Representing Outcome-driven Higher-order Dependencies in Graphs of Disease Trajectories

Steven J. Krieg†, Nitesh V. Chawla†*, and Keith Feldman‡
†Lucy Family Institute for Data and Society, University of Notre Dame, Notre Dame, IN 46556, USA
{skrieg,nchawla}@nd.edu
‡Children's Mercy, University of Missouri-Kansas City School of Medicine, Kansas City, MO 64108, USA
kfeldman@cmh.edu

2024 IEEE 12th International Conference on Healthcare Informatics (ICHI) | 979-8-3503-8373-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICHI61247.2024.00010

***Abstract*—The widespread application of machine learning techniques to biomedical data has produced many new insights into disease progression and improving clinical care. Inspired by the flexibility and interpretability of graphs (networks), as well as the potency of sequence models like transformers and higher-order networks (HONs), we propose a method that identifies combinations of risk factors for a given outcome and accurately encodes these higher-order relationships in a graph. Using historical data from 913,475 type 2 diabetes (T2D) patients, we found that, compared to other approaches, the proposed networks encode significantly more information about the progression of T2D toward a variety of outcomes. We additionally demonstrate how structural information from the proposed graph can be used to augment the performance of transformer-based models on predictive tasks, especially when the data are noisy. By increasing the order, or memory, of the graph, we show how the proposed method illuminates key risk factors while successfully ignoring noisy elements, which facilitates analysis that is simultaneously accurate and interpretable.**

***Index Terms*—disease trajectories, sequence modeling, higher-order networks**

I. Introduction

In this era of digital medicine, computational analysis of historical patient data is a foundational approach for generating evidence-based insights into patient care, as well as developing new knowledge surrounding the etiology, risk factors, and progression of health conditions [1], [2]. While each assessment of an individual occurs at a discrete point in time, it is critical to recognize that data collected from these observations are not independent. The nature of human disease and the structure of the healthcare system itself impose temporal dependencies that connect information across an individual's lifetime [3]–[5]. As a result, appropriately utilizing historical data requires the capability to model not only the incidence of prior events but also the relationships among data over time.

To capture these complex interactions between events over time, researchers have widely adopted supervised neural architectures [6], [7]. In contrast to traditional, unsupervised trajectory models such as latent growth curves, group-based trajectory models, and temporal clustering [8], [9], these techniques are designed to directly learn relationships between patterns in sequential data and outcome incidence. Building on successes of recurrent neural network (RNN) architectures like long short-term memory (LSTM), transformer models have achieved state-of-the-art results in modeling sequence data by leveraging innovations such as self-attention and positional encodings [10], [11]. In the healthcare domain, models such as BEHRT and Med-BERT have translated the successes of BERT (Bidirectional Encoder Representations from Transformers) to tasks like disease prediction [12]–[14]. While these models have demonstrated a strong ability to predict if a future event will occur, they fail to provide insight surrounding how the outcome was reached.

The process, however, of systematically identifying these pathways across a range of health conditions presents a significant challenge. As electronic health records collect larger and more heterogeneous volumes of personalized data, temporal patterns become obfuscated by an intractable number of paths connecting information collected over time. This challenge is compounded by the fact that real-world data often contains gaps along relevant sequences, barring progression in a perfectly linear fashion, which can introduce significant amounts of noise to a trajectory. For example, diagnoses made during an emergency room visit following a car accident may interrupt sequences within a multi-year trajectory of congestive heart failure. While attention mechanisms can be used to capture relationships of data to an outcome, the specific dependencies between steps in a sequence remain difficult to extract, as the weights from the local interactions across each distinct attention head oversimplify the complex set of multilayer computations that are used to make a given prediction [15].

