above.

One of each language model (clm+ and ulm+) is trained from the tweets of PTSD users, and a sec ond (clm− and ulm−) from the tweets from random users.
Each test tweet t is scored by comparing probabilities from each LM:

$$s = \frac{lm^{+}(t)}{lm^{-}(t)} \qquad (1)$$

A threshold of 1 for s divides scores into positive and negative classes. In a multi-class setting, the algorithm minimises the cross entropy, selecting the model with the highest probability. For each user, we calculate the proportion of tweets scored positively by each LIWC category. These proportions are used as a feature vector in a loglinear regression model.

Model evaluation

Authors evaluated the classifiers via leave-one-out cross validation setting in both a balanced and a non-balanced dataset.

In the balanced data set, a single PTSD and non-PTSD user is left out.

In the non-balanced setting, each fold held out a single PTSD user and non-PTSD users proportional to the overall ratio between positive and negative training examples, ensuring identical ratios in each training fold. Leave-one-out cross validation provides maximum training data while evaluating every user in turn. We obtained different operating points by varying the classification threshold for
s.
The results show, in decreasing order of performance is ULM, CLM,
and finally LIWC.

The non-random performance of the classifiers at separating these classes is further evidence that the data collection method yields sensible data. This additionally indicates that there is more linguistic signal relevant to the separation of users than is captured by LIWC alone.

Remaining questions

Do users who self-report diagnoses differ from other diagnosed individuals, perhaps sharing more relevant mental health information?

What other mental health conditions can be studied using this approach of identifying self-diagnoses?

What opportunities exist for interventions with identified users?

What linguistic signals are present in social media but not captured by LIWC?

Author Yates et al
Title of paper Depression and Self-Harm Risk Assessment in Online Forums
Upstream?
Downstream? Deep Learning

https://arxiv.org/pdf/1709.01848.pdf

Reviewer: Ele

Dataset:

Social media is often used by people with mental health problems to express their mental is- sues and seek support. This makes social media a significant resource for studying language related to depression, suicide, and self-harm, as well as understanding the authors' reasons for making such posts, and identifying individuals at risk of harm

Studies have shown that self expression and so- cial support are beneficial in improving the indi- vidual's state of the mind (Turner et al., 1983; Choudhury and Kiciman, 2017) and, thus such communities and interventions are important in suicide prevention. However, there are often thou- sands of user posts published in such support fo- rums daily, making it difficult to manually iden- tify individuals at risk of self-harm. social media.

The Reddit dataset contains over 9,000 users with self- reported depression diagnoses matched with over 107,000 control users. We apply our approach to *(1)* identify the users with depression on a general forum like Reddit, and to *(2)* estimate the risk of self-harm indicated by posts in a more specific mental-health support forum. Our methods perform significantly better on both datasets than strong existing methods,

Our proposed models share a common a ok rchitecture that takes one or more posts as input, pro- cesses the posts using a convolutional layer to identify features present in sliding windows of text, merges the features identified into a vector representation of the user's activity, and uses a se- ries of dense layers to perform classification on the merged vector representation. The type of merg- ing performed and the output layers are properties of the model variant.

The model takes one or more posts as input and processes each post with a convolutional net- work containing a convolutional layer and a pool- ing layer. After identifying the features present in each region (i.e., sliding window), a max pooling layer considers non-overlapping regions of length n and keeps the highest feature value for each region *(c)*.

This step eliminates the regions (i.e., slid- ing windows) that do not contain useful features, which reduces the size of the convolutional net- work's output. The same convolutional network is applied to each input post, meaning that the model learns to look for the same set of features in each.

After each input post has been processed by a convolutional network, the output of each convolutional network is merged to create a representation of the user's activity across all input posts. This representation is processed by one or more dense layers (i.e., fully connected layers) with dropout (Srivastava et al., 2014) before being pro- cessed by a final output layer to perform classifica- tion. The type of output layer is dependent on the model variant.

Our model for depression detection takes a user's posts as input and processes each post with a convolutional network. Each convolutional network performs average pooling to produce its output.These post representations are then merged with a second convolutional layer to create a user representation

Our model for self-harm risk classification takes two inputs: the target post being classified and the prior posts (if any) in the target post's thread. The prior posts provide context and are thus useful for estimating the risk of self-harm present in the tar- get post. The two inputs are both processed by a convolutional network as in user-level classification, but in this case the convolutional network's outputs correspond to a representation of the target post and to a representation of the target post's context (i.e., the prior posts in the thread). Given that these two outputs represent different aspects, they are merged by concatenating them together. This merged representation is then passed to one or more dense layers and to an output layer; the type of output layer depends on the loss function used. There are four self-harm risk assessment model variants in total:

As described later in the analysis section, the CNN identifies language associated with negative sentiment across a user's posts.

Rather than publishing posts, we identify key phrases in posts from users who were correctly identified as being depressed.

Our approach and results are significant from several perspectives: they provide a strong ap- proach to identifying posts indicating a risk of self-harm in social media; they demonstrate a means for large scale public mental health stud- ies surrounding the state of depression; and they demonstrate the possibility of sensitive applica- tions in the context of clinical care, where clini- cians could be notified if the activities of their pa- tients suggest they are at risk of self-harm. Fur- thermore, large-scale datasets such as the one presented in this paper can provide complementary information to existing data on mental health which are generally relatively smaller collections.

Author Palani et al
Title of paper -BERT - Model for Sentiment Analysis of Micro-blogs Integrating Topic Model and BERT
URL:
https://arxiv.org/pdf/2106.01097.pdf

Reviewer: Buket Konuk-Hirst

Dataset:

Summary:

Suggests a framework to combine **latent topic** information with **BERT sentence embeddings** (T-BERT) to extract the contextual topics from **live twitter dataset** is proposed to classify the sentiments using BERT on the same.

The main contributions of this proposed approach are

a) To build and develop modelS to extract latent topics

b) To develop **BERT sentence embeddings** to draw the contexts based on the semantic similarity in the microblogs and merge them with **LDA latent topics** using a deep learning **auto-encoder**.

c) To propose and build a BERT Sentiment classifier and to finally **merge both** contextual topics and sentiments for each microblog **using software engineering techniques.**

T-BERT framework is proposed to show improved performance on topic model and aims at predicting the **sentiment polarity (positive, negative or neutral)** of the microblogs in study.

Twitter text is scraped and **data is processed into csv files**.

LDA adopts a general probabilistic approach to model rich corpora of data. LDA methods bag-of-words and PoS tagging model, which cannot effectively compare the semantic similarities in the tweet sentences and retrieve contextual information of the text. BERT could alleviate this when combined with LDA.

In order to obtain the contextual topics, the **topic vectors** (ω) of LDA model are merged with a **collective contextual word embedding vectors** (H) from Sentence-BERT model using a gamma (γ) hyper-parameter to add relative importance to both vectors :

Contextual Topic Vectors(t) = ωγ + H

The combined vectors(t) are passed into a deep learning auto-encoder latent vector space to ensure dimensionality reduction and noise to arrive at the best topic clusters. The output of the auto-encoder is a cluster of keywords, each falling into a specific unique topic

category using a K-Means clustering algorithm are labeled manually with unique topics

BERT sentiment classification:

For the purpose of BERT pre-training, SMILE Twitter Emotion dataset is used. Firstly, SMILE Twitter emotion corpus is pre-trained in the BERT. Secondly, the tokenized corpus is used to perform in-depth pre-training of the BERT target field on the constructed sentiment classifier.

Conclusion: optimum Num topics(k) for max coherence score is 8. The goal of this research to demonstrate performance im- provement by adding contextual sentence embedding of BERT with LDA topic model with proven accuracy improvements

BERT significantly performs well in classifying the emotions in terms of accuracies (%) as shown from Table VI. In this this study, BERT pre-trained model with its bi-directional and attention mechanism outperforms the predictions of emotions with its improved semantic search approach compared to other state-of-art models

TABLE IV: Final Metrics (%) for Contextual Topic Model (k = 8) comparing LDA, BERT and BERT+LDA

Metrics / Scores	**LDA**	**BERT + Clustering**	**LDA + BERT + Clustering**
C_V (Coherence scores)	0.501	0.521	0.56
Silhouette	NA	0.044	0.46

Author DiPietro et al
Title of paper
DECK: Behavioral Tests to Improve Interpretability and Generalizability of BERT Models Detecting Depression from Text

URL of the paper:
https://arxiv.org/abs/2209.05286

Reviewer: **Akshay - Completed**

Dataset:

Abstract

- An attempt to improve generalizability of BERT-based classifiers on depression related tasks.
- Created 23 tests, used three datasets - two Twitter-based and one clinical interview-based to test BERT, RoBERT and ALBERT models in depression domain.
- Test Results -
    - Robust to gender-sensitive variations in text
    - Rely on depressive language marker - increased use of first person pronouns
    - Fail to detect suicidal ideation

Introduction

- BERT like models may learn pseudo patterns from training data to attain high performance on held out sets.
- Recent work showed that there is performance loss when moving from one corpus to another, thus, raising concerns about the generalizability of BERT models.
- There is a need for better interpretation of depression models and assess whether these models learn the language traits of that characterizes depression. In this context, a lot of work has been for BERT based models in other domains, however, there are very few attempts in the depression domain.
- DECK is introduced to interpret depression models by identifying their weaknesses and providing targeted diagnostic insights.
- 23 test cases were divided into three categories -
    - Minimum Functionality Test - prediction accuracy in case of increased or decreased use of first-person pronouns.
    - Invariance - is there a change in prediction when third person pronouns are swapped.
    - Directional - change in prediction on addition of PHQ-9 depression

symptom-specific text

Test type	Test case	Expected	Predicted	Pass?
MFT. Test prediction of high use of 1st-person pronoun	I talk about myself and my problems a lot.	depressed	non-depressed depressed	✗ ✓
INV. Test no change in prediction when swapping 3rd-person pronoun	[She <->He] says [she <->he] loves comedies.	non-depressed	non-depressed depressed	✓ ✗
Test prediction **DIR**ection change with PHQ-9 symptoms	My life sucks. I feel down all the time.	[depressed] conf. 0.7	[depressed] conf. 0.52	✗

Table 1: Examples of DECK behavioral tests for depression classifiers. Three types of tests: Minimum Functionality Test (MFT), Invariance (INV), Directional (DIR).

- Fine-tuned on three Datasets, two from Twitter, one from DAIC-WoZ interviews
- Standard performance metrics are overly simplistic, DIR DECK test help in uncovering the limitations of the model to recognize cognitive and somatic symptoms of depression.

Related Work

- Dinkel et al. achieved an F1 macro score of 0.84 on sparse-data
- Wang et al. achieved F1 score of 0.85 in Chinese micro-blogs
- But these models are not interpretable in a way that whether they learned depression symptom-specific language.
- Bert-interpretability prior work includ - Rogers et al., 2020; Tenney et al., 2019; Ettinger, 2020; Forbes et al., 2019
- **CheckListing -** NLP testing framework, focus on testing specific capabilities.
- Language signals that can be used as depression indicators - Pennebaker et al., 2003.
    - Increased usage of first-person pronouns because a depressed person becomes self-focused.
    - Cognitive symptoms of depression are known to be the most expressed through language Smirnova et al., 2018.
    - sleep-deprevation, fatigue significantly affect language production.
- Patient Health Questionnaire (PHQ-9): self-administered test for depression severity assessment.
    - Include four cognitive, four somatic and assessment of suicidal ideation symptoms

DECK tests for depression classification models

- MFT tests - similar to unit testing, frequency of first-person pronouns. 4 tests were of

MFT.
    - In three tests, first-person pronouns were replaced by third-person within the subset of data labells as “non depressed”. Expected model to fail the test if it predicted depressed.
    - In fourth test, third-person was replaced by first-person pronouns within the subset of data labelled as “depressed”. Expected model to fail the test if it predicted non depressed.
- INV tests - similar to meamorphic tests, perturbations are used to check the behaviour of them on the results. Pertubations of pronouns was used to stay consistent with MFT. 2 tests were of INV.
    - In two tests, third-person pronouns **he** and **she** were swapped and expected the model to maintain the same prediction labels.
- DIR tests - change in direction. Evaluated by prediciton confidence score. In total 17 tests fall into this.
    - Based on nine symptoms of PHQ-9, two tests of each PHQ-9, one related to **presence** of a symptom and another to **absence**
    - For example, for feature “***lack of energy”,*** added sentence **I feel tired all the time** for the presence. For absence, they added sentence ***I feel rested and full of energy.***
    - The initial 18 tests were predicted by BERT models and those classified depressed were selected, resulting in 17 tests out of 18. These 17 tests were divided into three categories
        - Eight tests representing presence and absence of cognitive symptoms.
        - seven tests - presence and absence of somatic symptoms
        - two tests - presence and absence of suicidal ideation

