

README of Codes for paper titled “Silent Suffering: Using Machine Learning to Measure CEO Depression”

Sung-Yuan (Mark) Cheng
University of Kentucky
mark.cheng@uky.edu

Nargess M. Golshan
Indiana University
nargolsh@iu.edu

This document provides an overview of the codes used to generate our final sample. Running the Python code files sequentially produces the final datasets. For the purpose of generating log files required by the journal, a small subsample of the primary dataset is used. This approach is necessary because running the code for the full dataset is computationally intensive. The supercomputers at Indiana University and the University of Kentucky were utilized to process the entire data set, the code being submitted as batch jobs that do not generate log files.

The final data sets produced by the Python scripts are subsequently merged in the SAS project file, which serves as the input for the regression analyses conducted in the Stata file. Below is a brief description of each code file:

01_process_daic_audio_files.py

This code pre-processes the audio files to make them ready for the Google YAMNet model. The input to this code is DAIC audio files in WAV, and the output is 7-second audio file segments.

02_yamnet_embed_daic_audio_files.py

This code extracts embeddings from segmented audio files using the Google YAMNet model. The input to this code is segmented audio files in WAV, and the output is embeddings extracted from Yamnet.

03_convert_mp3_to_wav.py

This code converts the MP3 files audio files downloaded from S&P CapitalIQ platform to WAV format.

04_check_wav_file_contains_music.py

This code checks if WAV file contains music or abnormal sound and discards it as that would interfere with the diarization process.

05_diarize_ceo.py

This code extracts the part in the audio file where CEO speaks during the presentation portion of the conference call.

06_procee_ceo_audio.py

This code segments CEO audio files into 7-second parts.

07_yamnet_embed_ceo_audio_files.py

This code extracts embeddings from segmented CEO audio files using the Google YAMNet model. The input to this code is segmented CEO audio files in WAV, and the output is embeddings extracted from Yamnet.

08.01_svm_training_testing.py

This code trains and tests an SVM model using DAIC audio files.

08.02_cnn_training_testing.py

This code trains and tests a CNN model using spectrograms of the DAIC audio files.

08.03_yamnet_training_testing.py

This code trains and tests a depression score prediction model using Yamnet embeddings of the DAIC audio files.

09_predict_ceo_depression.py

This code uses YAMNET embeddings extracted from CEO audio files to generate depression prediction scores. The input to this code is the embedding files, and the output is a summary of the predictions in a CSV file.

10_ceo_speech_properties.py

This code calculates the textual properties of the CEO speech during the Q&A session of the call.

11_calculating_guidance_cessation.py

This code calculates when firms stop issuing guidance.

12_calculating_operating_cf_volatility.py

This code calculates cash flow volatility.

sas_codes_and_log_files.egp

This SAS project file cleans all the datasets necessary for the analyses. The codes in the project need to be executed in the order mentioned by the numbers in their names. The log files are included in the packaged project file.

regressions_final.do

This Stata do-file includes the codes to generate tables in the manuscript.