Reproducing Context-aware Health Event Prediction via Transition Functions on Dynamic Disease Graphs (?) CS598 DL4H Spring 2023

Shiyu (Sherry) Li and Wei-Lun (Will) Tsai
{shiyuli2, wltsai2}@illinois.edu

Group ID: 53
Paper ID: 28
Presentation link: #TODO
Code link: https://github.com/willtsai/dlh-sp23-team53

1 Introduction

In this report, we will focus on our reproduction study for *Context-aware Health Event Prediction via Transition Functions on Dynamic Disease Graphs* (**?**). This paper propose a new deep learning model called Chet (**c**ontext-aware **h**ealth **e**vent prediction via **t**ransition functions on dynamic disease graphs) that leverages the relationship between diseases and how they develop over time to predict future outcomes and diagnoses. Existing research on deep learning models for classification and prediction of diseases based on longitudinal EHR data have modeled disease diagnoses as independent events in their respective visits. However, intuition and data indicate that there are in fact hidden patterns within the combinations of disease diagnoses that may be useful for predicting future outcomes for patients, but yet have not been leveraged in existing best-in-class healthcare deep learning models. The Chet model is able to learn how diagnosed diseases develop over the course of each patient's doctor visits and then utilize this learned disease combination context to predict future outcomes and diagnoses. The most innovative part of the approach is the design to include both disease combinational information and the dynamic scheme of disease into the model. To include disease combinational information, the paper constructed a weighted disease combination based on the entire longitudinal EHR data globally and also a disease subgraph based on the specific visit locally. To include dynamic scheme of diseases, the paper utilized a disease-level temporal learning with multiple diagnosis roles and corresponding transition functions to extract historical contexts.

2 Scope of reproducibility

In our reproduction study, we will use the same methodology as proposed by the authors for data selection, cleaning, and preprocessing. Specifically, we will join the MIMIC-III (**?**) and MIMIC-IV (**?**) datasets along the same overlapping time ranges that the authors describe and split training/validation/test sets randomly using the same ratios they used. We will build the diagnosis graphs and calculate the adjacency matrices for their corresponding subgraphs using the same methodology described by the authors. We will train the model and at least one listed baseline model for diagnosis prediction and hearth failure prediction respectively and compare the performance.

2.1 Addressed claims from the original paper

- By utilizing disease combinational information and the dynamic scheme of diseases, the Chet model has higher accuracy for diagnosis prediction and hearth failure prediction than RETAIN model(one of the baseline models).

- The designed global disease graph and visit subgraphs can integrate global and local context from disease combinations to inform the deep learning model, so the Chet model has higher accuracy for diagnosis prediction and hearth failure prediction than $Chet_d-$ model where dynamic part of GNN is removed in Chet.

- The proposed three diagnosis roles and corresponding transition functions can extract historical context and learn the disease development schemes, so the Chet model has higher accuracy for diagnosis prediction and heart failure prediction than $Chet_t-$ model where transaction functions are removed in Chet.

- The proposed three diagnosis roles and corresponding transition functions can extract historical context and learn the disease development schemes, so the Chet model has higher accuracy for diagnosis prediction and heart failure prediction than $Chet_{att}-$ model where attention from the final patient embedding layer is removed in Chet. This is an additional ablation study we proposed.

3 Methodology

In this section , we demonstrate the details of the model used in the original paper, our implementation approach as well as necessary computational resource.

3.1 Model descriptions

The Chet model can be decomposed into three layers: graph layer, transition layer and embedding layer.

3.1.1 Graph Layer

The first layer is a dynamic graph learning layer to extract both local and global contexts for diagnosis and neighbors in visit t using a memory-efficient calculation:

$$Z_D^t = m^t \odot (M + A(m^t \odot M) + A(n^t \odot N)) \quad (1)$$

$$Z_N^t = n^t \odot (N + A(n^t \odot N) + A(m^t \odot M)) \quad (2)$$

Where M,N represent embedding matrices for diagnoses and neighbors, A is the static adjacency matrix, m^t and n^t represent diagnoses code and neighbors code in t visit, Z_D^t is aggregated diagnosis local context and diagnosis global context and Z_N^t is aggregated neighbor global context. Finally, the GNN outputs are calculated with a fully connected layer with LeakyReLU as the activation function from Z_D^t and Z_N^t.

$$H_{D,N}^t = \text{LeakyReLU}(Z_{D,N}^t W) \in \mathcal{R}^{d \times s'} \quad (3)$$