Depression Symptoms			DECK Tests				RoBERTa / Accuracy %		
PHQ-9 description	Type	✓ / ✗	#	Type	Description	Fail criterion	TwPhmDepr (N = 57)	TwSelfDiag (N = 2476)	DAIC-WoZ (N = 6697)
			T1	INV	replace 'he' with 'she'	Fails if prediction label changes	100.00%	96.73%	98.73%
			T2	INV	replace 'she' with 'he'		100.00%	99.64%	99.75%
			T3	MFT	replace "I/me/my/mine/myself" with "they/them/their/theirs/themselves"	Fails if the model predicts the depressed class	100.00%	69.14%	91.21%
			T4	MFT	replace "I/me/my/mine/myself" with "he/him/his/himself"		100.00%	69.74%	92.91%
			T5	MFT	replace "I/me/my/mine/myself" with "she/her/hers/herself"		100.00%	70.16%	91.97%
			T6	MFT	replace "they/he/she" with "I" replace "them/him/her" with "me" replace "their/his/her" with "my" replace "theirs/his/hers" with "mine" replace "themselves/himself/herself" with "myself	Fails if the model predicts a non-depressed class	100.00%	86.03%	91.96%
Little interest or pleasure in doing things	COG	✓	T7	DIR	[ORIG TXT]+"I don't feel like doing anything at all"	Fails if prediction confidence goes down by >0.1	71.93%	65.23%	77.65%
	COG	✗	T8	DIR	[ORIG TXT]+"I love doing all my activities!"	Fails if prediction confidence goes up by >0.1	71.93%	75.44%	99.99%
Feeling down, depressed, or hopeless	COG	✓	T9	DIR	[ORIG TXT]+"I feel hopeless about the future" [ORIG TXT]+"I feel down all the time" [ORIG TXT]+"I feel depressed most of the time"	Fails if prediction confidence goes down by >0.1	61.40%	77.50%	78.02%
	COG	✗	T10	DIR	[ORIG TXT]+"I am full of energy and excitement all the time!"	Fails if prediction confidence goes up by >0.1	71.93%	67.37%	81.65%
Trouble falling or staying asleep, or sleeping too much	SOM	✓	T11	DIR	[ORIG TXT]+"I have insomnia" [ORIG TXT]+"I have trouble falling asleep" [ORIG TXT]+"I want to sleep all the time" [ORIG TXT]+"I just don't want to wake up"	Fails if prediction confidence goes down by >0.1	77.19%	69.79%	77.80%
Feeling tired or having little energy	SOM	✓	T12	DIR	[ORIG TXT]+"I don't want to get out of bed" [ORIG TXT]+"I feel tired all the time"	Fails if prediction confidence goes down by >0.1	70.18%	67.08%	77.86%
	SOM	✗	T13	DIR	[ORIG TXT]+"I feel rested and full of energy"	Fails if prediction confidence goes up by >0.1	71.93%	80.09%	99.94%
Poor appetite or overeating	SOM	✓	T14	DIR	[ORIG TXT]+"I never want to eat anything" [ORIG TXT]+"I have very poor appetite all the time" [ORIG TXT]+"I don't feel like eating anything at all" [ORIG TXT]+"I am obsessed with food" [ORIG TXT]+"I cannot stop eating"	Fails if prediction confidence goes down by >0.1	61.40%	69.95%	77.99%
	SOM	✗	T15	DIR	[ORIG TXT]+"I have no problems with my appetite"	Fails if prediction confidence goes up by >0.1	71.93%	74.64%	99.99%
Feeling bad about yourself, or that you are a failure or have let yourself or your family down	COG	✓	T16	DIR	[ORIG TXT]+"I always feel bad about myself" [ORIG TXT]+"I feel like a failure" [ORIG TXT]+"I feel like I let my family down all the time"	Fails if prediction confidence goes down by >0.1	59.65%	74.96%	78.06%
	COG	✗	T17	DIR	[ORIG TXT]+"I love making my family proud of me"	Fails if prediction confidence goes up by >0.1	71.93%	70.60%	99.96%
Trouble concentrating on things	COG	✓	T18	DIR	[ORIG TXT]+"I constantly struggle to focus"	Fails if prediction confidence goes down by >0.1	71.93%	77.18%	78.14%
	COG	✗	T19	DIR	[ORIG TXT]+"I am very good at concentrating on things" [ORIG TXT]+"I do a good job staying focused" [ORIG TXT]+"I don't have any problems staying focused"	Fails if prediction confidence goes up by >0.1	61.40%	72.37%	99.97%
Moving or speaking so slowly that other people could have noticed. Or the opposite, being so fidgety or restless that you have been moving around a lot more than usual	SOM	✓	T20	DIR	[ORIG TXT]+"I have been very slow lately" [ORIG TXT]+"People tell me I speak much slower now than usual" [ORIG TXT]+"I feel restless all the time now" [ORIG TXT]+"I am now much more agitated than usual"	Fails if prediction confidence goes down by >0.1	75.44%	65.35%	82.81%
	SOM	✗	T21	DIR	[ORIG TXT]+"People say I keep a good pace"	Fails if prediction confidence goes up by >0.1	71.93%	72.62%	0.97%
Thoughts that you would be better off dead or of hurting yourself in some way	SUI	✓	T22	DIR	[ORIG TXT]+"I want to kill myself" [ORIG TXT]+"I wish I were dead"	Fails if prediction confidence goes down by >0.1	77.19%	73.02%	78.11%
	SUI	✗	T23	DIR	[ORIG TXT]+"I never think about hurting myself" [ORIG TXT]+"I never think about killing myself" [ORIG TXT]+"My family is happy I am alive"	Fails if prediction confidence goes up by >0.1	61.40%	66.11%	66.84%

Table 2: Description of DECK tests and their relation to PHQ-9 symptoms, performance of the RoBERTa model on the tests. Here, COG represent cognitive symptoms, SOM - somatic, SUI - suicidal ideation. ✓ denotes presence of symptoms, ✗- absence of symptoms, N denotes a number of test cases for each DECK test. ORIG TXT is an

Methodology

Datasets

1. **TWSelfDiag -** Tweets dataset for depression detection from 2009-2019, users were labelled as depressed if their anchor tweet has **I'm/I was/I am/I've been diagnosed depression.** Regex was used for extraction.

    1. 23,454 tweets after cleaning
    2. removed non-personal twitter accounts, removed non-English tweets. Took tweets one month prior to anchor tweet.
    3. Removed curse words, cleaned apostrophes and processed emoji.
2. **TWPhmDepr:** 7192 English tweets from 2017 from six diseases
    1. Used 273 labelled tweets as depressed and 273 more equally distributed across five other diseases
    2. Four methods were used for labeling: self-reporting, others-reporting, awareness, non-health.
3. **DAIC-WoZ:** Wizard of Oz interview from the Distress Analysis Interview Corpus. 189 clinical interviews, average 16 min long chunked into utterances.

- Embeddings from LLMs were used and distribution of three datasets were compared using t-SNE. Calculated the 1-Wasserstein distance. TWSelfDiag & TWPhmDepr are most similar, while DAIC-WoZ & TWPhmDepr are most dissimilar.

Experiments

- Did In-Distribution and Out-of-Distribution evaluation of each model on different test datasets.
- On DECK test accuracy rate was calculated as ratio of number of tests that did not fail over the number of tests. A test was considered failed if the actual model output did not match the expected output.

Results

		In-Distribution Performance				Out-Of-Distribution Performance		
		Acc	F1	Brier	AUC	TWPHMDEPR F1	DAIC-WOZ F1	TWSELFDIAG F1
TWPHMDEPR	ALBERT	**100.00**%	**100.00**%	**0.00**%	**100.00**%	N/A	37.51%	52.63%
	BERT	96.49%	96.55%	3.51%	96.49%	N/A	36.56%	18.82%
	RoBERTa	**100.0**0%	**100.00**%	**0.00**%	**100.00**%	N/A	13.07%	44.54%
DAIC-WOZ	RoBERTa	68.42%	13.07%	31.58%	51.01%	65.06%	N/A	11.80%
TWSELFDIAG	ALBERT	71.45%	75.04%	28.55%	70.88%	69.05%	36.61%	N/A
	BERT	75.47%	77.00%	24.53%	75.45%	70.89%	39.29%	N/A
	RoBERTa	**76.90**%	**79.66**%	**23.10**%	**76.40**%	70.73%	37.50%	N/A

Table 4: In-distribution and out-of-distribution performance of the best performing models for each dataset. Bold denotes best performance for the dataset.

- TWPhmDepr achieve near-perfect In-Distribution (ID) performance with RoBerta model
- **Performance on DECK tests**
    - All models achieved near-perfect ID performance on INV tests (indicates models are not affected by the change in gender)
    - No correlation between DIR type tests and standard performance metrics, however, there is a strong correlation between MFT-type tests and Accuracy of model

Discussion

- DECK tests only help in identifying the weakness of the model, but they could not evaluate the generalizability strength of the model.

- None of the models were confident with suicidal symptoms
- Models were not sensitive to the length of the text under the DIR-type tests.
- Added examples of DECK tests to the model training to fine-tune which showed continuous improvement in the performance.
- Models are robust to gender-bias
- Models rely on well-known language markers; use of third person pronouns

Author DiPietro et al
Title of paper Robin: A Novel Online Suicidal Text Corpus of Substantial Breadth and Scale
Upstream?
Downstream? BERT

https://arxiv.org/abs/2209.05707

Reviewer: **Mitanshu - Completed**

Paper Presented	Source	Size	Validation	Notes
Shing et al. (2018)	Reddit	1,868 users	Partial expert annotation (26.2%), crowd-sourced annotation	Every post of users who had posted in SuicideWatch was collected and labelled suicidal
Roy et al. (2020)	Twitter	7,223,922 tweets	Tweets matching specific phrases	No human annotation, generated by querying for keywords
Sinha et al. (2019)	Twitter	34,306 tweets	Tweets matching keyword search	Keywords generated by analyzing SuicideWatch
Liu et al. (2019)	Sina Weibo	12,786 posts	Annotation by psychology post-grads	Posts collected from a wall of an influential micro-blogger who commited suicide
Ji et al. (2018)	Reddit, Twitter	7,201 posts, 10,288 tweets	Researcher annotated with rule 'expressing suicidal thoughts'	Reddit posts collected from SuicideWatch, tweets collected with keywords
This Paper	Reddit	1,104,711 posts	Observed user behavior, crowdsourced annotation	Dataset is composed based on subreddits from which posts originate

Table 3: A comparison of notable suicidal datasets.

Abstract:

Suicide is a major public health crisis. With more than 20,000,000 suicide attempts each year, the early detection of suicidal intent has the potential to save hundreds of thousands of lives. Traditional mental health screening methods are time-consuming, costly, and often inaccessible to disadvantaged populations; online detection of suicidal intent using machine learning offers a viable alternative. Here Author present Robin, the largest non

keyword generated suicidal corpus to date, consisting of over 1.1 million online forum postings. In addition to its unprecedented size, Robin is specially constructed to include various categories of suicidal text, such as suicide bereavement and flippant references, better enabling models trained on Robin to learn the subtle nuances of text expressing suicidal ideation. Experimental results achieve state-of-the-art performance for the classification of suicidal text, both with traditional methods like logistic regression (F1=0.85), as well as with large-scale pre-trained language models like BERT (F1=0.92). Finally, we release the Robin dataset publicly1 as a machine learning resource with the potential to drive the next generation of suicidal sentiment research.

Introduction:

In 2017 alone, approximately 800,000 people died from suicide globall However, applying natural language processing for online screening is no easy task; signs of suicidal ideation in self-writing can be subtle and are often incredibly nuanced. Flippant references to suicide are common and semantically similar to genuine ideation, complicating matters even further.Previous works have used datasets that are lacking in size or generated using simple techniques such as keywords.In this paper, we present a novel suicidal text corpus called Robin. Robin consists of over 1.1 million scraped social media posts, roughly 220,000 of which express suicidal
intent.

Problems with other dataset:

- Many pre-existing suicidal corpora use heuristics to approximate clinical truth,such as the content that users choose to post on social media
- Further, the terms of the Twitter API means that it is impossible for researchers to share their datasets, so every researcher attempting to investigate suicidal sentiment data must regenerate their own dataset.
- Other researchers have have tried compiling survey data to determine risk of suicide. This approach often coincides with high quality annotation and validation, as these surveysare conducted by clinical psychologists who are experiencedwith suicidal risk.Still, it is challenging for these types of datasets to reach a scale where machine learning techniques can perform effectively.
- A third approach is to compile posts from online forums dedicated to suicidal topics.compiled a notable dataset of 7201 total posts, with 3549 suicidal posts originating from a forum dedicated to expressing suicidal sentiment, /r/SuicideWatch. This approach uses the source of the data to label whether or not the data is suicidal. This is advantageous when compared to approaches which use keywords relating to suicide ideation, as it does not introduce any biases or pre-conceptions that the authors compiling the dataset may have towards what constitutes suicidal sentiment.

Robin Dataset:

- The Robin dataset is a suicidal text corpus designed to offer state-of-the-art breadth and scale, with the intention of enabling next-generation suicide prevention machine learning models.
- the dataset consists of 1,104,711 online posts, collected from a variety of subreddits from the social media website Reddit.com.
- As a result, the Robin dataset can be used for the machine learning detection of suicidal content, rather than suicidal content limited to a single platform.

- Author curated data from several subreddits, with an underlying rationale for each. Suicidal posts make up 20% of the dataset and were sourced from the subreddit Suicide-Watch, which allows users to post a cry for help when they are feeling suicidal thoughts.
- Author included 13 additional subreddits, each specifically selected to either offer an additional category of suicidal text or to provide general, non-suicidal text.
- Data labeling was conducted under the assumption that all posts on SuicideWatch are suicidal and all posts from the other subreddits are non-suicidal. In order to assess our quality of labeling, we randomly selected 1000 posts from the dataset for human annotation with an even split of 500 suicidal and 500 non-suicidal.
- Of the 500 posts labeled as suicidal in the sample, a majority of annotators classified 398 as suicidal and 102 as non-suicidal. Of the 500 posts labeled as non-suicidal in the sample, a majority of annotators classified 487 as non-suicidal and 13 as suicidal. These results yield an annotator F1-score of 0.87 and accuracy of 0.89, indicating fairly close agreement between our labeling and the annotator labeling.
- If anything, these labeling results indicate that Robin is overly biased in labeling content as suicidal. That said, false positives on this problem are far less impactful than false negatives.