Seeking more efficient and interpretable representations of sequence data, researchers have leveraged graphs (networks) to disentangle these overlapping and heterogeneous relationships. This approach has found success in many tasks such as predicting future comorbidities [16], inferring causality [17], identifying disease subtypes [18], and trajectories [19], [20]. Despite their advantages, graphs often underfit the complex relationships that govern trajectories [21], [22], as they operate on a first-order Markovian paradigm, such that the next step in a sequence depends on probabilities from the current state (Fig 1). To address this, higher-order networks (HONs) were proposed to detect dependencies (i.e., conditional probabilities) across multiple steps in a sequence, and encode those dependencies in a graph [23]. Using this approach enables far more robust and interpretable trajectories to be built in

*corresponding author

2575-2634/24/$31.00 ©2024 IEEE
DOI 10.1109/ICHI61247.2024.00010

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

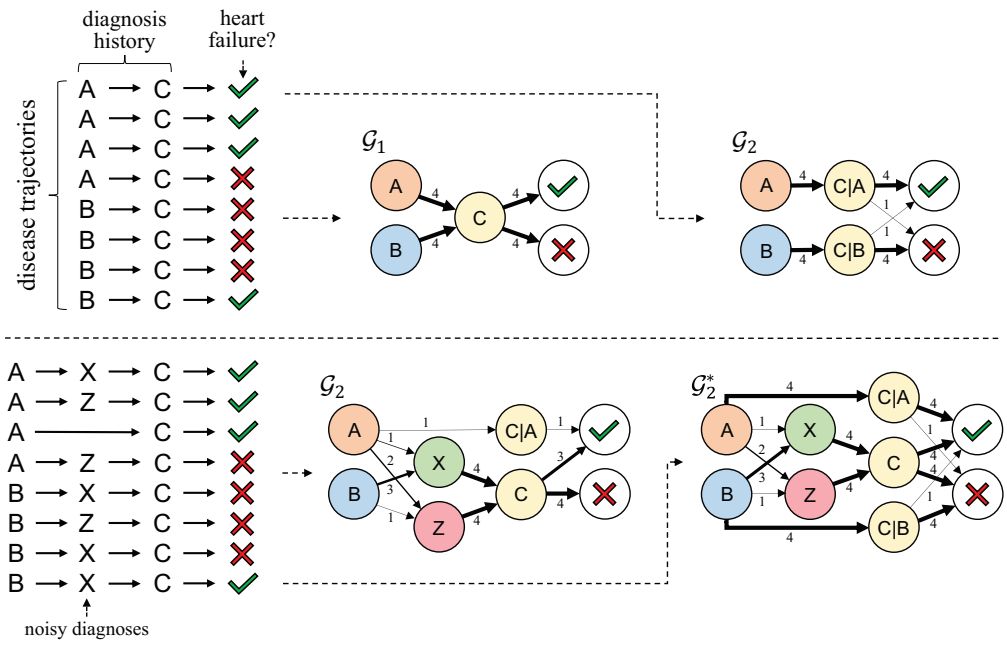

Fig. 1. Given a set of disease trajectories (top), a FON $\mathcal{G}_1$ cannot capture the higher-order dependency that C is more likely to lead to heart failure when it is preceded by A. A HON with $k = 2$ ($\mathcal{G}_2$) addresses this problem by creating the conditional nodes C|A and C|B. However, when the noisy diagnoses X and Z are present (bottom), $\mathcal{G}_2$ fails to detect the importance of C|A and C|B. Our proposed model, $\mathcal{G}_2^*$, addresses this problem by explicitly conditioning paths on the outcome variable and enabling them to skip noisy diagnoses. Edge weight is represented by arrow thickness.

a computationally feasible manner. HONs and similar approaches have found early applications in several tasks both within and beyond the healthcare domain, including modeling diabetes comorbidities [24], anomaly detection for medication order [25], and many more [26]–[29]. Unfortunately, HONs exhibit a fundamental limitation, as they capture transitions between steps in a strictly linear fashion. Thus, without knowing *a priori* which data are relevant for a particular pathway, existing HONs cannot detect which steps are most relevant for a particular long-term outcome, and likewise cannot robustly handle the noisy probabilities that we observe in real-world health data.