3.1.2 Transition Layer

The Transition(second) Layer is to learn the disease development schemes, it takes the vector of diagnosis codes m^t per visit as input and partitions it into three disjoint vectors: (1) persistent diseases m_p^t representing diagnoses in visit t that are also present in visit $t-1$, (2) emerging neighbors m_{en}^t representing diagnoses in visit t that are neighbors in visit $t-1$, (3) emerging unrelated diseases m_{eu}^t representing diagnoses in visit t that are unrelated diseases in visit $t-1$. The layer is composed of three transition functions corresponding to each partition of m^t and are designed to extract historical context from previous visits to compute the hidden values. The transition function for calculating the hidden values for both m_{en}^t and m_{eu}^t is a scaled dot-product attention (**?**):

$$\begin{matrix}\text{Attn} \\ \text{(Q, K, V)}\end{matrix} = \begin{matrix}\text{soft} \\ \text{max}\end{matrix} \left(\frac{QW_q(KW_k)^T}{\sqrt{a}} \right) VW_v \quad (4)$$

Where a is the attention size, W_q, W_k, W_v are the weight matrices. For h_{en}^t, Q and K are the hidden neighbor embeddings H_N^{t-1}. For h_{eu}^t, Q and K are the universal embeddings of unrelated diseases R. For both h_{en}^t and h_{eu}^t, V is the diagnosis embeddings H_D^t. The transition function for calculating the hidden values for m_p^t is a modified gated recurrent unit (M-GRU) (**?**):

$$h_p^t = M\text{-}GRU(m_p^t \odot H_D^t, h_{en}^t, h_{eu}^t, h_p^{t-1}) \quad (5)$$

Finally, to calculate the visit embedding v^t, we apply max pooling to the transition output of the three partitions, which are all contained in h_p^t:

$$v^t = \texttt{max\_pooling}(h_p^t). \quad (6)$$

3.1.3 Embedding Layer

The third layer is an embedding layer with a location-based attention to calculate the final hidden representation of all visits embeddings.

$$\alpha = \text{softmax}([v^1, v^2, \ldots, v^T]W_\alpha) \in \mathcal{R}^T \quad (7)$$

$$o = \alpha[v^1, v^2, \ldots, v^T]^T \in \mathcal{R}^p \quad (8)$$

Where W_α is a context vector for attention, α is the attention score for visits and o represents the final patient embedding.

3.2 Data descriptions

For their reproduction study, we will use the MIMIC-III (**?**) and MIMIC-IV (**?**) datasets downloaded from PhysioNet(**?**). for training/validation/testing, same as the original paper. In MIMIC-III data, there are 7493 patients in total from 2001 to 2012 with an average of 2.6 visits per patient and an average of 13.06 diagnose codes per visit. We randomly split the data into training set, validation set and test set with a size of 6000, 493 and 1000 respectively. In MIMIC-IV data, there are 10000 patients in total from 2013 to 2019 with an average of 3.79 visits per patient and an average of 13.51 diagnose codes per visit. We split the data into training set, validation set and test set with a size of 8000, 1000 and 1000 respectively.

3.3 Hyperparameters

# TODO: Describe how you set the hyperparameters and what the source was for their value (e.g. paper, code or your guess).

3.4 Implementation

In our preliminary reproduction implementation, we built a python notebook to build a complete flow including hyperparameters setting, data preprocessing, data loading, model building, model training and evaluation. For now, we kept the same hyperparameters as the original paper and reused the author's code in data preprocessing. Our major code efforts went into model rebuilding and the training/validation flow. In training/validation part, we built our own training and validation method to streamline the training, validation and test process while reusing the existing schedulers and metrics. In model rebuilding part, we tried to follow closely with the model structure and all the equations in the paper step by step.

3.5 Computational requirements

All code are implemented with Python and PyTorch. For additional package and version details, please refer to requirements.txt. In our initial investigation, it took around 10 minutes for data preprocessing and approximately 96 hours to complete total 200 epoches of training for a combined MIMIC-III/MIMIC-IV training set on our local machine with 16GB memory and Apple M1 PRO chip. In our reproduction implementation, it took 10 hours to actual finish 10 epoches of training for a combined MIMIC-III/MIMIC-IV training set on the same machine to get an initial result. In order to unblock the computational constraints we have with the CPUs on our local machine, we would like to further explore the following computational resources:

- We will try to use the available GPUs(16 cores) on one of our local machines for the model training.
- We will explore with Google Colab to utilize their free standard GPUs(NVIDIA T4 Tensor Core GPUs) for the model training.
- We will explore with Microsoft Azure for available Virtual Machines with GPU using the available credit we have.