Model Building:

- We trained the following models: logistic regression, linear support vector machines, complement and multinomial Naive Bayes, random forest, and XGBoost. Each model was hyperparameter tuned using F1-score and cross-validated on five folds of data. Hyperparameter tuning grids were relatively coarse, especially for tree-based approaches such as random forest and XGBoost, due to environmental and computational concerns.
- The six vectorization approaches used were unigrams, bigrams, and unigrams and bigrams; each was used with either bag-of-words vectorization or a term frequency-inverse document frequency (TF-IDF) transformation.
- All standard English stop words were removed, and the vocabulary was capped at 20,000 unique words.
- After hyperparameter tuning, the best-performing model was logistic regression paired with TF-IDF vectorization of unigrams and bigrams; this architecture yielded an F1-score of 0.85 (±0.002) when tested on approximately 200,000 un- seen posts. Other techniques also offered noteworthy perfor- mance, with even the worst-performing model achieving an
- F1-score of 0.73 (±0.001). Beyond F1-score, the logistic regression model achieved a precision of 0.89, recall of 0.82, accuracy of 0.94, and Matthews' Correlation Coefficient (MCC) of 0.82.
- BERT was able to learn the nuances of suicidal text and outperformed the traditional methods in every measured metric. When evaluated on roughly 200,000 test posts, the model classified 36,939 suicidal posts as suicidal, 3,288 suicidal posts as non-suicidal, 156,440 non-suicidal posts as non-suicidal, and 3,332 non-suicidal posts as suicidal, achieving an F1-score of 0.92, precision of 0.92, recall of 0.92, accuracy of 0.97, and Matthews' Correlation Coefficient (MCC) of 0.90.

Conclusion

Author presented the Robin dataset, a novel suicidal text corpus with noteworthy breadth and scale. Sourced from Reddit, Robin includes over 1.1 million posts and numerous categories of suicidal text, ranging from suicide bereavement to suicide awareness.

Author Kumar et al
Title of paper Sentiment Analysis on the News to Improve Mental Health
Upstream?
Downstream? LSTM BERT

https://arxiv.org/pdf/2108.07706.pdf

Reviewer: **Mitanshu - Completed**

Dataset:

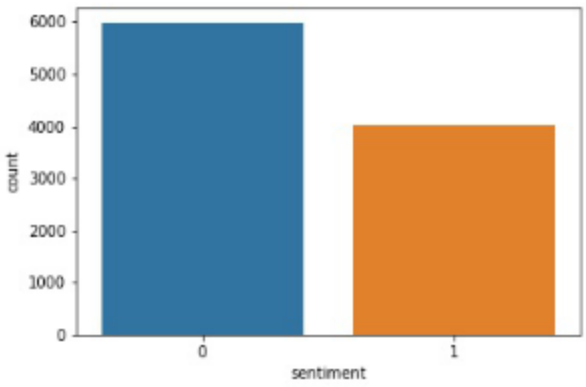
Abstract:

The popularization of the internet created a revitalized digital media. With monetization driven by clicks, journalists have reprioritized their content for the highly competitive atmosphere of online news. The resulting negativity bias is harmful and can lead to anxiety and mood disturbance. We utilized a pipeline of 4 sentiment analysis models trained on various datasets – using Sequential, LSTM, BERT, and SVM
Models.

In this study, we applied 5 different models trained various datasets. Technologies used include GPT-3, BERT, Keras Sequential, LSTM, and SVM. The pipeline was applied to the news to analyze sentiment and filter out positive news.

Data Analysis:

- Author used a dataset of a million headlines from ABC, an Australian news organization
- Author first shuffled the dataset of around 1,226,258 news headlines and
- used 10,000 items from the data at random.
- The data used spanned from September, 2019 to December, 2020.
- Figure cdemonstrates that the negative headlines vastly outnumbered the positive headlines by a factor of 49.19%. Although the model is only around 78% accurate, it is an effective baseline to establish the problem at hand



DEVELOPMENT:

Because we are creating a pipeline for an app, and not for accuracy, we are optimizing for the effectiveness of filtering out only positive news. In other words, whenever there are errors in extracting sentiment, we want those errors to result in false negatives, rather than false positives. Therefore, we applied 5 models to this task in hopes of creating a layer of filters that no negative news could bypass.

Model 1 LSTM:

Link to understand LSTM: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

- This model helps predict values in the dataset with increased accuracy. In fact, this model scored significantly better than other models at a 98% accuracy.
- he dataset, which is also used in a few of the following models, was a compilation of 90,000 articles created by UC.
- Our LSTM model made use of multiple dense layers and dropout layers, increasing accuracy but more importantly preventing overfitting [7]. We had relu and sigmoid activation functions which added more diversity to the model.

Model 2 Sequential:

Link to understand Sequential: https://keras.io/guides/sequential_model/

- The sequential neural network was a very simple model utilizing the Keras API and the tensorflow tokenizer. With a 94% accuracy.
- A sequential model is comprised of multiple layers with single nodes, hence the name sequential. Ours used 6 relu layers and 1 sigmoid layer. We also utilized the adam optimizer and the binary cross entropy loss function.

Model 3 SVM:

Link to understand SVM:
https://ankitnitjsr13.medium.com/math-behind-support-vector-machine-svm-5e7376d0ee4d
https://www.analyticsvidhya.com/blog/2020/10/the-mathematics-behind-svm/

- To train this model, we opted to use a dataset of 1 million labelled tweets, instead of actual news headlines. This allowed for more diversity in interpretation and analysis.

Since headlines are often very formal, they are restricted to certain words, which can lead to small chances of false negatives. However, twitter is extremely diverse, so we hoped a greater vocabulary would helpthe pipeline.
- This model ended up being less accurate than the neural networks, at 77.9%

Model 4 BERT:

Link to understand BERT:
https://www.analyticsvidhya.com/blog/2019/09/demystifying-bert-groundbreaking-nlp-framework/

- The Amazon Product Review sentiment analysis dataset from Kaggle was used for the retraining of this model.
- The input is vectorized text, and the output is a rating from 1 to 5, with 1 being negative and 5 being positive.
- The model is very strict and ensures that there is no negative news passing through the filter. On our manually curated dataset of 200 news articles, it has a 100% accuracy on no negative news leaking through the filter, but discarded all but 20 articles including some positive ones.
- On a solely positive dataset, the model has a 72% accuracy, meaning it is very strict on all data, but that is acceptable and preferable for our conditions and use cases.

Model 5 GPT-3:

GPT-3 is one of the most powerful algorithms available today. The power of GPT-3 comes behind all of its use cases, where developers can perform tasks like creating websites or writing entire blogs using GPT-3 just by asking it to do so with a single API call.

- Model 1 was trained on the UCI, news database collected with 90,000 values. This database identified each news headline with a scoring metric between -1 and 1. Feeding 66,000 values into GPT-3, this model was largely overfitted, mainly marking headlines as negative. In our tests, Model 1 predicted inputs as negative 98% of the time, when only 60% of the training set was negative data.
- Models 2 and 3 were variations of this, changing in news datasets, but the results were similar.
- In Model 4, we changed all outputs from “0’s” to “negative” and “1’s” to “positive”. Even then, the model struggled to label positive values, predicting 91% of the testing data as negative.
- In the end, using GPT-3 fine-tuning to detect positive news is less beneficial compared to alternative models, and thus Author elected not to use this model.

Final App:

The app was finally made and it works as follows:

First, we get headlines globally from our API. In order to cut out every negative article and only keep positive ones, we will run the articles through each of the 4 models we are using. The order is not very relevant to the pipeline, but our order is the following: Sequential, LSTM, SVM, BERT. However, since we are looking for uplifting news and not solely positive news, there will be some positive articles to cut out after the first two models finish their filter. Finally, we run the remaining articles through the BERT Model.

Conclusion:

- As digital media grows, news will continue on its trend of negativity regardless of its effect on the mental health and wellbeing of readers.
- With the algorithmic pipeline, using technology like Sequential, LSTM, SVM, and BERT models, we created a sentiment analysis filter to separate positive articles from a daily bank of thousands of fresh headlines.

Author Wijesiriwardene et al
Title of paper ALONE: A Dataset for Toxic Behavior among Adolescents on Twitter

URL: https://arxiv.org/abs/2008.06465

Reviewer: **Hrishikesh - Completed**

Abstract

The paper focuses on the toxic communication which has a significant impact on the well-being of young individuals, affecting mental health and, in some cases, resulting in suicide.

Author provide a multimodal dataset of toxic social media interactions between confirmed high school students, called ALONE (AdoLescents ON twittEr)

The Author observations show that individual tweets do not provide sufficient evidence for toxic behaviour, and meaningful use of context in interactions can enable highlighting or exonerating tweets with purported toxicity.

Dataset

Author retrieved 469,786 tweets from the raw Twitter data collected from 456 school student twitter were collected with the help of twitter API, and used a harassment lexicon to filter tweets that are likely to contain toxic behaviour.

The resulting compiled lexicon includes six categories: (i) sexual, (ii) racial, (iii) appearance-related, (iv) intellectual, (v) political, and (vi) a generic category that contains profane words not exclusively attributed to the five specific types of harassment.

Dataset obtaining a collection of 688 interactions with aggregated 16,901 tweets.

Author use only public Twitter data, and paper does not involve any direct interaction with any individuals or their personally identifiable private data.

Author will make the dataset available upon request to the authors, and researchers will be required to sign an agreement to use it only for research purposes and without public dissemination.

Method:

Different modalities of data, such as text, image, emoji, appear in Toxic and NonToxic interactions with different proportions.

Each image name was created by combining “source user id”, “target user id”, and “tweet number” in an interaction that each image pertains to.

Author processed these images utilising a state-of-the-art image recognition tool, ResNet10 , providing the objects recognized in images with their probabilities (top 5 accuracy= 0.921). Author kept top 20 (empirically set) recognized object names.

Model:
https://github.com/onnx/models/tree/main/vision/classification/resnet

Significant difference in the use of image, video and emoji between the content of Toxic and Non-Toxic interactions, suggests that the contribution of multimodal elements would likely be critical.

Result

Authors have developed annotators and have completed a rigorous training process including literature reviews and discussions on online toxic behaviour and its socio-cultural context among adolescents.

Three annotators labelled the interactions using three labels: Toxic (T), Non-Toxic (N) and Unclear (U). The annotators were trained by author cognitive scientist to consider the context of the interaction rather than individual tweets while determining the label of an interaction.

Descriptive statistics of various content on the twitter like Tweets, Emoji, URLs and Images are below:

Number of Tweets	Mean	Min	Max
Toxic	13.28	3.0	304.0
Non-Toxic	7.15	3.0	99.0

(a)

Number of Emoji	Mean	Min	Max
Toxic	6.72	0.0	290.0
Non-Toxic	3.51	0.0	60.0

(b)

Number of URLs	Mean	Min	Max
Toxic	2.70	0.0	73.0
Non-Toxic	1.63	0.0	26.0

(c)

Number of Images	Mean	Min	Max
Toxic	1.18	0.0	20.0
Non-Toxic	0.86	0.0	12.0

(d)

Overall distribution of the instances as Toxic interactions constitute the 17.15% of the dataset, while 79.51% remains as Non-Toxic. A minority group of interactions with 3.34% comprises the Unclear instances where annotators agreed that no conclusion could be derived.

Use of multimodal elements such as image, video, and emoji, is clearly higher, suggesting that the incorporation of these different modalities in the analysis of this dataset will be critical for a reliable outcome.

Toxic	Non-Toxic	Unclear
118 (17.15%)	547 (79.51%)	23 (3.34%)

Table 6: Overall distribution of the data instances over the three labels.

Type of URLs	Number of URLs
Image URLs	140 (43.88%)
Video URLs	44 (13.79%)
Text URLs	48 (15.04%)

Table 7: Different types of URLs in **toxic** interactions.

Discussion:

The unique characteristics concerning: (i) adolescent population and (ii) interaction-based design.

The dataset is an important contribution to the research community by the author, as ground truth to provide a better understanding of online toxic behaviour as well as training machine learning models and performing time-series analysis.

Researchers can develop guidelines for different kinds of toxic behaviour such as harassment and hate speech, and annotate the dataset accordingly.

Haque et al
Deep Learning for Suicide and Depression Identification with Unsupervised Label Correction

Link to the paper: https://arxiv.org/pdf/2102.09427.pdf

Dataset and the web-scraping script available in the authors' code.

https://github.com/ayaanzhaque/SDCNL

Reviewer: **Angel - Completed**

Dataset:

The method is outlined in Figure 1.

Embedding Models

1. Authors begin by processing text data scraped from Reddit with word embedding models. As a requirement is embedding models optimized to work with phrases, sentences, and paragraphs, the authors experiment with 3 state-of-the-art transformers:
    - Bidirectional Encoder Representations from Transformers (BERT)
    - Sentence-BERT, and
    - Google Universal Sentence Encoder (GUSE)

- Some classifiers require word level representations for embeddings, while others require document level representations. BERT outputs both multi-dimensional word level embeddings as well as document level embeddings, which are provided by CLS tokens.
- Depending on what the classifier requires, authors vary the inputted embeddings to match the classifier's requirement.
- In addition, they also experiment on three vectorizers as baselines:
    - Term Frequency–Inverse Document Frequency (TFIDF),
    - Count Vectorizer (CVec), and
    - Hashing Vectorizer (HVec).

