Time-Aware and Co-Occurrence-Aware Network for Medical Prediction Reproducibility

Yumei Wang and Joe Laniado
12/1/2023

Outline

I. Introduction
II. Scope of reproducibility
III. Methodology
IV. Results
V. Discussion

Introduction

The hierarchical structure of EHRs

Each patient has a sequence of admissions

Time-aware methods

Data need to be presented in a sequential manner

Each admission has unordered co-occurrence diagnoses

Co-occurrence-aware modeling

Data need to be in the form of a bipartite graph

Current issues

!!! No approach uses both time and co-occurrence relations for modeling

!!! The Deep Learning (DL) model is difficult to interpret for non-specialists due to its black box nature

The paper

1. Proposed an interpretable Time-aware and Co-occurrence-aware Network (TCoN) to model both relations
2. Performed four different predictive tasks: mortality prediction, readmission prediction, disease prediction, and next diagnoses prediction

Scope of reproducibility

- Recreate the proposed network
- Evaluate on four binary classification tasks proposed in the paper: mortality prediction, readmission prediction, heart failure prediction, and sepsis prediction

Methodology

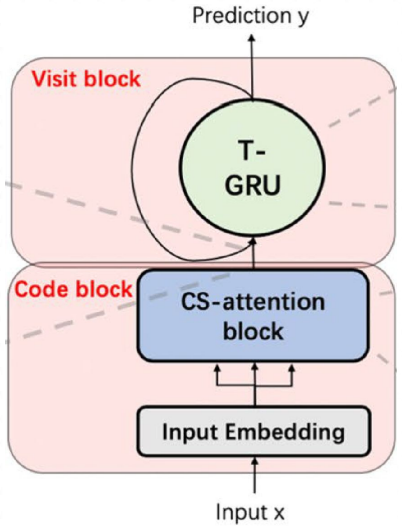
- Model description
- Dataset description
- Computational implementation
- Code

Methodology

- Model description

➢ Model architecture:

Layers: an initial embedding layer, a multi-head self attention and finally a time-aware GRU network



Same architecture, but a simplified version of the TCoN network

Methodology

- Model description

➢ Model architecture:

Activation function: ReLu for embedding layer

Training objectives: cross entropy loss as our loss function, using Adam Optimizer with a lr = 0.001, β1 = 0.9, and β2 = 0.999 (same as the paper)

Methodology

- Dataset description

➢ Source of the data: the MIMIC-III clinical database (https://physionet.org/content/mimiciii/1.4/)

➢ Statistics of the extracted MIMIC-III datasets

Avg. rate of in-hospital mortality	12.49% (5809/46520)
Avg. rate of readmission to ICU	16.20% (7537/46520)
Avg. rate of sepsis	7.22% (1444/19993)
Avg. rate of heart failure	16.86% (3370/19993)

Georgia Tech

Methodology

- Dataset description

❖ Mortality dataset: used the DEATHTIME variable in the admission table to create mortality label. If DEATHTIME was null then our mortality label would be a 0, otherwise 1.

❖ Readmission dataset: if the patient has more than one visit, it would be classified as a 1, otherwise 0.

Methodology

- Dataset description

For disease prediction, the observation window was set as before the diagnosis of the disease.

❖ Sepsis dataset: Following the latest sepsis 3.0 definition (SOFA >=2), ICD9 Code for this disease is 995.92. Sepsis label is 1 if the patient presented sepsis and a 0 if they didn't.

❖ Heart failure dataset: ICD9 code for heart failure started with 428.x. Heart failure label is 1 if the patient had heart failure and a 0 if they didn't.

Methodology

- Dataset description

❖ All four datasets were split into train, validation and test sets in the ratio of 0.75 : 0.1 : 0.15 respectively.

Methodology

- Computational implementation

Software and hardware:

- ❑ Data preprocessing was done using python on Google Colab.
- ❑ PyTorch was used for model implementation.
- ❑ Local CPU: 2.4 GHz Dual-Core Intel Core i5

Methodology

- Computational implementation

Hyper-parameters initialized in experiments:

- Learning rate: 0.001
- Dropout rate: 0
- Number of iterations: 10

Methodology

- Code

➢ Data preprocessing and model implementation codes were both developed by ourselves.

Repo link: https://github.com/joelaniado/CS6250_Project/tree/main

Results

Table 1. Results of original and our binary classification.

Method	Mortality		Readmission		Sepsis		Heart Failure	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
GRU	0.7902	0.7400	0.7023	0.6713	0.6202	0.6063	0.6525	0.6187
Med2Vec	0.8025	0.7950	0.7125	0.6833	0.8211	0.7943	0.7225	0.7101
Dipole	0.8133	0.8103	0.7341	0.7243	0.8001	0.7823	0.7067	0.6923
TLSTM	0.7893	0.7392	0.7256	0.7023	0.6432	0.6189	0.7432	0.6033
TCoN	0.8224	0.8134	0.7403	0.7278	0.8433	0.8233	0.7698	0.7313
TCoN BDH	0.7605	0.4737	0.6955	0.2932	0.9506	0.8758	0.8321	0.7747

Our network performed better than the original TCoN when predicting disease, but worse for mortality or readmission prediction.

