Dynamic Temporal Graph Neural Network Framework for Predicting Heart Disease Progression Mid Stage Report

Jake Ciocca, Sarthak Adhangale
Advisor: Yu-Dong Yao
AAI-800
Stevens Institute of Technology
Fall 2025

“We pledge to adhere to the Stevens Graduate Student Code of Academic Integrity, and we have discussed the proposal with our Project Advisor” - Jake Ciocca, Sarthak Adhangale

Dynamic Temporal Graph Graph Neural Network Framework for Predicting Heart Disease

Jake Ciocca

Department of Electrical and Computer Engineering

Stevens Institute of Technology

Hoboken, NJ, USA

jciocca@stevens.edu

Sarthak Adhangale

Department of Electrical and Computer Engineering

Stevens Institute of Technology

Hoboken, NJ, USA

sadhanaga@stevens.edu

Yu-Dong Yao

Department of Electrical and Computer Engineering

Stevens Institute of Technology

Hoboken, NJ, USA

yyao@stevens.edu

***Abstract*— For the longest time, heart disease has remained one of, if not, the leading causes of mortality worldwide. Newly leveraged methods using machine learning often fail to capture the complex relationships among patient features that are displayed in medical datasets. This paper presents a Graph Neural Network (GNN) approach for the prediction and detection of heart disease by modeling patients and their clinical similarities as graph structures to capture relational dependencies. Preliminary results indicate that GNNs offer a step in a promising direction for developing interpretable and robust tools in medical diagnostics.**

***Keywords*— Graph Neural Networks, Heart Disease Prediction, Machine Learning, Electronic Health Records, Medical AI**

I. INTRODUCTION

The most prominent and dominant leading cause of morbidity and mortality worldwide is heart disease which places a significant burden on the healthcare system. The most crucial events when it comes to reducing risk of death and improving patient outcomes are early detection and accurate prediction of these cardiac events. Some of the events are, but not limited to, coronary artery disease, heart failure, and arrhythmias. Typical events that currently help reduce the mortality rate of heart disease include timely intervention, monitoring, and personalized treatment. Traditional diagnostic approaches, as of right now, rely on clinical risk factors, such as age, hypertension, and cholesterol, among others, as well as imaging, and invasive procedures. The dilemma with these procedures is that they may fail to capture underlying complex interactions among the multiple variables that indicate heart disease. Often, these procedures produce limited predictive accuracy in heterogeneous patient populations.

In recent years, similar to almost every other field, machine learning and deep learning techniques have been adopted and applied to cardiovascular risk prediction and classification tasks. The common models being highly used today, such as logistic regression, random forests, convolutional neural networks and recurrent neural networks have shown promising results in identifying heart disease risk factors. With that being said, there still remains a problem as these tend to treat patient features as independent variables essentially ignoring the underlying dependencies and correlations between them. This current limitation can lead to suboptimal predictions, most specifically when relationships among features or patients play a significant role in disease manifestation.

Graph Neural Networks, also known as GNNs, provide a novel framework for learning from graph structured, or connected, data. This in turn enables the ability to capture complex dependencies through message passing mechanisms that aggregate information from neighboring roles [1], [2]. By utilizing these graph representations, GNNs can model patient to patient similarities or, in terms of deep learning, feature to feature correlations, which provide richer relational context for classification tasks. As of recent, some studies have shown the ability of GNNs to learn complex patient relationships and risk trajectories when being applied to large, diverse datasets allowing us to see the promising results in healthcare applications beyond cardiovascular disease [6], [7]. This approach aligns with principles of geometric deep learning, extending upon traditional neural networks to non-Euclidean data domains [2]. Recent studies in the area have shown the potential of GNNs in healthcare applications such as heart failure prediction from electronic health records [3] as well as ejection fraction estimation from echocardiography data [4], insinuating the growing use of relational learning in medical artificial intelligence.

In spite of this promise, applying GNNs to heart disease detection and prediction presents some key challenges. These challenges include constructing suitable graph representations from heterogeneous medical data, addressing issues of class imbalance and data sparsity, and, lastly, ensuring that the resulting models are interpretable and clinically valid. Furthermore, it is essential to show whether GNN based models can achieve superior predictive performance in comparison to conventional machine learning and deep learning methods.