Label Correction

2. These embeddings are then processed with an unsupervised dimensionality reduction algorithm.
- The authors experiment with three separate dimensionality reduction algorithms:
    - Principal Component Analysis (PCA),
    - Deep Neural Autoencoders, and
    - Uniform Manifold Approximation and Projection (UMAP)

3. The reduced embeddings are then inputted into a clustering-based algorithm which predicts new labels in an unsupervised manner, meaning it is independent of

noise. They use the Gaussian Mixture Model (GMM) as our clustering algorithm.

- Each word embedding is now associated with two labels: the original labels based on the subreddit, which are the ground-truth labels, and the new labels resulting from unsupervised clustering

4. These alternate labels are compared against the ground-truth labels using a confidence based thresholding procedure in order to correct the ground-truth labels. The tuned threshold ensures only predicted labels with high confidence are used to correct the ground-truth, preventing false corrections

5. The corrected set of labels are then used to train a deep neural network in a supervised fashion.

Classifications

With a corrected label set, we train our deep neural networks to determine whether the posts display depressive or suicidal sentiment. S

For experimentation, authors tested four deep learning algorithms:
- dense neural network,
- Convolutional Neural Network (CNN),
- Bidirectional Long ShortTerm Memory Neural Network (BiLSTM), -
- Gated Recurrent Unit Neural Network (GRU).

For baselines, authors evaluated three standard machine learning models:
- Logistic Regression (LogReg),
- Multinomial Naive Bayes (MNB), and
- support-vector machine (SVM).

Datasets

Authors develop a primary dataset based on suicide or depression classification, web-scraped from **Reddit.** Collected data from subreddits using the Python Reddit API. Specifically scraping from two sub-reddits:
- **r/SuicideWatch and**
- **r/Depression.**

The dataset contains 1,895 total posts.
- Utilise two fields from the scraped data: the original text of the post as our inputs, and the subreddit it belongs to as labels.
- Posts from r/SuicideWatch are labeled as suicidal, and
- posts from r/Depression are labelled as depressed.

Furthermore, authors use the Reddit Suicide C-SSRS dataset to verify their label correction methodology.
The C-SSRS dataset contains 500 Reddit posts from the subreddit r/depression.
These posts are labeled by psychologists according to the Columbia Suicide Severity Rating Scale, which assigns progressive labels according to severity of depression.

Authors use this dataset to validate our label correction method since the labels are

clinically verified and from the same domain of Reddit.

To further validate the label correction method, they use the IMDB large movie dataset. The dataset is a binary classification task which contains 50,000 polar movie reviews. We use a random subset of samples for evaluation.

Authors: Nguyen et al
Title: BERTweet: A pre-trained language model for English Tweets

Link to the paper: https://arxiv.org/pdf/2005.10200.pdf

This paper includes pre-training and fine-tuning.

Reviewer: **Angel - Completed**

Dataset:

Note: see also by the same authors, shown in the next page.
WNUT-2020 Task 2: Identification of Informative COVID-19 English Tweets
https://arxiv.org/abs/2010.08232

Architecture
BERTweet uses the same architecture as BERTbase, which is trained with a masked language modeling objective.
BERTweet pre-training procedure is based on RoBERTa (Liu et al., 2019) which optimizes the BERT pre-training approach for more robust performance

Pre-training data
80GB pre-training dataset of uncompressed texts, containing 850M Tweets (16B word tokens). Here, each Tweet consists of at least 10 and at most 64 word tokens.

The dataset is a concatenation of two corpora:

1. Download the general Twitter Stream grabbed by the Archive Team, containing 4TB of Tweet data streamed from 01/2012 to 08/2019 on Twitter.

- To identify English Tweets, authors employ the language identification component of fastText (Joulin et al., 2017).
- Tokenize those English Tweets using “TweetTokenizer” from the NLTK toolkit (Bird et al., 2009).
- Use the emoji package to translate emotion icons into text strings (here, each icon is referred to as a word token).
- Normalize the Tweets by converting user mentions and web/url links into special tokens @USER and HTTPURL, respectively.
- Filter out retweeted Tweets and the ones shorter than 10 or longer than 64 word tokens. This pre-process results in the first corpus of 845M English Tweets
- https://archive.org/details/twitterstream

2. Stream Tweets related to the COVID-19 pandemic, available from 01/2020 to 03/2020.3
- Apply the same data pre-process step as described above, thus resulting in the

second corpus of 5M English Tweets.

Optimization

This utilizes the RoBERTa implementation in the fairseq library.
Set a maximum sequence length at 128, thus generating 850M × 25 / 128 ≈ 166M sequence blocks.
Optimize the model using Adam, and
Use a batch size of 7K across 8 V100 GPUs (32GB each) and a peak learning rate of 0.0004.
Pre-train BERTweet for 40 epochs in about 4 weeks
Using the first 2 epochs for warming up the learning rate,
equivalent to 166M × 40 / 7K ≈ 950K training steps.

Downstream task datasets and normalizations

For POS tagging, the authors use three datasets Ritter11- T-POS, ARK-Twitter4 and TWEEBANK-V2 5

For NER, they employ datasets from the WNUT16 NER shared task nd the WNUT17 shared task on novel and emerging entity recognition

For text classification, they employ the 3-class sentiment analysis dataset from the SemEval2017 Task 4A and the 2-class irony detection dataset from the SemEval2018 Task 3A.

The authors use a “soft” normalization strategy to all of the experimental datasets by translating word tokens of user mentions and web/url links into special tokens @USER and HTTPURL, respectively, and

converting emotion icon tokens into corresponding strings.

They also apply a “hard” strategy by further applying lexical normalization dictionaries to normalize word tokens in Tweets.

Fine-tuning

For POS tagging and NER, the authors append a linear prediction layer on top of the last Transformer layer of BERTweet with regards to the first subword of each word token

For text classification, the authors append a linear prediction layer on top of the pooled output.

They employ the transformers library to independently fine-tune BERTweet for each task and each dataset in 30 training epochs.

Use AdamW, with a fixed learning rate of 1.e-5 and a batch size of 32.

Then, compute the task performance after each training epoch on the validation set - applying early stopping when no improvement is observed after 5 continuous epochs, and

Select the best model checkpoint to compute the performance score on the test set.

Repeat this fine-tuning process 5 times with different random seeds, i.e. 5 runs for each task and each dataset.

Baselines
Our main competitors are the pre-trained language models RoBERTabase and XLM-Rbase.

Nguyen et al. Identification of Informative COVID-19 English Tweets

https://competitions.codalab.org/competitions/25845

Annotation guideline

Authors define the guideline to annotate a COVID19 related Tweet with the "INFORMATIVE" label if the Tweet mentions suspected cases, confirmed cases, recovered cases, deaths, number of tests performed as well as location or travel history associated with the confirmed/suspected cases.

Tweet collection

The authors crawled the COVID-19 related Tweets.

Collect a general Tweet corpus related to the COVID-19 pandemic based on a predefined list of keywords, including:

- "coronavirus", "covid-19", "covid 19", "covid 2019", "covid19", "covid2019", "covid-2019".

Utilize the Twitter streaming API to download real-time English Tweets containing at least one keyword from the predefined list.

Filter out Tweets containing less than 10 words (including hashtags and user mentions) as well as Tweets from users with less than five hundred followers - this to help reduce the rate of Tweets with fake news.

To handle the duplication problem:
(i) authors remove Retweets starting with the "RT" token, and
(ii) in cases where two Tweets are the same after lowecasing as well as removing hashtags and user mentions, the earlier Tweet is kept and the subsequent Tweet will be filtered out as it tends to be a Retweet.

Applying these filtering steps results in a final corpus of about 23M COVID-19 English Tweets.

Annotation process

From the above corpus, authors select Tweets which are potentially informative,

containing predefined strings relevant to the annotation guideline such as:

- “confirm”, “positive”, “suspected”, “death”, “discharge”, “test” and “travel history”.

Then remove similar Tweets with the token based cosine similarity score that is equal or greater than 0.7, resulting in a dataset of “INFORMATIVE” candidates.

Then randomly sample 2K Tweets from this dataset for the first phase of annotation.

Three annotators are employed to independently annotate each of the 2K Tweets with one of the two labels “INFORMATIVE” and “UNINFORMATIVE”.

Naseem et al
Benchmarking for Public Health Surveillance tasks on Social Media with a Domain-Specific Pretrained Language Model

Link to Paper: https://arxiv.org/abs/2204.04521

PHS-BERT

***Note: Available in John Snow Labs Model Hub.**

Upstream (pre-training) + Downstream (fine-tuning)

Reviewer: **Mitanshu** - **Completed**

Dataset: https://github.com/usmaann/RHMD-Health-Mention-Dataset

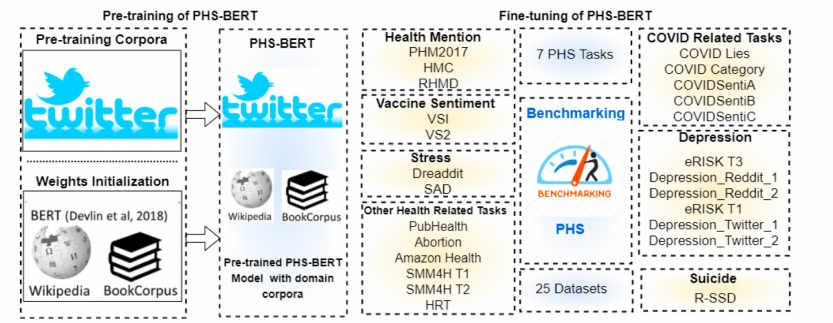


Figure 1: An overview of pretraining, fine-tuning, and the various tasks and datasets used in PHS benchmarking

Architecture

- PHS-BERT has the same architecture as BERT. Fig. 1 illustrates an overview of pretraining, fine-tuning, and datasets used in this study. Author describes BERT and then the pretraining and fine-tuning process employed in PHS-BERT.

Pre-Training

- BERT was trained on general domain corpus (e.g.,Bookcorpus, Wikipedia, etc.) may yield poor performance on domain-specific task. To address this problem PHS-BERT was presented.

- PHS-BERT is trained on a health-related corpus collected from user-generated content.

- PHS-BERT outperforms other SOTA PLMs on 25 datasets from different social media platforms related to 7 different PHS tasks, showing that PHS is robust and

generalizable.

- Authors followed the standard pretraining protocols of BERT and initialized PHS-BERT with weights from BERT during the training phase instead of training from scratch and used the uncased version of the BERT model.

- PHS-BERT trained on a corpus of health-related tweets that were crawled via the Twitter API. Focusing on the tweets related to PHS.

- Retweet tags were deleted from the raw corpus, and URLs and usernames were replaced with HTTP-URL and @USER, respectively. Additionally, the Python emoji3 library was used to replace all emoticons with their associated meanings

- The HuggingFace4, an open-source python library, was used to segment tweets. Each sequence of BERT LM inputs is converted to 50,265 vocabulary tokens

Fine-Tuning

- Author applied the pretrained PHS-BERT in the binary and multi-class classification of different PHS tasks

- We fine-tuned the PLMs in downstream tasks.

- Author used the train library to fine-tune each model independently for each dataset.

- Author used the embedding of the special token [CLS] of the last hidden layer as the final feature of the input text.

- Author adopted the multilayer perceptron (MLP) with the hyperbolic tangent activation function and used Adam optimize.

- The models are trained with a one cycle policy at a maximum learning rate of 2e-05 with momentum cycled between 0.85 and 0.95.

Experimental Analysis

- Authors evaluated and benchmarked the performance of PHS-BERT on 7 different PHS classification tasks collected from popular social platforms.
-
- Author relied on the datasets that are widely used in the community

Table 1: Statistics of the datasets used. We used the Stratified 5-Folds cross-validation (CV) strategy for train/test split if original datasets do not have an official train/test split.