Possible reasons: variation on the preprocessing, simplification of the network itself, or specific hyperparameters when building and training the network.

Georgia Tech

Results

Hyper-parameters tuning: learning rate and dropout rate

Table 2. Results of Hyper-parameter Tuning on TCoN BDH

Method	Mortality		Readmission		Sepsis		Heart Failure	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
Lr = 0.001, Drp = 0	0.7605	0.4737	0.6955	0.2932	0.9506	0.8758	0.8321	0.7747
Lr = 0.005, Drp = 0	0.5224	0.2158						
Lr = 0.0005, Drp = 0	0.7791	0.5162	0.6982	0.2909	0.9696	0.9105	0.8442	0.7805
Lr = 0.0005, Drp = 0.1	0.8081	0.5438	0.6987	0.2870	0.9748	0.9132	0.8481	0.8227
Lr = 0.0005, Drp = 0.15	0.8077	0.5384	0.6978	0.2879	0.9764	0.9161	0.8465	0.7880
Lr = 0.0002 , Drp = 0.1	0.8094	0.5203	0.6982	0.2878	0.9765	0.9096	0.8365	0.7659

Dropout rate of 0.1 and a smaller learning rate (0.0005) proved to achieve the best result.

Georgia Tech

Results

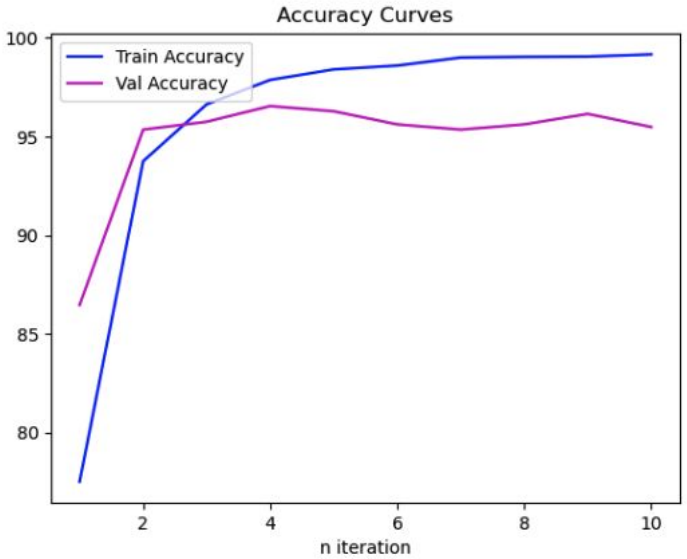


Figure 7. Accuracy curve

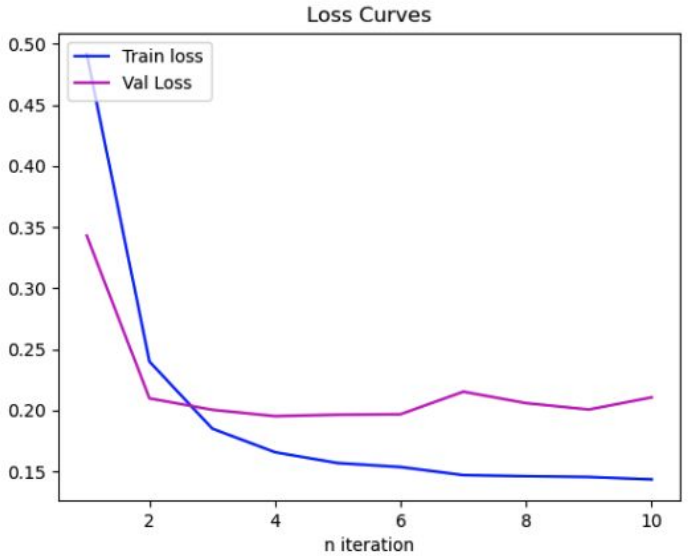


Figure 8. Loss Curve

Discussion

- This paper could be reproduced but it requires a deeper understanding of deep learning concepts and processes to truly grasp their methodology and manipulate the different weights, biases, and data to achieve a similar result with a proper interpretation path.
- The data preprocessing part was relatively easy to reproduce.
- The original paper built every cell of the network from scratch calculating the outputs by multiplying the weights and biases with the data using TensorFlow, which was difficult to reproduce.

Discussion

The paper itself provides a great tool for medical professionals that could be adapted to many predictive modeling tasks as well as provides deeper insight on personalized healthcare for each patient, but for their work to be easily reproduced, further documentation is needed on network specific configurations, data preprocessing, and the overall code implementation of their approach.

Georgia Tech