The main contributions of this projects are as follows:

1. We propose a GNN based framework for heart disease detection and prediction where patients or features are represented as graph nodes and their relationships as edges.
2. Relational modeling through GNNs can enhance predictive performance is explained by capturing the structural dependencies among clinical variables.
3. The proposed GNN approach is compared against the conventional models to evaluate performance gains.
4. The challenges and potential for integrating graph based learning into clinical decision support systems is discussed.

The remainder of this research paper is organized as follows. Section II presents the formal definition of the research problem being addressed and the system model. Lastly, Section III discusses the proposed methodology, including model architecture, graph construction, and preliminary results.

II. PROBLEM DESCRIPTION

The main research problem addressed in this study is defined as follows:

How can a Graph Neural Network (GNN) be effectively designed to detect and predict heart disease by modelling relational dependencies among clinical features and patients, consequently improving predictive performance and interpretability to conventional approaches?

Specifically, we consider a dataset containing multiple patients, each described by clinical attributes including age, blood pressure, cholesterol levels, and echocardiogram (ECG) readings. Conventional models treat these attributes as independent features where we represent the dataset as a graph $G = (V, E)$, where each node $v \in V$ corresponds to a patient, or a clinical feature, and each edge $e \in E$ represents a relationship of substance defined as patient similarity based on feature distance or correlation between clinical variables. Each node has an associated feature vector x_v, and the adjacency matrix A encodes the graph's structure. Alternatively, prior models using this structure have tended to rely on just structured tabular features and temporal sequences which lack the ability to model heterogeneous relationships between patients, biomarkers, and diagnostic events [8].

A GNN model learns node representation by accumulating information from neighboring nodes over several layers, in turn, producing embeddings that include relational context. The model $f(G, X; \Theta)$ is parameterized by weights Θ and outputs a prediction $y \in \{0, 1\}$, where $y = 1$ indicates the presence of heart disease. Differing from past graph convolutional architectures [5], which assume static graph structures, the approach we employ utilizes temporal graph updates in effort to dynamically encode evolving patient states. The main goal is to minimize a loss function (e.g. binary cross entropy) while maximizing classification metrics such as accuracy, F1-score, and the area under the ROC curve (AUC).

The system assumes access to labeled data with known heart disease outcomes for supervised training. Constraints include potential class imbalance, meaning fewer positive cases, small dataset size, and the need for model interpretability to ensure clinical trust. By addressing these challenges, this research aims to demonstrate that relational modeling through GNNs provides measurable benefits for heart disease prediction and could lay the groundwork for more advanced, explainable AI tools in healthcare.

I. PRELIMINARY IDEAS AND RESULTS

4. *Overview of the Approach*

To begin, the objective of this project is to develop a Graph Neural Network, also known as a GNN, based temporal learning framework for heart disease and prediction using patient specific data. The idea behind this is that common deep learning approaches in this field like CNNs or LSTMs struggle to capture the complex relational dependencies that exist between patients, biomarkers, and medical conditions. Heart disease progression in itself is not an isolated phenomenon as it is influenced by interrelated physiological parameters and historical patient data that naturally form a dynamic graph structure.

This led us to our approach as we decided to leverage a Temporal Graph Network, also known as a TGN, implemented in PyTorch Geometric, designed to model both the temporal evolution that comes with patient states and their relational dependencies. Extending upon static GNNs, TGNs introduce memory modules that track historical information for each node which represent a patient or biological entity and dynamically update node embeddings as new events occur over time. By incorporating both the temporal and structural insights, our systematic approach aims to achieve more accurate predictions of heart disease outcomes compared to static models.

In our work, the core novelty lies in applying this temporal graph based framework to healthcare data where the node attributes evolve over time. For example, as someone ages or from one checkup to the next, there may be changes in cholesterol levels, blood pressure and, or, heart rate. The edges within the graph network represent clinically meaningful relationships like similarities in medical history or genetic predispositions. By utilizing label propagation across temporal nodes, we further ensure that the model can learn from evolving patient conditions which leads us to more stable and generalizable predictions.

B. Dataset Description and Preprocessing

For this project, the dataset used consists of three key components:

1. features.csv: time stamped physiological and demographic features for each patient node;
2. edges.csv: temporal interactions between patients such as correlations or shared risk factors; and
3. labels.csv: specifying whether or not a patient eventually developed a heart disease condition during the observed period.