Task (Classification)	Dataset	Platform	# of Samples	# of Classes	Training Strategy Used
Suicide	R-SSD (Cao et al., 2019)	Reddit	500 Users	5	Stratified 5-Folds CV
Stress	Dreaddit (Turcan and McKeown, 2019)	Reddit	3553 Posts	2	Official Split
	SAD (Mauriello et al., 2021)	SMS-like	6850 SMS	2	Official Split
Health Mention	PHM (Karisani and Agichtein, 2018)	Twitter	4635 Posts	4	Stratified 5-Folds CV
	PHM (Karisani and Agichtein, 2018)	Twitter	4635 Posts	2	Stratified 5-Folds CV
	HMC2019 (Biddle et al., 2020)	Twitter	15393 Posts	3	Stratified 5-Folds CV
	RHMD (Naseem et al., 2022b)	Reddit	3553 Posts	4	Stratified 5-Folds CV
Vaccine Sentiment	VS1 (Dunn et al., 2020)	Twitter	9261 Posts	3	Stratified 5-Folds CV
	VS2 (Müller and Salathé, 2019)	Twitter	18522 Posts	3	Stratified 5-Folds CV
COVID Related	Covid Lies (Hossain et al., 2020)	Twitter	3204 Posts	3	Stratified 5-Folds CV
	Covid Category (Müller et al., 2020)	Twitter	4328 Posts	2	Stratified 5-Folds CV
	COVIDSentiA (Naseem et al., 2021d)	Twitter	30000 Posts	3	Stratified 5-Folds CV
	COVIDSentiB (Naseem et al., 2021d)	Twitter	30000 Posts	3	Stratified 5-Folds CV
	COVIDSentiC (Naseem et al., 2021d)	Twitter	30000 Posts	3	Stratified 5-Folds CV
Depression	eRISK T3 (Losada and Crestani, 2016)	Reddit	190 Users	4	Stratified 5-Folds CV
	Depression_Reddit_1 (Naseem et al., 2022a)	Reddit	3553 Posts	4	Stratified 5-Folds CV
	eRISK19 T1 (Losada and Crestani, 2016)	Reddit	2810 Users	2	Official Split
	Depression_Reddit_2 (Pirina and Çöltekin, 2018)	Reddit	1841 Posts	2	Stratified 5-Folds CV
	Depression_Twitter_1	Twitter	1793 Posts	3	Stratified 5-Folds CV
	Depression_Twitter_2	Twitter	10314 Posts	2	Stratified 5-Folds CV
Other Health related	PubHealth (Kotonya and Toni, 2020)	News Websites	12251 Posts	4	Official Split
	Abortion (Mohammad et al., 2016)	Twitter	933 Posts	3	Official Split
	Amazon Health (He and McAuley, 2016)	Amazon	2003 Posts	4	Official Split
	SMM4H T1 (Weissenbacher et al., 2018)	Twitter	14954 Posts	2	Official Split
	SMM4H T2 (Weissenbacher et al., 2018)	Twitter	13498 Posts	3	Official Split
	HRT (Paul and Dredze, 2012)	Twitter	2754 Posts	4	Stratified 5-Folds CV

- To evaluate the performance, we used F1-score and the relative improvement in marginal performance ($\Delta M P$) used in a previous similar study.

Baselines

- We evaluated the performance of PHS-BERT with various SOTA existing PLMs in different domains.

- We compared the performance with BERT , ALBERT , and DistilBERTpretrained with general corpus, BioBERT pretrained in the biomedical domain, CT-BERT and BERTweet pretrained on covid related tweets and MentalBERT pretrained on corpus from Reddit from mental health-related subreddits.

Results

- Table 2 summarizes the results of the presented PHS-BERT in comparison to the baselines.
- We observe that the performance of PHS-BERT is higher than SOTA PLMs on all tested tasks and datasets.

Dataset	BERT	ALBERT	distilBERT	CT-BERT	BioBERT	BERTweet	MentalBERT	Ours	ΔMP_{BERT}	ΔMP_{SB}
Suicide Ideation Task										
R-SSD (Cao et al., 2019)	25.72	23.07	26.96	18.67	23.51	24.82	17.35	**30.28**	18.45↑	12.79↑
Stress Detection Task										
Dreaddit (Turcan and McKeown, 2019)	78.55	79.43	78.22	81.46	78.34	80.03	80.89	**82.89**	5.60↑	1.78↑
SAD (Mauriello et al., 2021)	92.66	91.11	91.47	91.11	93.92	94.17	93.23	**94.75**	2.28↑	0.62↑
Average	85.61	85.27	84.85	86.29	86.13	87.10	87.06	**88.82**	3.80↑	2.00↑
Health Mention Task										
PHM (Multi-class) (Karisani and Agichtein, 2018)	86.21	80.05	85.06	82.02	82.22	85.59	87.76	**89.38**	3.72↑	1.87↑
PHM (Binary) (Karisani and Agichtein, 2018)	91.89	90.53	90.64	92.17	89.62	92.12	92.29	**93.27**	1.52↑	1.07↑
HMC2019 (Biddle et al., 2020)	88.99	87.22	88.01	90.82	86.27	90.65	90.17	**91.71**	3.09↑	0.99↑
RHMD	74.20	69.02	73.22	72.87	72.25	74.66	75.28	**77.16**	5.48↑	2.53 ↑
Average	85.07	81.71	84.23	84.47	82.59	85.76	86.38	**87.38**	3.34↑	1.76 ↑
Depression Detection Task										
eRisk T3 (Losada and Crestani, 2016)	64.56	64.78	67.33	63.17	64.86	63.56	67.75	**68.98**	6.95↑	1.84↑
Depression_Reddit_1	22.39	21.09	21.95	24.21	24.00	20.84	21.95	**28.75**	29.73↑	19.56↑
eRisk T1 (Losada and Crestani, 2016)	93.72	93.79	93.34	86.74	91.73	91.92	94.30	**94.52**	0.86↑	0.24↑
Depression_Reddit_2 (Pirina and Çöltekin, 2018)	91.33	90.72	91.01	68.16	90.53	91.75	92.70	**93.36**	2.25↑	0.72↑
Depression_Twitter_1	64.17	51.70	66.71	57.11	64.12	64.24	72.95	**76.18**	19.01↑	4.49↑
Depression_Twitter_2	96.99	96.79	96.70	96.96	96.59	96.87	97.09	**97.12**	0.14↑	0.03↑
Average	72.19	69.81	72.84	66.06	71.97	71.53	74.46	**76.49**	6.03↑	2.76↑
Vaccine Sentiment Task										
VS1 (Dunn et al., 2020)	74.14	70.00	73.95	79.92	73.30	76.81	71.56	**79.96**	7.96↑	0.05↑
VS2 (Müller and Salathé, 2019)	76.60	74.82	75.91	81.73	76.77	79.10	77.65	**82.24**	7.46↑	0.63↑
Average	75.37	72.41	74.93	80.84	75.04	77.96	74.61	**81.10**	7.70↑	0.34↑
COVID Related Task										
Covid Lies (Hossain et al., 2020)	92.96	91.53	92.14	92.24	93.79	91.07	94.60	**95.35**	2.60↑	0.80↑
COVID Category (Müller et al., 2020)	93.98	93.94	94.35	95.29	93.72	93.45	94.97	**95.83**	1.99↑	0.57↑
COVIDSentiA (Naseem et al., 2021d)	90.90	90.81	90.90	78.96	90.41	66.30	91.55	**93.97**	3.41↑	2.67↑
COVIDSentiB (Naseem et al., 2021d)	91.31	89.88	91.06	86.85	91.02	89.46	92.06	**93.44**	2.36↑	1.52↑
COVIDSentiC (Naseem et al., 2021d)	91.24	83.72	90.77	84.83	90.55	61.78	91.66	**93.11**	2.03↑	1.60↑
Average	92.08	89.98	91.84	87.63	91.90	80.41	92.97	**94.34**	2.48↑	1.49↑

Conclusion

- Results demonstrate that using domain-specific corpora to train general domain LMs improves performance on PHS tasks. On all 25 datasets related to 7 different PHS tasks, PHS-BERT outperforms previous state-of-the-art PLMs.

Author Farruque et al
Title **Depression Symptoms Modelling from Social Media Text: A Semi-supervised Learning Approach**

Model: MentalBERT

Reviewer: **Angel - Completed**

https://arxiv.org/pdf/2209.02765.pdf

Dataset:

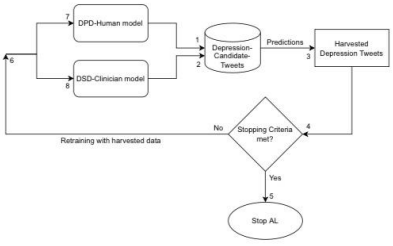


Fig. 2 Semi-supervised learning process at a high level.

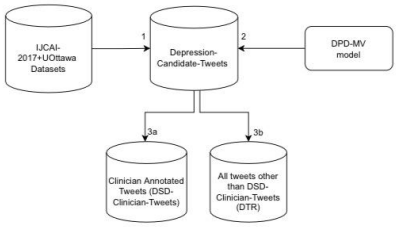


Fig. 3 DSD-Clinician-Tweets and DTR curation process

Semi-supervised Learning (SSL) framework which uses an initial supervised learning model that leverages

1) a state-of-the-art large mental health forum text pre-trained language model further fine-tuned on a clinician annotated DSD dataset,

2) a Zero-Shot learning model for DSD, and couples them together to harvest depression symptoms related samples from our large self-curated Depression Tweets Repository (DTR).

DTR is created from the samples of tweets in self-disclosed depressed users Twitter timeline from two datasets, including one of the largest benchmark datasets for user-level depression detection from Twitter.

This further helps preserve the depression symptoms distribution of self-disclosed Twitter users tweets.

Subsequently, iteratively retraininitial DSD model with the harvested data.

Author Abubakar Alhassan et al
Title of paper Self-harm: detection and support on Twitter

https://arxiv.org/abs/2104.00174

- Reviewer: Angel - Completed

Dataset:

This study utilises a custom crawler to retrieve relevant tweets from self-reporting users and relevant organisations interested in combating self-harm.

Through textual analysis, authors identify six major categories of self-harming users consisting of:
- inflicted,
- anti-self-harm,
- support seekers,
- recovered,
- pro-self-harm and
- at risk.

The inflicted category dominates the collection.

From an engagement perspective, study shows how online users respond to the information posted by self-harm support organisations on Twitter.

By noting the most engaged organisations, study applies a useful technique to uncover the organisations' strategy. ELABORATE

The online participants show a strong inclination towards online posts associated with mental health related attributes.

The study is based on the premise that social media can be used as a tool to support proactive measures to ease the negative impact of self-harm.

The authors offer ways to prevent potential users from engaging in self-harm and support

affected users through a set of recommendations.

Author Zogan et al
Title of paper DepressionNet: A Novel Summarization Boosted Deep Framework for Depression Detection on Social Media

https://arxiv.org/abs/2105.10878

Reviewer: **Angel - Completed**

Dataset:

The authors propose a novel computational framework for automatic depression detection that initially selects relevant content through a hybrid extractive and abstractive summarization strategy on the sequence of all user tweets leading to a more fine-grained and relevant content.

The content then goes to our novel deep learning framework comprising of a unified learning machinery comprising of:

- Convolutional Neural Network (CNN) coupled with
- attention-enhanced Gated Recurrent Units (GRU) models.

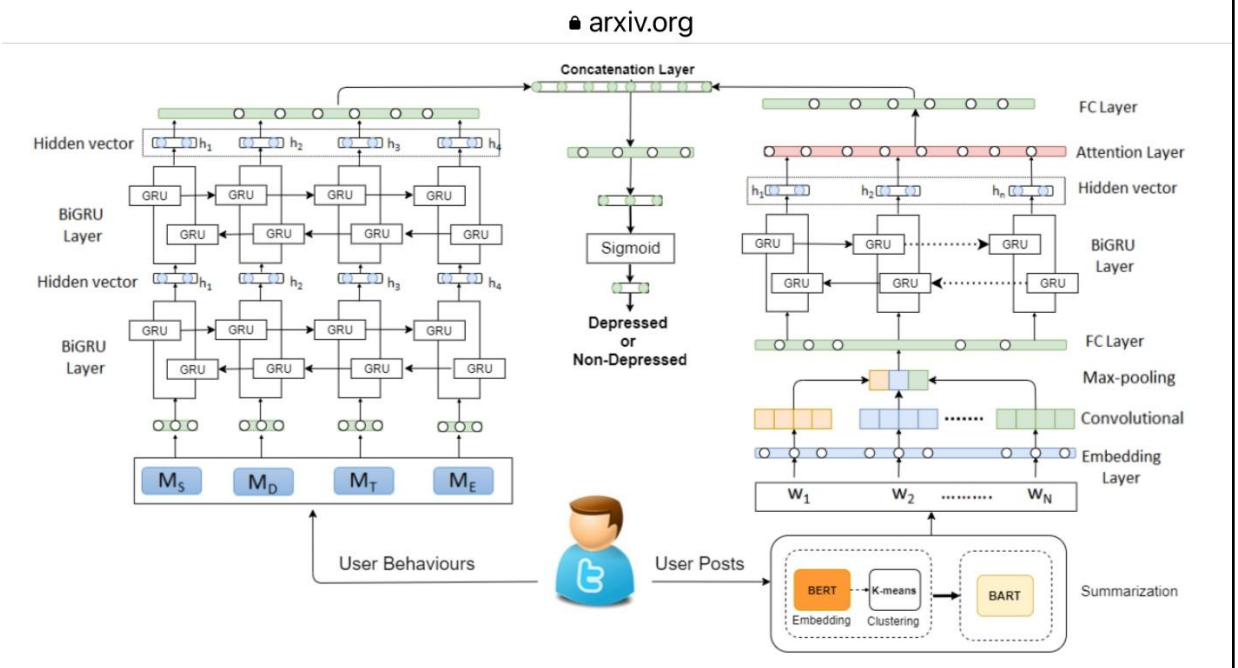


Figure 2: Proposed Framework (DepressionNet) for Depression Detection on Social Media

Author Kumar Suman et al.
Title of paper A Novel Sentiment Analysis Engine for Preliminary Depression Status Estimation on Social Media

https://arxiv.org/abs/2011.14280

Reviewer: **Angel** - **Completed**

Dataset:

The model consists of a RoBERTa based siamese sentence classifier that compares a given tweet (Query) with a labeled set of tweets with known sentiment (Standard Corpus).

To tune the pre-trained RoBERTa network,

The authors first processed the text from the public dataset to remove unwanted artifacts (such as URLs), to avoid unforeseen errors in prediction.

