# Abstract

In this project, I propose a classification method using BERT [1] as sentence embedding and support vector machine (SVM) as classifier to classify the status of stigma from MIMIC-III clinical notes. The ground truth stigma status (1 or 0) is defined whether the key words appear each sentence of clinical notes. After cleaning and preprocessing the raw clinical notes, I use BERT embedding with TensorFlow 2.0, and finally train and validate an SVM classifier with 10-fold cross-validation. The average classification accuracy is almost 100%.

# Introduction

Stigma is discrimination against an identifiable group of people, a place, or a nation [2]. In this project, I try to identify whether stigma exists in clinical notes. In the following section, I will describe in detail the data, label, and processing steps.

# Method

In this section, I will describe the whole workflow shown in Figure 1 in more details

Clinical notes

Preprocessing

Labels

BERT Embedding

SVM Classifier

10-fold cross validation

Figure Whole Workflow

## Data

In this project, I use clinical text from the notes in the MIMIC-III database [3], which is a freely available database comprising deidentified health-related data from patients who were admitted to the critical care units of the Beth Israel Deaconess Medical Center in Boston, Massachusetts. MIMIC-III contains data from 2001-2012. In this demo, I will use MIMIC-III table NOTEEVENTS, which contains all notes for each hospitalization.

Without loss of generality, I only randomly sample 100 subjects in the following analysis.

## Preprocessing

In this step, I will preprocess and prepare the data from NOTEEVENTS table for later usage. Firstly, all notes are spitted using the strategy that break down segments and chip away structure until just prose. Secondly, I use SpaCy to perform sentence extraction, tokenization, customized sentence boundaries setting and parser, with ‘en\_core\_sci\_md’ tokenizer which is designed for biomedical data with a large vocabulary and word vectors. In additional, I convert de-identification text such as admission date (‘[2010-02-14]’) into one token. Finally, I remove extra whitespace characters, stop words and others. The cleaned output text file has one sentence per line, which is desired by the following BERT embedding.

## Auto labeling

I assign each sentence a label (‘1’ represents stigma, and ‘0’ otherwise), depending on whether any of the following key words appears in it. The full list of key words used is listed in Appendix A.

## BERT embedding

In this project, I generate BERT contextualized embedding vectors for each sentence. I start from the original BERT base model, with 12 layers, 768 hidden size, 12 self-attention heads, and in total 110 million parameters. To transfer the text inputs into numeric token ids and arranged in memory before being input to BERT, a matching preprocessing model is provided in TensorFlow Hub. Therefore, the overall embedding model is shown as following.

Input Layer

text

Keras Layer

preprocessing

Keras Layer

BERT encoder

output Layer

Embedding

Figure 2 Flowchart of BERT embedding

Two embedding vectors can be defined from BERT output: sequence ([CLS]) embedding and pooled embedding. In the original paper, the [CLS] separator is used as a representation of the whole sentence and is considered as a contextualized embedding. Besides, the bert\_layer from TensorFlow Hub returns with a pooled output for the represent of the entire input sequence. Those two embedding vectors are very similar. In this project, I will compare and use one of them as embedding vector.

## SVM Classifier

After embedding, I represent each sentence using a vector, which can be interpreted as a point in high-dimensional space. The goal of SVM classifier is to find a hyperplane to separate those points with different labels with maximum margin.

In order to evaluate the performance of embedding and classifier, a 10-fold cross validation is applied. The whole data is split into 10 equal groups. Each time, I train the SVM classifier on 9 groups and test on the left one group. The classification metrics, such as accuracy, sensitivity and specificity are averaged.

# Result and Discussion

In this project, I randomly select 100 subjects from overall 2083180 subjects, as described previously. In total I extract 3151 sentences from the 100 notes. However, only 5 sentences are labeled as ‘1’, which means that those 5 sentences contain at least one of the key words in Appendix A.

The accuracy is almost 99.9%, and the specificity is 99.9%.

# Conclusion

In this project, I use BERT as embedding and SVM as classifier to try to extract stigma status from clinical notes. However, since the label is unbalanced, more careful tuning needs to be done in the future.

# Reference

[1] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805* (2018)

[2] <https://www.cdc.gov/mentalhealth/stress-coping/reduce-stigma/index.html?CDC_AA_refVal=https%3A%2F%2Fwww.cdc.gov%2Fcoronavirus%2F2019-ncov%2Fdaily-life-coping%2Freducing-stigma.html>

[3] Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo An- thony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:160035

# Appendix A.

The full list of key words for stigma labelling:

keywords = ["Addict", "addiction", "user", "drug abuser", "drug seeking", "abuser", "former addict", "reformed addict", "addicted", "use drugs", "drug baby", "opioid abuse", "opioid dependence", "addiction", "want drugs", "problem", "use problem", "habit", "clean", "clean from drugs", "clean urine test", "dirty urine test", "relapse", "opioid substitution", "relapse therapy", "treatment failure", "being clean"]