Inspired by the generalizability and interpretability of graphs, this manuscript proposes a novel, supervised implementation of a HON to address exactly this. First, it introduces a method for extracting the steps along a trajectory that are most informative of an outcome, even in the presence of noisy or irrelevant subsequences. Next, it demonstrates how the interpretable path information learned by our noise-resilient model can be integrated into state-of-the-art transformer models like BERT, to further improve already strong estimates of outcome probability. Additionally, it shows that our proposed method can be used to solve more general problems, like isolating key risk factors along a trajectory of diagnoses. Ultimately, the manuscript concludes with a discussion regarding the future extensions of the proposed method for complex care patterns and translation into other clinical scenarios.[1]

II. Higher Order Networks: An Overview

To facilitate analysis, digital health data have commonly been structured to mirror paper records by characterizing an individual as a set of discrete data elements. These elements can then be mapped easily to transactional database structures for which statistical and machine-learning methods have predominantly been developed. Although possible, the ability to model conditional relationships between two (or more) events requires large and highly sparse arrays of data representing pre-defined combinations of all data elements. Consequently, network analysis has gained recognition due to its ability to model observed co-occurring data without defining the entire matrix of possibilities. To do so, networks typically utilize directed graphs. Yet these representations operate under a first-order Markovian paradigm, implying transitions to a future next step are a direct result of the current position. This limits the capacity of these networks to capture temporal trends and explicate progression. For example, Figure 1 shows a typical graph $\mathcal{G}_1$, or a first-order network (FON). In these disease trajectories, the transition $A \rightarrow C$ is the key predictor of heart disease; however, because graphs rely on transitive paths of pairwise connections, the structure of $\mathcal{G}_1$ cannot account for this.

To address the limitations of FONs, Xu et al. [23] proposed higher-order networks (HONs), which create conditional

[1]Code is publicly available at https://github.com/sjkrieg/growhonsup.

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

nodes (e.g., $B|A$, read as B given A) to encode higher-order dependencies (i.e., conditional probabilities in sequence data) in a graph [23], [28]. Intuitively, a HON is a graph with memory. The difference between a conditional node like $B|A$ and its lower-order relative B is typically quantified via relative entropy over the transition probabilities of its future states [30]. For example, consider a diagnosis of hypertension (ICD9 code 401). When this diagnosis is preceded by a diagnosis of renal failure (ICD9 code 584), it is more likely to be succeeded by other kidney-related diseases [24]. If this change in probability is sufficiently high across all diagnoses, the relative entropy would be high, and the HON would create a conditional node $401|584$. While HON models have produced new insights in many domains, they have been limited to unsupervised learning settings. When considering disease trajectories, we are often interested in studying not just divergent probabilities at each step, but *differences in outcomes* across all patients. For example, if we are studying factors that lead to heart disease, even if the transition probability distribution of $401|584$ differs from 401, there is no guarantee that this difference provides any meaningful information about the risk factors or progression of the disease. Further, although the conditional nodes in a HON allow it to consider more complex interactions between historical data, prior models a strictly linear, meaning that once data are entered into the trajectory, a HON cannot detect or skip irrelevant elements. As the HON $\mathcal{G}_2$ exemplifies in Figure 1, this results in fragmented paths that often fail to capture the key relationships that drive the outcome. Constructing a HON in a supervised learning setting thus poses the following key challenges:

1) How to define and detect outcome-driven higher-order dependencies (i.e., how to define the set of conditional nodes).
2) How to encode the higher-order dependencies (i.e., how to connect the conditional nodes).

III. Methods

A. Preliminaries

Let $\mathcal{S} = \{S_1, S_2, ..., S_n\}$ be a set of observed **trajectories**, where each $S_i = \langle s_1, s_2, ..., s_m \rangle$ is a sequence of **entities** such as diagnosis codes. Let $\mathcal{A} = \bigcup \mathcal{S}$ denote the set of entities across all sequences. In the supervised learning setting, we additionally consider an outcome or class variable $Y = \{0, 1\}^n$. We also use the vector notation $\mathbf{y} = (y_1, y_2, ..., y_n)$ to denote the observations of Y. To model the system, we can create a graph, or network, $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ where the node set $\mathcal{V}_1 = \mathcal{A}$ and the edge set $\mathcal{E}_1$ is the set of node pairs $(u, v) \in \mathcal{V}_1 \times \mathcal{V}_1$ that are adjacent elements in at least one S_i. To represent Y within the graph, we also create a sink node (i.e., a node with no out-edges) for each possible outcome.