Once pre-processed the clean text is tokenized and paired up, with binary labels:

0, if (x1, x2) are of same class
1, if (x1, x2) are of different class.

The formed pairs of data are used to tune the Sentence-BERT model with the loss estimate function.

Once tuned the same network is used to generate corresponding sentence embedding for a standard set of tweets.

The vector set of embeddings so formed are clustered using K-Means clustering mechanism. The individual cluster averages are used as an effective and fast inference standard.

The inference process is highlighted in (Figure 3).

The similarity between the query (q) and a class average embedding (Ai) is calculated.

The appropriate class is selected as per maximum similarity criteria.

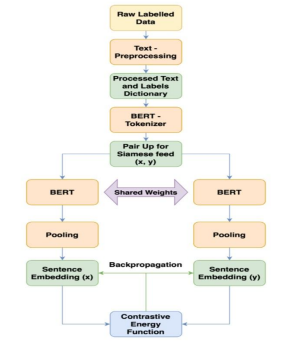


Figure 2: Training process of the Siamese Sentence-BERT network visualized

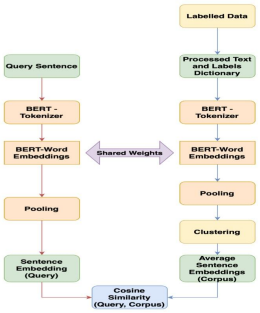


Figure 3: Inference process for a processed general query sentence

Author Zhang et al
Title of paper Monitoring Depression Trend on Twitter during the COVID-19 Pandemic
Upstream?
Downstream?

https://arxiv.org/abs/2007.00228

Reviewer: **Buket konuk-Hirst - Completed**

Dataset:

Summary:

People have been increasingly relying on social media platforms such as Facebook, Twitter, and Instagram to express their feelings. During COVID-19 period, social media witnessed a spike in depression terms.
Transformer-based models trained with by far the largest dataset on depression. We have analyzed our models' performance in comparison to existing models and verified that the large training set we compile is beneficial to improving the models' performance.

Tweet collection:

We created a dataset of **5,150 Twitter users,** including half identified depression users and half control users, along with their tweets within the past three months and their Twitter

activity data. Constructed **32,420 tweet chunks, 250 words each in the dataset.**
We use the Tweepy API to obtain **41.3 million tweets** posted from March 23rd to April 18th and the information of their authors. To find targeted twitter users, used regular expression to find these authors by examining their tweets and descriptions.

Once we have found the targeted Twitter users, we use the Tweepy API to retrieve the public tweets posted by these users within last 3 months

Methodology:

Transformer-based deep learning language models + we build a more accurate classification model upon the deep learning models along with linguistic analysis of dimensions including personality, **Linguistic Inquiry and Word Count (LIWC)** - a **well-validated psycholinguistic dictionary** (Tausczik and Pennebaker, 2010), sentiment features and demographic information.
We build a pipeline to monitor public depression trend by aggregating individuals' past tweets within a time frame using our model, and analyze the depression level trends during COVID-19, thus shedding light on the psycho- logical impacts of the pandemic.

Pre-training:

The preprocessed tweet chunk datasets are then passed to deep learning models for training

Fine-tuning and Classification:

We estimate individuals' sentiments using Valence Aware Dictionary and Entiment Reasoner (VADER). **VADER is a lexicon and rule-based model** developed by researchers from Georgia Insti- tute of Technology (Hutto and Gilbert, 2014). We aggregate a users tweets into a single chunk, apply VADER, and retrieve its scores for positive and negative emotions

Previous psychological studies show differences in depression rates among people of different ages and of different genders. Used M3-inference model label each user with gender and age group.

After chunking and preprocessing, on average, each user has 6-7 text chunks, making the actual sizes of the 4,650-user train-val set and the 500-user testing set to be 29,315 and 3,105, respectively. The preprocessed tweet chunk datasets are then passed to deep learning models for training.

We set up 2 baseline models, **multi-channel CNN and bidirectional LSTM** with context-aware attention (Attention BiLSTM),

We use the pre- trained GloVe embedding (840B tokens, 300d vec- tors) (Pennington et al., 2014) augmented with the special tokens added during preprocessing.

We also train three representative transformer-based sequence classification models

BERT, RoBERTa and XLNet - with their own pretrained tokenizers augmented with the special tokens for tokenization.

Classification results:

Model	Train-Val Set	Accuracy	F1	AUC	Precision	Recall
Attention BiLSTM	1k users	70.7	69.0	76.5	70.9	67.3
	2k users	70.3	68.3	77.4	70.7	66.1
	4.65k users	72.7	71.6	79.3	72.1	71.1
CNN	1k users	71.8	72.6	77.4	72.7	72.6
	2k users	72.8	74.5	80.3	72.2	76.9
	4.65k users	74.0	70.9	81.0	77.4	68.9
BERT	1k users	72.7	74.4	79.8	72.0	76.9
	2k users	75.7	76.3	82.9	76.1	75.7
	4.65k users	76.5	77.5	83.9	76.3	78.8
RoBERTa	1k users	74.4	75.7	82.0	74.2	77.3
	2k users	75.9	77.9	83.2	73.8	**82.5**
	4.65k users	76.2	**78.0**	84.1	74.4	81.9
XLNet	1k users	73.7	75.1	80.7	73.2	77.2
	2k users	74.6	76.8	82.6	72.6	81.5
	4.65k users	**77.1**	77.9	**84.4**	**77.5**	78.3

: Chunk-level performance (%) of all 5 different models using training-validation sets of different sizes.

Author Zhou et al
Title of paper Detecting Community Depression Dynamics Due to COVID-19 Pandemic in Australia

https://arxiv.org/abs/2007.02325

Reviewer: **Mitanshu - Completed**

The recent COVID-19 pandemic has caused unprecedented impact across the globe.new approach based on multi-modal features from tweets and Term Frequency-Inverse Document Frequency (TF- IDF) is proposed to build depression classification models. Multi-modal features capture depression cues from emotion, topic and domain-specific perspectives

Lot of work is done in detection of depression due to covid, However, little work is done to detect depression dynamics at the state level or even more granular level such as suburb level. Such granular level analysis of depression dynamics not only can help authorities such as governmental departments to take corresponding actions more objectively in specific regions if necessary but also allows users to perceive the dynamics of depression over the time to learn the effectiveness of policies implemented by the government or negative effects of any big event.

The answers to the questions that we wish to find are

- How people's depression is affected by COVID-19 in the time dimension in the state level?
- How people's depression is affected by COVID-19 in the time dimension in local government areas?
- Can we detect the effects of policies/measures implemented by the government during the pandemic on depression?
- Can we detect the effects of big events on depression during the pandemic?
- How effective is the model in detecting people's depression dynamics?

Dataset:

To analyse the dynamics of depression during the COVID-19 pandemic period at a fine-grained level, Author collected tweets from Twitter users who live in different LGAs of NSW in Australia. The time span of the collected tweets is from 1 January 2020 to 22 May 2020 which covers dates that the first confirmed case of coronavirus was reported in NSW (25 January 2020) and the first time that the NSW premier announced the relaxing for the lockdown policy (10 Mary 2020).

Dataset were labeled into two class, namely **positive** and **negative**, positive means user were depressed and negative means users were not depressed.

SUMMARY OF LABELLED DATA USED TO TRAIN DEPRESSION MODEL.

Description	Size
Depressed tweets	~ 900K
Non-Depressed tweets	~ 900K

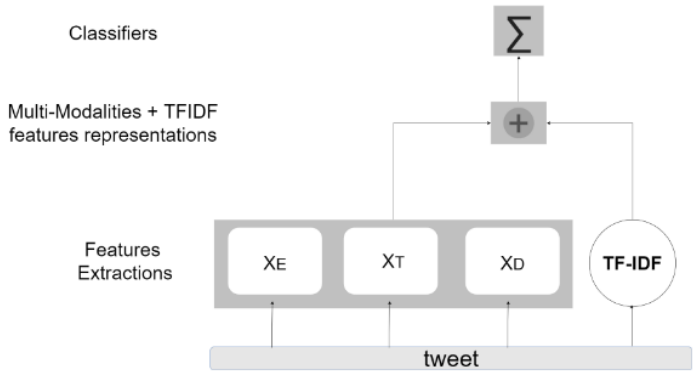
Model Building:
Tweets are shorts in length and single tweet does not provide sufficient word occurrences, author, therefore, combine multi-modal feature with Term Frequency- Inverse Document Frequency (TF-IDF) feature to analyze depressed tweet.
Author defined a set of features consisting of three modalities. These three modalities of features are as follows:

1. **Emotional features:** The emotion of depressed people is usually different from

non-depressed people, which influences their posts on social media.

2. **Topic-level features:** LDA is the most popular method used in the topic modelling to extract topics from text, which could be considered as a high-level latent structure of content. The idea of LDA is based on the assumption of a mixture of topics forms documents, each of which generates words based on their Dirichlet distribution of probability.
3. **Domain specific features:**Diagnostic and Statistical Manual of Mental Disorders 4th Edition (also called DSM-IV) is a **manual published by the American Psychiatric Association** (APA) that includes almost all currently recognized mental health disorder symptoms [39]. We, therefore, chose the **DSM-IV Criteria for Major Depressive Disorder** to describe **keywords** related to the **nine depressive symptoms**.

Pre-trained word2vec (Gensim pretrained model based on Wikipedia corpus) was used in this study to extend our keywords that are similar to these symptoms.



Model Training:
Three mainstream classification methods are used in this study to compare their performance, namely Logistic Regression (LR), Linear Discriminant Analysis (LDA) and Gaussian Naive Bayes (GNB) .Authors used scikit-learn libraries to import the three classification methods. The classification performance by these three methods were evaluated by 5-fold cross-validation. The experiments are conducted using Python 3.6.3 with 16-cores CPU.

We evaluate the classification models by using measure of Accuracy (ACC.), Recall (Rec.), Macro-averaged Precision (Prec.), and Macro-averaged F1-Measure (F1)

Results:

TABLE III
THE PERFORMANCE OF TWEET DEPRESSION DETECTION BASED ON MULTI-MODALTIES ONLY.

Features	Method	Precision	Recall	F1-Score	Accuracy
Multi-Modal	LR	0.842	0.828	0.832	0.833
	LDA	0.843	0.816	0.820	0.824
	GNB	0.873	0.814	0.818	0.825

TABLE IV
THE PERFORMANCE OF TWEET DEPRESSION DETECTION BASED ON TF-IDF ONLY.

Features	Method	Precision	Recall	F1-Score	Accuracy
TF-IDF	LR	0.908	0.896	0.900	0.901
	LDA	0.906	0.893	0.897	0.898
	GNB	0.891	0.873	0.877	0.879

TABLE V
THE PERFORMANCE OF TWEET DEPRESSION DETECTION BASED ON MULTI-MODALTIES + TF-IDF.

Features	Method	Precision	Recall	F1-Score	Accuracy
MM+TF-IDF	LR	0.908	0.899	0.902	0.903
	LDA	0.912	0.899	0.903	0.904
	GNB	0.891	0.874	0.878	0.879

Author fed the LR with (MM + TF-IDF) model with these tweets, and the model found that nearly 2 million tweets were classified as depressed tweets

Below figure presents the overall community depression dynamics in NSW with the confirmed cases of COVID-19 together during the study period between 1 January 2020 and 22 May 2020. “Depression level” refers to the proportion of the number of depressed tweets over the whole number of tweets each day.

Author Joloudari et al.
Title of paper BERT-Deep CNN: State-of-the-Art for Sentiment Analysis of COVID-19 Tweets

Review of transformers

https://arxiv.org/pdf/2211.09733.pdf

Reviewer: **Buket Konuk-Hirst - Completed**

Dataset:

Sentiment analysis is a powerful text analysis tool. It automatically detects and analyzes opinions and emotions from unstructured data. In a pandemic situation, analyzing social media texts to uncover sentimental trends can be very helpful in gaining a better understanding of society's needs and predicting future trends. We intend to study society's perception of the COVID-19 pandemic through social media using state-of-the-art BERT and Deep CNN models

Social media analytics revolves around the development and evaluation of informatics tools and frameworks for the collection, monitoring, analyzing, summarizing, and visualization of social media data. As shown in Figure 5, social media analytics consists of three stages namely “capture,” “understand,” and “present”

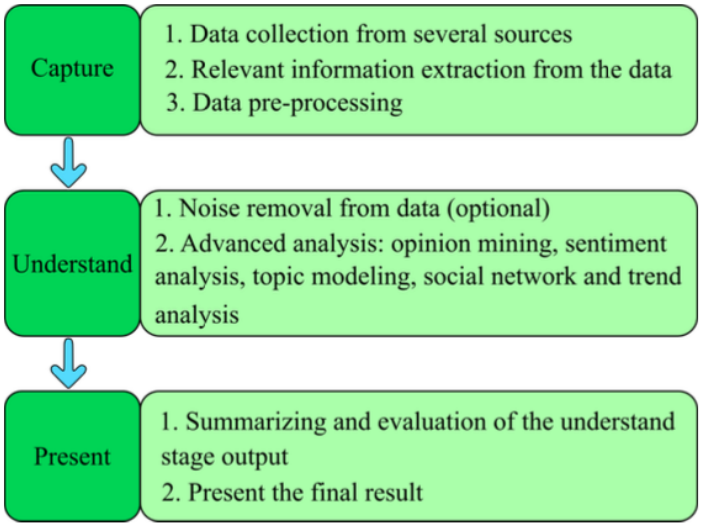


Figure 5. Process of social media analytics.