4 Results

We evaluated prediction performance for our models against the test datasets. Our preliminary results are from training each of our reproduced Chet models one time with 10 epochs instead of the 200 epochs originally used in the paper. From our more limited runs, we have found that our Chet model performance results align more closely with results from the original paper for the heart failure prediction task than for the diagnosis prediction task. Additionally, we also make the same observation as the original paper that MIMIC-IV trained models outperform MIMIC-III trained models in terms of AUC and F1 score. Since we have not yet reproduced results from any of the baseline models, we are not in a position to determine whether our work supports the claims from the original paper that Chet outperforms all the baseline models. The plan for finalizing our experiment results is as follows:

- Retrain the reproduced Chet model for *diagnosis* prediction using as close to the original 200 epochs as we are computationally able to, then collect, analyze, and compare the results.
- Retrain the reproduced Chet model for *heart failure* prediction using as close to the original 200 epochs as we are computationally able to, then collect, analyze, and compare the results.
- Reproduce at least one of the baseline models (likely RETAIN) to compare reproduced Chet prediction results against those from a reproduced baseline model.
- Investigate the discrepancy in the number of parameters between our reproduced Chet models and the original Chet models, making any necessary changes to our code as needed.
- Implement at least one of the ablation studies from the original paper - collect, analyze, and compare the results.
- Implement an additional ablation study of our own - collect, analyze, and compare the results.

4.1 Diagnosis Prediction Results

Model performance results for the diagnosis prediction task from our preliminary experiments are summarized in Table **??**. Compared to the 200-epoch Chet model performance results from the original paper, our reproduced 10-epoch Chet model performed only approximately half as well in terms of F1 score and approximately 73% as well in terms of R@10 and R@20, across both MIMIC-III and MIMIC-IV datasets. We suspect that this large discrepancy in performance is due to having trained our reproduced Chet models for only 10 epochs instead of the 200 epochs used in the original paper. Interestingly, for the Chet model, the number of parameters in our experiments were approximately 58% of the number of parameters in the original experiment despite using the same hyperparameters and datasets.

4.2 Heart Failure Prediction Results

Model performance results for the heart failure prediction task from our preliminary experiments are summarized in Table **??**. Compared to the 200-epoch Chet model performance results from the original paper, our reproduced 10-epoch Chet model achieved similar levels of performance in terms of AUC and F1 score, across both MIMIC-III and MIMIC-IV datasets. This is suprising given that we trained on 95% less epochs than the original paper, but we suspect that the decent prediction performance can be attributed to the fact that the heart failure prediction task is a much easier task than the diagnosis prediction task. Intuitively, predicting a general diagnosis is much more ambiguous and involves more complexity than predicting a single specific condition such as heart failure. Indeed, we observe in the original paper that the prediction performance for heart failure is much better than that of diagnosis prediction, across all baseline models and Chet, for both MIMIC-III and MIMIC-IV datasets. Similar to our diagnosis prediction Chet models, we observe that the number of parameters in our experiments were approximately 68% of the number of parameters in the original experiment despite using the same hyperparameters and datasets. We would have expected the space complexity of our Chet models to be the same of the original Chet models. We think that this difference might be due to how we are computing data embeddings in the model layers and plan to investigate this further to hopefully resolve this discrepancy.

4.3 Additional results not present in the original paper

We plan to complete at least one ablation study from the original paper as well as an additional ablation experiment of removing the attention from the final patient embedding layer (not part of the original paper). We will include our results for these additional experiments in the final report.

5 Discussion

# TODO:

5.1 What was easy

# TODO:

5.2 What was difficult

# TODO:

5.3 Recommendations for reproducibility

# TODO:

6 Communication with original authors

# TODO:

Model	MIMIC-III				MIMIC-IV			
	w-F1	R@10	R@20	# Params	w-F1	R@10	R@20	# Params
Repro Baseline	#TODO	#TODO	#TODO	#TODO	#TODO	#TODO	#TODO	#TODO
Repro Chet	12.89	20.19	28.96	1.22M	14.21	21.64	30.32	1.49MM
Original Chet	22.63	28.64	37.87	2.12M	26.35	30.28	38.69	2.59M

Table 1: Diagnosis prediction results on MIMIC-III and MIMIC-IV using w-F1 (%) and R@k (%).

Model	MIMIC-III			MIMIC-IV		
	AUC	F1	# Params	AUC	F1	# Params
Repro Baseline Models	#TODO	#TODO	#TODO	#TODO	#TODO	#TODO
Repro Chet	84.16	69.32	0.47M	94.00	75.14	0.58M
Original Chet	86.14	73.08	0.68M	90.83	71.14	0.88M

Table 2: Heart failure prediction results on MIMIC-III and MIMIC-IV using AUC (%) and F1 (%).