In a HON $\mathcal{G}_k = (\mathcal{V}_k, \mathcal{E}_k)$, $\mathcal{V}_k \subseteq \mathcal{A}^k$ is a set of conditional nodes and $k > 1$ denotes the maximum order of the graph. We denote each $u' \in \mathcal{V}_k$ as a tuple, i.e., $u' = (a_0, a_1, ..., a_m)$ where $m <= k$. We refer to a_m as the **base node** and each other a_j as **conditions**, and in practice use the notation $u' = a_m|a_1, ..., a_{m-1}$ to emphasize the fact that u' encodes a conditional distribution. In other words, k indicates the maximum number of conditions that can be encoded in a single node. Like $\mathcal{E}_1$, the edge set $\mathcal{E}_k \subseteq \mathcal{V}_k \times \mathcal{V}_k$ is a set of node pairs; however, because each $u' \in \mathcal{V}_k$ represents multiple entities, each pair of nodes $(u', v') \in \mathcal{E}_k$ naturally represents a higher-order interaction while remaining a 2-tuple. Edges are directed such that $(u, v) \neq (v, u)$ and weighted via $w_k : \mathcal{E}_k \to \mathbb{R}_{\geq 0}$, determined by the population prevalence and where 0 indicates that there is no edge from u to v.

B. Detecting outcome-driven higher-order dependencies

To address the key challenges presented above, we first define a measure of conditional entropy on a generic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with respect to Y. This allows us to identify and encode the conditional nodes that we expect to reduce this entropy. First, we compute the class entropy Y according to the standard formula:

$$h(Y) = -\sum_{y \in Y} P(y) log_2 P(y). \tag{1}$$

To compute class entropy with respect to $\mathcal{G}$, let us first assume that for each of the possible outcomes of Y, a corresponding sink node exists in $\mathcal{G}$ (though, for notational simplicity, we will refer to $\mathcal{V}$ as though it does not include these sink nodes). Let $\pi_{\mathcal{G}} : \mathcal{A} \to Y$ denote a random walker that begins at a random first-order node and continues traversing random edges in $\mathcal{G}$ until it reaches one of the sink nodes. We define the conditional entropy of Y given $\mathcal{G}$ as follows:

$$h(Y|\mathcal{G}) = -\sum_{u \in \mathcal{A}} P(u) \sum_{y \in Y} P(\pi_{\mathcal{G}}(u) = y) log_2 P(\pi_{\mathcal{G}}(u) = y), \tag{2}$$

where $P(u) = \frac{\sum_{v \in \mathcal{N}(u)} w(u,v)}{\sum_{(u,v) \in \mathcal{E}} w(u,v)}$ (i.e., the normalized, weighted out-degree of u) is the probability that the random walker starts at u. We can easily estimate $P(\pi_{\mathcal{G}}(u) = y)$ by sampling random walks on $\mathcal{G}$. Finally, we can formulate the information gain of $\mathcal{G}$ with respect to Y as:

$$IG(Y, \mathcal{G}) = H(Y) - H(Y|\mathcal{G}), \tag{3}$$

While the combinatorial nature of $\mathcal{G}$ means that we cannot optimize $IG(Y, \mathcal{G})$ directly, we can do so indirectly by creating conditional nodes that will reduce the entropy of $\pi_{\mathcal{G}}$. Since $\mathcal{G}$ is a summary of the observed paths $\mathcal{S}$, nodes that provide meaningful information about Y with respect to $\mathcal{S}$ will also reduce $IG(Y, \mathcal{G})$. Given a candidate conditional node $u' \in \mathcal{A}^k$, we can compute its conditional entropy according to:

$$h(Y|u') = \sum_{y \in Y} P(y, u') log_2 \frac{P(y, u')}{P(u')}, \tag{4}$$

where $P(u')$ is the probability that u' is a **subsequence** of some S_i, i.e., that each $a_i \in u'$ is in S_i and in the same