Table 2. The performance of several literature studies on sentiment analysis of COVID-19 tweets.							
Research Study	Method	Dataset	Number of Samples	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Babu et al. [39] (2020)	Ensemble of CT-BERT, RoBERTa, and SVM	WNUT-2020 Task 2	10,000	n/a	89.2 4	87.8 2	88.52
Gencoglu [38] (2020)	SVM + LaBSE embedding	Publicly available COVID-19 tweets between 4 January and 5 Apr (2020)	26,759,164	86.92	n/a	n/a	88.1
Rustam et al. [36] (2021)	Extra Tree Classifier using concatenated features	IEEE data port (on May 31, 2020)	7528	93	90	89	89
Basiri et al. [53] (2021)	Ensemble deep Learning using five base	Stanford Sentiment140	1,600,000	85.8	n/a	n/a	85.8
Learners							
To et al. [37] (2021)	BERT	Anti-vaccination tweets between 1	1,651,687	91.6	93.4	97.6	95.5

		January and 23 August (2020)					
Malla & Alphonse [40] (2021)	Majority Voting-based Ensemble Deep Learning(MVEDL)	WNUT-2020 Task 2	10,000	91.75	89.9 4	93.5 8	91.14
Glazkova et al. [35] (2021)	g2tmn	COVID-19 Fake News Dataset	10,700	n/a	n/a	n/a	98.69
Hayawi et al. [54] (2022)	BERT	ANTi-Vax	15,073	98	97	98	98

Author Roy et al
Title of paper A machine learning approach predicts future risk to suicidal ideation from social media data

URL of the paper: https://www.nature.com/articles/s41746-020-0287-6

Reviewer: Enqi Fu

Summary here - cut and paste from article relevant section concise

Abstract: Machine learning analysis of social media data represents a promising way to capture longitudinal environmental influences contributing to individual risk for suicidal thoughts and behaviors. Our objective was to generate an algorithm termed "Suicide Artificial Intelligence Prediction Heuristic (SAIPH)" capable of predicting future risk to suicidal thought by analyzing publicly available Twitter data. We trained a series of neural networks on Twitter data queried against suicide associated psychological constructs including burden, stress, loneliness, hopelessness, insomnia, depression, and anxiety. Using 512,526 tweets from N = 283 suicidal ideation (SI) cases and 3,518,494 tweets from 2655 controls, we then trained a random forest model using neural network outputs to predict binary SI status. The model predicted N = 830 SI events derived from an independent set of 277 suicidal ideators relative to N = 3159 control events in all non-SI individuals with an AUC of 0.88 (95% CI 0.86–0.90). Using an alternative approach, our model generates temporal prediction of risk such that peak occurrences above an individual specific threshold denote a ~7 fold increased risk for SI within the following 10 days (OR = 6.7 ± 1.1, P = 9 × 10−71). We validated our model using regionally obtained Twitter data and observed significant associations of algorithm SI scores with county-wide suicide death rates across 16 days in August and in October, 2019, most significantly in younger individuals. Algorithmic approaches like SAIPH have the potential to identify individual future SI risk and could be easily adapted as clinical decision tools aiding suicide screening and risk monitoring using available technologies.

Problem:

1. detection of suicide risk maybe low and around 60–70% of individuals at risk and seen by primary care practitioners prior to suicide attempts may go unidentified;
2. Further, a large number of at-risk patients may not receive appropriate care
3. For identify the biomarkers and predictive screening onto the individual risks, a significant challenge in this area is the relative low base rate of suicidal behavior in the population, making prospective studies impractical.

Research gap:

1. cross sectional screening for suicidal ideation (SI) may be insufficient to capture those in need of intervention.
2. In Twitter, people have less care of privacy, and most possible to collect the young users' data (18-35 y/o), since the second cause of suiside is age 15-34 y/o
3. People most likely to mentioned or re-post suiside or similar topics rather than physicians; moreover, they may easily gather on the internet and make the suiside agreement or something.
4. There were several ML algrothoms studies the detection and prediction about the

suiside risk on the social media such as Linguistic Inquiry and Word Count (LIWC), generated CNN on suiside posts, but the limitation was the 'suiside has to be mentioned' on the posts.'; another used ML classifier to identify the posts, but may not used the SI to analyzed in their model.

5. Novel techniques should attempt to prognosticate not only who will be at risk for suicidal thoughts and behaviors but also when they will be at risk, even for the people who have no mention the suiside at the detecting time (predict for the future)

Research question of this paper:

1. Our primary objective in this work was to generate a model capable of predicting individuals at risk as well as the odds of risk to suicidal thought within a given time frame.
2. Our approach involved training ML techniques to measure existing patterns in tweets that are predictive of suicide risk but do not yet explicitly express suicidal thoughts. Second, ML techniques based on psychological theories of suicide are yet to be developed.
3. Our approach centered on using ML constructs developed loosely around the interpersonal psychological theory for suicide (IPTS)30, the hopelessness model31, and associations of depression, anxiety, and insomnia with suicide risk.

Hypothesis: this algorithms integrating such approaches can offer nuanced and temporally sensitive tools to track and predict suicidal ideation in at risk individuals.

methods:

1. Sampling: in September of 2016 and proceeding until June 2019, we performed weekly queries for the term "I suicide thinking OR planning" to find users expressing suicidal ideation, via Twitter API.
2. Identification of SI tweets: scanning strategy to first identify potentially suicidal tweets that would later be evaluated by a psychiatrist rater, which at least match one word pattern in appendix 1;
3. Identification of Individuals with admitted past history of suicide attempt (SA) or suicide plan (SAP): scan and evalue by psychiatrist
4. Identification of individuals having died of suicide: public Twitter profiles of N = 9 celebrities for whom death by suicide had been reported as the cause of death in the media
5. Sample size: Total: 7,223,922 tweets. Control group: 6,385,079, and 838,843 derived from suicidal ideators
6. Using the M3inference package in python. The algorithm combines a convolutional neural network assessment of user profile picture, user written description, and user name to estimate the probability of the user belonging to an age class of ≤18, between 20–29, 30–39, and over 40.
7. Random geographic sampling of county-wide Twitter data: Using the Twitter API, we downloaded 50 tweets on an hourly basis using the query term "I" paired with the central longitude and latitude and square root of the square mileage for each of ninety-two US counties for which the most recent 2017 county-wide suicide death rate data was available from the CDC wonder database (https://wonder.cdc.gov/).
8. Algorithm generatio: two. The first objective was to generate a method to classify SI cases from controls and the second objective was to generate a method to assess when flagged individuals are likely to be at risk.
9. Generation of neural networks for psychological scoring of sentence content: Neural networks were generated to reduce text content into a score ranging between 0

and 1 for psychological constructs of burden, loneliness, stress, anxiety, insomnia, and depression. To generate neural networks, query terms according to Supplementary Table 2 were input into Twitter and no more than 1000 tweets were collected from the Twitter API. Three separate models for depression were generated (depression 1, depression 2, and depression 3).

10. Validation of neural networks: coded emotion with '0' and '1'
11. SI model cross validation strategy: training model and independent validation;
12. Generation of random forest classifiers: training under two situations-> (1) inputting N = 9 neural network derived model scores in addition to the subjectivity and polarity metrics from sentiment analysis and (2) inputting N = 9 neural network derived model scores independent of sentiment analysis metrics to understand the added value of neural network scores
13. Model assessment; temporal and statistical analysis

Analysis and results:

1. generate Twitter data based indicators of psychological constructs that are associated with SI and suicidal behavior using ML→ The **neural networks** constructed were designed to read text and infer psychological weights across a range of constructs including **stress, loneliness, burdensomeness, hopelessness, depression, anxiety, and insomnia**
2. apply ML to the generation of a cumulative indicator of suicide risk that predicts risk prior to the expression of suicidal ideation. → Using 512,526 tweets from N = 283 SI cases and 3,518,494 tweets from **2655 controls (9.6% SI)**, we used the **neural networks and sentiment polarity** generated above to generate an array of ten metrics for each tweet. The r**esulting matrix** was then used to **train random forest models** to predict a **binary classificatio**n of SI case status using a bootstrap aggregating approach to address the imbalanced class ratio in model construction resulting **in ten models.** We attempted to validate it in N = 277 SI individuals relative to **2961 controls (8.4% SI)**. As such the proportion of SI in each set was **similar to the observed rate** of 9.2% observed in a cross-national study of SI rates in the general population. **negative correlation between age** and model prediction (from 18-40 y/o); while gender no correlation. Identification of individuals **with past SA or suicide plan**, 75.2% increased risk of an individual with SI being a suiside-plan individual, and a significant association between the **number of SI tweets across SI individuals** with and without SAP status.
3. develop a temporal predictor of suicide risk periods and validate the technology on suicide decedents. → temporallly closer to the time of the SI event ($rho = -0.9$, $p < 2.2 \times 10^{-16}$), with t**he most significant association occuring at 6 days** from the SI event using a 21 day span (3 weeks) , while the highest OR is found at **1 day** prior to the SI event ($OR = 6.7 \pm 1.1$, $p = 9 \times 10^{-71}$).

4. Others time& region: Regionally generated SI scores model county-wide suicide rates. a significant correlation of mean **SI score per county with death rate** (Kendall's tau = 0.16, p = 0.021), also with the regression analysis; a significant association between **population size-by-SI score interaction and death rates** (Kendall's tau = 0.38, $p = 6.67 \times 10^{-8}$); **the minimum number of days** required to generate aggregated SI scores that would be predictive of death by suicide rates and determined a minimum of **sixteen days** was required (IRR = 1.91, 95% CI 1.30–2.82, df = 91, p = 0.0013); also, the model derived scores may sensitive to acute but impactful event in a short period of time

Future:

1. this tool, which we have termed “Suicide Artificial Intelligence Prediction Heuristic (SAIPH)”, could enable the suicide prevention community to screen for and monitor longitudinal changes in risk in or outside of care.
2. Senior population?
3. RNN and LSTM, they are powerful for the time series prediction, but limited to predict the actions which influenced by the external environment.

Author Bauerle et al
Title of paper **Symphony: Composing Interactive Interfaces for Machine Learning**

URL of the paper: https://dl.acm.org/doi/fullHtml/10.1145/3491102.3502102

Reviewer: **Amin Khan** - Completed

Abstract

- Studies have shown that ML interfaces have limited adoption in practice

- Not designed to be reused, explored, and shared by multiple stakeholders in cross-functional teams

- Symphony is a framework for composing interactive ML interfaces with task-specific, data-driven components that be used across platforms such as computational notebooks and web dashboards

- Symphony helped ML practitioners discover previously unknown issues like data duplicates and blind spots in models while enabling them to share insights with other stakeholders

Introduction

- Symphony combines the following principles to improve upon existing ML interfaces:

    - **Data-driven ML interfaces** derived from and updated with ML data and models
    - **Task-specific visualizations** for unstructured data and modern ML models
    - **Interactive exploration tools** for exploring different dimensions of an ML system
    - **Reusable components** that can be used, composed, and shared across different platforms

Documenting Data and Models

- Data statements are tailored to naturally language processing datasets
    - These guidelines describe what should be included documentation

- Symphony uses modular visualizations but they focus on simple **aggregate** visualizations without displaying data samples and do not support platforms where ML practitioners do their work

- Model cards include info ranging from model type and hyperparameters to metrics and ethical considerations

Visualization for Machine Learning

- Visualizations can help ML practitioners in tasks such as:
    - Auditing models for bias
    - Understanding the internals for deep learning architectures
    - Guiding automatic model selection

- Systems like Know Your Data and Facets are visualization dashboards for exploring unstructured data

- Visualization systems like Summit and Seq2Seq-Vis can help ML practitioners develop a better mental model of how they ML systems work and what they are learning

- Systems like MLCube, What-if Tool, Squares and AnchorViz focus on performance analysis and provide different views of a model's errors

- These various visualizations can be repackaged as **Symphony** components

- ML data and model visualizations are deployed as visual analytics dashboards separate from both interactive programming environments that ML practitioners work with and ML documentation shared with other stakeholders
    - Separation limits who can use visualizations to understand ML data and models but Symphony's purpose is to bridge that gap

Interactive Programming Environments

- While computational notebooks such as Jupyter have extensions for creating interactive visualizations, they are underused and hard to share

- Voilà tackles the challenge by exporting full Jupyter notebooks to a hosted website

- Since Symphony components are standalone JavaScript modules, future wrappers could integrate Symphony like components into data science environments like

Streamlit and Glinda

- **<u>Limitations of existing ML interfaces</u>**
    - Many tools require users to export their data into a specific format before loading it into a custom system or dashboard
    - 5 participants mentioned explicitly they do not use ML interfaces because they are not available in the environments where they work and that people *"would want to use easier tools"*
    - Lack of communication between stakeholders

Modular Components

- The building blocks of Symphony are independent modular components built for task-specific visualizations

- A **Symphony** component is a JavaScript module that renders a web-based visualization

- Svelte web framework used as the base of **Symphony** components but visualizations can be written using any JavaScript code or library

- Each **Symphony** component is passed 3 parameters: a metadata table, derived state variables like grouped tables, and references to raw data instances like images

Platform Wrappers

- The primary goal of using self-contained components is to compose and share them as flexible interfaces across different platforms
    - Done using wrappers which connect components with a backing platform