Each record in the features file corresponds to a unique pair being the patient and timestamp alongside features such as cholesterol level, blood pressure, ECG results, BMI, and glucose readings. The edge file being used models the relationships as directed temporal edges between nodes which allows the GNN to capture information propagation across similar or connected patients. Lastly, the label file provides supervision as it has labels propagated across timestamps using a label propagation method that ensures that every time slice of a node has an associated outcome label.

Before model training, we applied several preprocessing steps which are as follows:

- Node Mapping: Each unique node_id is mapped to a continuous integer index allowing for efficient graph representation.
- Feature Normalization: All numeric attributes are standardized in effort to improve model convergence.
- Timestamp Encoding: Timestamps are turned into 64-bit integers for temporal alignment.
- Label Propagation: Each node's most recent label is used across all historical timestamps in order to keep consistent supervision.

This structured and systematic approach results in a TemporalData object that is compatible with PyTorch Geometric's TemporalDataLoader, allowing efficient mini batch temporal training.

Model Design and Architecture

The architecture of the model is based on the Temporal Graph Network framework by utilizing TGNMemory, IdentityMessage, and LastAggregator modules. Each and every patient node maintains a memory state that evolves over time as new edges are processed. Edge events trigger memory updates which allow the network to retain important temporal dependencies in its node representations.

Mathematically, the TGN can be summarized as follows:

$$h_v(t) = f_\theta(h_v(t^-), Agg\{m_{u\to v}(t)\}) \quad (1)$$

where $h_v(t)$ denotes the hidden state of node v at time $t, h_v(t^-)$ is its memory before time t, and $Agg\{\cdot\}$ represents an aggregation of messages $m_{u\to v}(t)$ from neighboring nodes. The function f_θ is a learnable memory update module parameterized by θ.

Within our implementation, messages get initialized as zero tensors aligned to each event's feature dimension all while the LastAggregator module helps ensure the most recent information from a node's neighbors is emphasized. This choice in design helps prevent over smoothing of the data and zooms in on learning recent and clinically relevant interactions.

The TGNClassifier module is then used to apply a temporal embedding layer followed by a classification head to predict the likelihood of heart disease occurrence for each of the patient nodes. To train the model, we use cross-entropy loss which is then optimized by optimizer Adam with a learning rate of 1×10^{-3}. In order to stabilize the training, we apply memory detachment after each mini-batch helping us prevent gradients from propagating through the entire temporal history which is a key strategy we use to ensure both computational efficiency and numerical stability.

D. Training Procedure and Implementation Details

The temporal graph data is then divided into train, validation and test sets based on distinct edge timestamps in order to address chronological consistency between training and evaluation phases. This systematic split prevents data from being leaked across time and better reflects real world forecasting scenarios.

The run_epoch() function executes the training and evaluation loops. In each batch, the following happens:

1. Transfers data to the cloud GPU.
2. Selects the events that match the current training phase using timestamp marks.
3. Generates node message tensors that are aligned with feature representations.
4. Computes node-level predictions by using the TGNClassifier module.
5. Evaluates the loss and accuracy then updates model parameters when in training mode.
6. Detaches the memory module to reset the graph history between each batch.

Over ten training epochs, we saw stable convergence and improved validation accuracy across each epoch. The separation of training, validation, and test edge times help the model to be able to generalize effectively when it comes to unseen temporal patterns.

One of the key observations in this phase was that memory detachment significantly enhanced the stability in training by preventing long range gradient accumulation across time. This technique reduced the risk of exploding gradients and made TGN training feasible on large temporal graphs. Moreover, the label propagation mechanism helped improve label density across the temporal graph in turn allowing better supervision and faster convergence.

E. Preliminary Results and Discussions

After this preliminary round of results, we can tell that the model demonstrates promising performance as it achieved consistent improvements in both training and validation accuracy across each epoch. In this round, we could see validation accuracy exceeded seventy percent which suggests that the temporal graph learning effectively captures evolving patient dynamics. Also, it was observed that the test accuracy remained stable which indicates good generalization to unseen temporal windows.