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

relative order. Finally, we compute the information gain (IG) of u' with respect to Y as:

$$IG(Y, u') = h(Y) - h(Y|u'), \tag{5}$$

This formulation allows us to compute information gain for each candidate u', but does not tell us how to interpret the computed value. Unsupervised HON methods use a user-defined threshold function, but this approach is ad hoc [28]. To more objectively determine if the $IG(Y, u')$ value for a candidate conditional node occurs due simply noise, rather than a true dependency, we can utilize a Fisher-Yates shuffle. To do so, we build a reference distribution for each candidate u' by randomly permuting the outcome vector ($\mathbf{y}$) 100 times and recomputing the information gain. We then directly compute the likelihood that the observed $IG(Y, u')$ is greater than expected from a distribution where the outcome has been randomly assigned. Using a t-statistic for $IG(Y, u')$, which we denote as $t_{u'}$, we construct the set of nodes as follows:

$$\mathcal{V}_k = \mathcal{A} \bigcup \{u' \in \mathcal{A}^k : t_{u'} \geq \alpha\}, \tag{6}$$

where α denotes the standard score threshold, which we default to 1.0.

C. Skipping noisy data

After constructing $\mathcal{V}_k$, we must determine which nodes to connect; i.e., how to construct $\mathcal{E}_k$. Although prior HON methods construct edges in a strictly linear fashion [28], [30], the supervised nature of our proposed approach allows us to focus the growth of the network to consider subsequences rather than substrings (i.e., where each $a_i \in u'$ is directly adjacent in S_i). This is demonstrated in our formulation of Eq. 4, where we compute $IG(Y|u')$ in a way that does not enforce a strict adjacency constraint and enables the model to "skip" noisy diagnoses. We construct $\mathcal{E}_k$ similarly. Specifically, for each S_i in $\mathcal{S}$, we add edges such that each node $u' \in \mathcal{V}_k$ that is also a subsequence of S_i is connected to every node that succeeds it in S_i, including the outcome (sink) nodes.

D. Scalability trade-offs

Although the approach above removes the linear constraint and thus allows for much greater model expression, it has the drawback of increased computational cost, since each S_i has $\sum_{j=1}^{k} \binom{|S_i|}{j}$ subsequences to consider. In other words, the proposed method scales poorly with the combination of larger k and longer sequences. If we did not allow for skip steps, we would only need to consider $\sum_{j=1}^{k} (|S_i| - j)$ subsequences for each S_i. Some form of pre-pruning would mitigate this problem, but risks missing useful higher-order combinations. Fortunately, computing Eq. 6 is embarrassingly parallelizable. Additionally, for larger data sets, computational overhead could be reduced by considering a minimum frequency threshold x such that each u' is only tested if it appears at least x times in $\mathcal{S}$. Future work could explore additional strategies for increasing the sparsity of these skip connections.

TABLE I
REFERENCE FOR ICD9 CODES IN THE T2D DATA SET.

Code	Description
250	Diabetes mellitus
401	Essential hypertension
414	Ischemic heart disease
428	Congestive heart failure
458	Hypotension
461	Sinusitis
462	Pharyngitis
584	Renal failure
585	Chronic kidney disease

IV. EXPERIMENTS

A. Data

For the experiments presented in this work, we leveraged a longitudinal claims dataset for type 2 diabetes (T2D) patients in the state of Indiana. The complexity of structural relationships between T2D comorbidities makes it a particularly compelling use case [31], and these particular data have been utilized in prior HON works to allow for direct comparison to existing baselines [24]. Each S_i represents a patient, and each entity in S_i is an ICD9 major code representing that patient's diagnosis history. In total, the data comprises 908 distinct ICD9 codes across 913,475 distinct patients, all of whom received either a clinical diagnosis of T2D, reported a laboratory glycated hemoglobin (HbA1C) test result of at least 6.5%, or was prescribed at least one Medi-Span-defined anti-diabetes medication. After preprocessing according to the procedure proposed by [24] (discarding E and V codes, removing repeated diagnoses), each trajectory contained an average of 14.3 diagnoses.