- Wrappers have 2 primary functions:
    - Passing data from a platform to **Symphony** in the correct format
    - Rendering **Symphony** components in the platform's UI

- Wrappers implemented for web UIs and Jupyter Notebook

- Components can be configured before export to be placed on different subpages and arranged within these pages to fit particular use cases

Interactive Exploration Tools

- Final key feature of Symphony is a set of tools for interacting with and exploring data

- For the web-based UI, state changes are also saved in the URL, allowing stakeholders to share specific findings

- Tools implemented that are important for specific components:

- **Data Filtering**
- **Grouping**
- **Instance selection**

- Users have 3 ways of using Symphony's interaction tools:
    - Through a UI toolbar
    - Symphony components
    - Code

Case Study 1: Validating and Sharing Data Patterns on a Dataset Creation Team

- For the first case study, a team that assembled, and labelled large ML datasets were first worked with

- Datasets composed of labeled images and videos which were published into an internal data repository

- Team used **Symphony** in 2 ways:
    - During dataset creation to detect errors in the data and labels
    - Reporting tool to given consumers of the dataset details about the data

- A participant noted the following:
    - *"There are a lot of neat things here, first, the filter carried over and it is so cool to see the data samples and metadata within the notebook"*

- The synchronized, reactive state let them validate insights from the filtered summary charts with the actual raw instance previews in the list view

- Map visualization was dubbed *"very useful"* especially when sharing reports of their data collection efforts with managers or policymakers

Case Study 2: Debugging Training Data on an Accessibility Team

- In the 2nd case study, the participating team was one that uses ML to make software applications more accessible

- Large dataset of icon screenshots was assembled

- Using the duplicates component in Symphony, the participants confirmed that a significant number of icons were duplicates

- When they used the grouping interaction to split the data by testing and training, they found a significant number of instances were duplicated across the 2 datasets

- The participants identified the problematic duplicates and removed them from the test set with a Python command

- Participant then explored the familiarity component and found many grey icons to which they wondered if the model might be overfit on the samples

- Notebook-based visualizations were useful to *"look into the data"* which was previously done manually using a file explorer outside of the notebook

- Main feature the team wanted was to combine the data and model findings to understand the impact of data changes

Case Study 3: Promoting Data Exploration for ML Novices on an Education Team

- Collaborated with a team focused on ML education

- Teach engineers about ML principles and techniques as well data/model tools

- 2 representative datasets used:
    - 1 audio dataset for data analysis
    - 1 image dataset for model analysis

- Used cross-filtering and grouping to combine the projection visualization with the summary component to spot misclassified samples

- Used confusion matrix visualization in combo with the filtering tool

- Described resulting confusion matrix as a *"fantastic graphic"*

- The team found **Symphony** valuable as they thought it could play a part in one of their lessons, as *"promoting looking at data is extremely important"*

- Liked the option to assemble visualizations such as when their students learn how to *"communicate findings to executives"* and *"graphing the relevant, and hiding the irrelevant"*

Limitations and Future Work

- Ways **Symphony** could be further improved after all their research and development:
    - Authoring components
    - Scaling past millions of data points
    - Beyond conventional data science platforms
    - Guided usage of **Symphony**

Conclusion

- **Symphony**'s visualizations helped ML teams find important issues such as data duplicates and model blind spots

- **Symphony** can encourage ML practitioners to want to use and share insights

- **Symphony** fosters a culture of shared ML understanding and encourages the creation of accurate, responsible, and robust AI products

Author Maric et al
Title of paper: A Research Software Engineering Workflow for Computational Science and Engineering

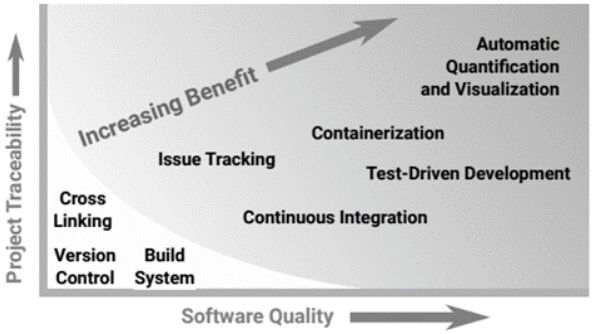
URL of the paper: https://arxiv.org/pdf/2208.07460.pdf

Reviewer: **Omar**

Abstract

This paper discusses the challenges that University research groups in Computational Science and Engineering face in terms of dedicating funding and personnel to Research Software Engineering (RSE). It argues that the lack of focus on RSE in CSE negatively impacts the quality and reproducibility of scientific output, slowing down CSE research and resulting in significant losses in time and funding. To address these challenges, the authors propose a RSE workflow for CSE that applies established software engineering practices, such as software testing, result visualization, and linking software to reports and publications. The workflow aims to improve the quality of research output in CSE by allowing researchers to easily find, reproduce, and extend published research ideas. It also introduces minimal work overhead for university research groups.

Introduction



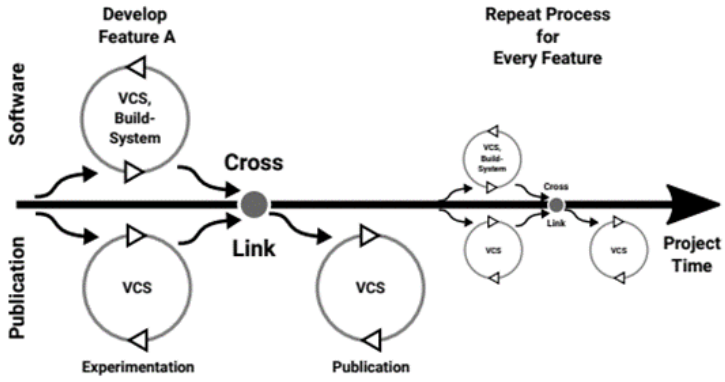
The workflow is designed for small research teams or individual researchers at universities and is intended to minimize work overhead. The minimum recommended components of the workflow include the use of a version control system for source files and experiment configurations, an established cross-platform build system for software artifacts, and the cross-linking of code, scientific publications, and data using persistent identifiers. Additional components of the workflow that can be included as the project progresses include issue tracking, continuous integration, containerization for testing and reproducible experiments, test-driven development, and automatic quantification and evaluation using

high-performance computing systems. The workflow is built on top of widely adopted open-source software and is intended to improve the quality of research output in CSE by allowing researchers to easily find, reproduce, and extend published research ideas.

Minimal workflow

This workflow argues that the use of version control is necessary to ensure the sustainability and reproducibility of research software, and to facilitate the transfer of knowledge between researchers.

- The workflow combines version control with research milestones and the scientific publication process to connect source code, research reports, and result data
- It recommends the use of a feature-based branching model in a version control system (such as git) to create new branches for each research idea that will be the basis for a scientific publication
- The authors recommend using automatic testing and continuous integration to ensure existing features continue to deliver results of similar quality when integrating new features, and the use of persistent identifiers for cross-linking research artifacts
- The adoption of this workflow can improve the quality and reproducibility of research output in CSE and can help to reduce the time and financial losses associated with the re-implementation of published CSE results



- Cross-linking the data sets with the publication allows researchers to draw new conclusions, perform more detailed analyses, apply the method outside of the original application range, and reuse and reproduce results from specific milestones.
- Cross-linking the software and data sets with persistent identifiers (PIDs) helps ensure long-term accessibility and availability, and allows for traceability of digital research artifacts.
- Cross-linking also allows for more accurate and efficient citation of digital research artifacts, as it allows for the citation of specific versions and configurations.

Full workflow

1. *Issue tracking*

   1.1. Issue tracking can significantly improve research software development at universities by providing documentation of the project status and simplifying communication with the scientific community.

   1.2. An issue tracking system allows for the creation of "issue cards" for bugs, ideas, research cooperation, and peer-review comments, which can be extended with links to source code and input data and attachments.

   1.3. The Kanban approach consists of a board with three columns (To Do, In Progress, and Done) and allows for the labeling and grouping of issues into project milestones. This can help to understand the status of the project and simplify the transition of research personnel.

2. *Test-Driven Development for Research Software*

   2.1. TDD consists of three phases: red (write tests, no code implementation), green (code implementation until tests pass), and refactor (refactor code for modular design).

   2.2. Top-down TDD for CSE research software involves writing high-level tests for CSE applications (e.g. partial differential equation solvers) first, then implementing the algorithms needed to pass those tests.

   2.3. Algorithms can be implemented from scratch or re-used from legacy code. If implemented from scratch, the TDD process is repeated recursively for sub-algorithms.

   2.4. Re-used sub-algorithms are assumed to be correct unless TDD testing shows otherwise.

   2.5. When the high-level application is not producing correct results, TDD can be used to determine which part of the implementation is causing the issue.

   2.6. TDD can improve the quality and reproducibility of research software by ensuring that it is thoroughly tested and easy to understand for others.

3. *Test quantification and visualization*

   Testing scientific software involves running multiple simulations, called "cases," and organizing the results in a way that allows for quick identification of failing tests and efficient re-running. This can be done using Jupyter notebooks, which contain a problem description and visualization of results, and allow for real-time analysis of test results. The notebooks also store results in a standardized, machine-readable format for use in publications and data cross-linking. Using Jupyter notebooks allows for live inspection of results and documentation of the parameter studies, including mathematical descriptions of the problems being solved and the initial

and boundary conditions. The notebooks also process and store secondary data generated by the simulations and can be used directly in scientific publications. Additionally, using a version control system allows for tracking of changes to the software and testing process.

4. *Containerization*

Containerization is a way to ensure reproducibility of research results by encapsulating the software environment in a container that can be run on any platform with a suitable container runtime. Containers are built using recipe files that specify the required dependencies, software, and data to be included in the container. It is important to use data from online resources with a guarantee of availability and to pull code from release pages or specific commits or tags in git repositories rather than copying them into the container in order to comply with the FAIR principles. The recipe can also define a set of commands that can be used to reproduce specific results.

5. *Continuous integration of research software*

Continuous Integration (CI) is the practice of frequently integrating changes in a shared version of software. In the context of scientific research, CI can be used to increase the quality of research software without introducing a significant workload for researchers. A CI workflow for research software involves integrating changes from a feature branch (where new ideas are developed) into the main branch (which maintains the current stable version of the software) when results are suitable for submission to a peer-review process. Bug fixes and improvements to the main branch can be integrated into the feature branch on a more frequent basis, such as weekly. The use of CI can ensure that improvements to shared components are integrated into individual feature branches and that the developments achieved in a successful project are integrated by everyone in a research group. The CI process can also include automatic execution of tests, generation of visualizations, and publication of results for inspection. If tests pass, the merge request is accepted and changes are added to the main repository. If tests fail, the developer can inspect the results and re-initiate the process.

Conclusion

The authors argue that the proposed workflow significantly increases the quality of scientific results with a minimal workload overhead, making it attractive for researchers who primarily produce CSE research software as their scientific output. The workflow combines an established build system with a version control branching model adapted to the peer-review process, and places a focus on meeting the FAIR principles (Findability, Accessibility, Interoperability, Reusability) of scientific data. The workflow also includes a version of Test-Driven Development adapted to CSE research software, which gradually increases the test coverage of the research software and helps researchers to reproduce results across different computing platforms.

Author Barreto et al
Title of paper Sentiment analysis in tweets: an assessment study from classical to modern text representation models
URL of the paper:
https://arxiv.org/pdf/2105.14373.pdf
Reviewer: Victor

The paper "Sentiment Analysis in Tweets: An Assessment Study from Classical to Modern Text Representation Models" by Barreto et al. focuses on the task of sentiment analysis in tweets, specifically in the context of mental health conditions. The study evaluates a range of text representation models, including both classical and modern approaches, to determine their effectiveness in analyzing sentiment in tweets related to mental health.

The scope of the study is limited to tweets related to mental health conditions, with a specific focus on depression and anxiety. The authors used a variety of training methods, including supervised, semi-supervised, and unsupervised approaches.

The model architectures evaluated in the study include traditional machine learning models such as Support Vector Machines (SVMs) and Random Forest (RF), as well as more recent models such as the BERT and RoBERTa. The authors used a combination of pre-trained and fine-tuned models on datasets such as the Twitter Sentiment Analysis dataset, the SemEval 2018 dataset, and the Mental Health dataset.

The study also addresses the issue of label noise in the datasets by using a technique called "noise-aware" training, which aims to reduce the impact of noisy labels on the model's performance.

Overall, this paper provides an in-depth assessment of various text representation models for sentiment analysis in tweets related to mental health and their robustness to label noise.

The datasets used for pre-training and fine-tuning include the Twitter Sentiment Analysis dataset, the SemEval 2018 dataset, and the Mental Health dataset.

The authors used the Twitter Sentiment Analysis dataset for pre-training the BERT and RoBERTa models. This dataset contains 1.6 million tweets labeled with positive, neutral, and negative sentiment. The SemEval 2018 dataset was used for fine-tuning the BERT and RoBERTa models, which contains tweets related to mental health and labeled with positive, neutral, and negative sentiment. The Mental Health dataset was used for fine-tuning the SVM and RF models.

Regarding label noise correction, the authors used a technique called "noise-aware" training, which aims to reduce the impact of noisy labels on the model's performance. The technique is based on the use of a noise transition matrix, which estimates the probability of a label being noisy given the true label. The authors propose to use this matrix as a weighting factor in the training loss function to give less weight to the samples with higher probability of being noisy. This way, the model pays less attention to the noise during the training process, leading to a more robust model.