These results emphasize the potential of temporal graph networks in medical prediction tasks. Different from traditional static GNNs or temporal LSTMs, the TGN framework updates its internal memory dynamically with each new event allowing for preserved long term dependencies while underscoring recent changes in patient condition. This facet is particularly valuable for heart disease prediction since subtle temporal patterns, which can be defined as gradual changes in ECG readings or cholesterol levels, carry significant predictive value.

From this point on, we expect our future work to include the integration of clinical text embeddings from patient notes as well as multi-modal medical data to further enrich the graph representation in the model. Following experiments will compare our TGN based approach with the likes of Graph Attention Networks and Temporal Convolutional Networks as a way to quantify the advantages of the proposed framework in both accuracy and interpretability.

Planned Evaluations Metrics and Future Work

In order to comprehensively assess the performance of our proposed Temporal Graph Network model, we plan to utilize a range of classification evaluation metrics that are typically used in medical prediction tasks. While the metric we currently use, accuracy, provides a general indication of performance, it can still end up being misleading when used with imbalanced datasets where the number of healthy patients significantly exceeds those diagnosed with heart disease. Ergo, a more detailed evaluation using metrics like precision, recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve, also known as AUC-ROC, will be conducted.

This is due to the fact the metrics mentioned above provide a better understanding of model performance. Precision is able to measure the proportion of true positive predictions among all positive predictions made by the model effectively reflecting how the model may avoid false alarms. Recall, also known as sensitivity, quantifies the proportion of actual positive cases correctly identified which is specifically very important in healthcare applications as missed detections can have very serious, and potentially fatal, consequences. F1-score represents the harmonic mean of precision and recall as it provides a balanced indicator of model robustness. Lastly, AUC-ROC is great at capturing the trade-off between true positive and false positive rates across thresholds since it offers a holistic view of model discriminative capability.

These evaluation metrics will be calculated across both the validation and test sets in effort to ensure generalization. Also, we plan to analyze temporal performance trends, such as how accuracy evolves across different time windows, to understand whether the model's predictive ability improves as more patient history becomes available. Thus, an analysis like this, will help identify whether or not the TGN memory mechanism effectively captures the indication that long term dependencies or if adjustments to memory retention strategies are needed.

In the next phase of experimentation, more ablation studies will be performed in effort to evaluate the contribution of each component in the model pipeline. This will include analyzing the effects of as follows:

- Removing memory updates by testing static GNN performance;
- Using alternative aggregators such as mean or attention-based instead of the LastAggregator module;
- Comparing performance with non-graph temporal models using LSTMs and Temporal Convolutional Networks.

These analyses will help us validate the unique advantages of temporal message passing in model disease progression and confirm the interpretability of learned embeddings. Ultimately, the end goal is to establish a clinically reliable and explainable framework capable of supporting early diagnosis and personalized treatment recommendations in cardiac healthcare as a whole.

REFERENCES

[1] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, “A Comprehensive Survey on Graph Neural Networks,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, 2021.

[2] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric Deep Learning: Going Beyond Euclidean Data,” *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[3] H. O. Boll, A. Amirahmadi, A. Soliman, S. Byttner, and M. Recamonde-Mendoza, “Graph Neural Networks for Heart Failure Prediction on an EHR-Based Patient Similarity Graph,” *arXiv preprint* arXiv:2411.19742, 2024.

[4] M. Mokhtari, T. Tsang, P. Abolmaesumi, and R. Liao, “EchoGNN: Explainable Ejection Fraction Estimation with Graph Neural Networks,” *arXiv preprint* arXiv:2208.14003, 2022.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[6] H. Lu and S. Uddin, “A Weighted Patient Network-Based Framework for Predicting Chronic Diseases Using Graph Neural Networks,” *Scientific Reports*, vol. 11, no. 1, p. 22607, 2021.

[7] Y. Zhu, J. Li, and K. Wang, “Graph Neural Networks for Clinical Risk Prediction Based on Electronic Health Records,” *Journal of Biomedical Informatics*, vol. 149, p. 104666, 2024.

[8] M. Kim, H. Park, D. Lee, and J. Lee, “Interpretable Temporal Graph Neural Network for Prognostic Prediction of Longitudinal Neuroimaging Data,” *Frontiers in Neuroscience*, vol. 15, p. 835416, 2021.