Table I provides a summary of the diagnosis codes we frequently reference. Our experimentation centers on three patient outcomes that have been relevant to other studies of T2D trajectories: congestive heart failure (ICD9 code 428), hypotension (ICD9 code 458), and acute renal failure (ICD9 code 584) [24], [31]–[33]. For each of these outcomes, we performed these additional preprocessing steps:

1) We assigned a positive label to all patients for whom the outcome code appears in their trajectory, and a negative label to all others.
2) For patients with a positive outcome, we removed both the outcome code and any subsequent diagnoses. This is because we are interested in predicting the outcomes, i.e. modeling the events that precede the outcome.
3) We discarded any trajectories with fewer than 5 diagnoses preceding the outcome.

Table II summarizes the final set of trajectories for each outcome, which we then partitioned into training, validation, and testing sets according to an 80/10/10 ratio. In all cases, we built graphs and trained predictive models using only the training set. For predictive tasks, we used the validation set to select the best hyperparameters and report only the results as measured on the testing set.

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

TABLE II
SUMMARY OF OUTCOMES FOR T2D PATIENT TRAJECTORIES.

Outcome	# train/val/test	P(Y=1)	h(Y)	$\lvert\mathcal{V}_1\rvert$	$\lvert\mathcal{E}_1\rvert$	$\lvert\mathcal{V}_2^*\rvert$	$\lvert\mathcal{E}_2^*\rvert$	$\lvert\mathcal{V}_3^*\rvert$	$\lvert\mathcal{E}_3^*\rvert$	$\lvert\mathcal{V}_4^*\rvert$	$\lvert\mathcal{E}_4^*\rvert$
Heart failure	460,179/57,523/57,522	0.1068	0.4904	906	281k	10.6k	6.7m	16.4k	9.8m	18.2k	10.8m
Hypotension	473,736/59,218/59,217	0.0414	0.2485	906	289k	15.3k	9.9m	33.3k	27.4m	45.6k	36.0m
Renal failure	472,800/59,100/59,100	0.0480	0.2777	906	289k	13.9k	8.8m	29.8k	27.3m	45.2k	39.2m

B. Tasks and experimental setup

To robustly assess the proposed model, we designed a series of analyses that quantify how much information each graph provides about an outcome and demonstrate their utility for traditional outcome-prediction tasks. For the proposed HON, we set $k = \{2, 3, 4\}$, where 4 is both the highest value with which we could compute our model efficiently and the point at which higher-order dependencies saturated in prior work on HONs of T2D disease trajectories [24], and constructed $\mathcal{G}_k^*$ for each of the aforementioned outcomes. We evaluated multiple graphs as baselines, including a FON ($\mathcal{G}_1$), existing unsupervised HON ($\mathcal{G}_4$) [30], and ablated versions of the proposed HON without the skip-step ($\mathcal{G}_4'$).

1) Information Gain of Graphs and Paths: We first evaluated how well the global structure of each graph corresponded to meaningful disease pathways. To do this, we quantified the overall information of each graph with respect to each outcome using our formulation of $IG(Y, \mathcal{G})$ (Eq. 3). Specifically, for each first-order node, we used a random walker to sample 10,000 paths. For each path, the walker began at the corresponding first-order node and continued traversing the graph until reaching one of the outcome (sink) nodes. We used these paths to compute $h(Y|\mathcal{G})$ (Eq. 2), and subsequently $IG(Y, \mathcal{G})$. We repeated this procedure 10 times for each graph and reported the average and standard deviation. Metrics produced in this way are meaningful as they quantify the amount of information each graph provides as related to the purity of paths in reaching an outcome.

Next, we analyzed the local structures of our proposed model, $\mathcal{G}^*$, by extracting the most informative nodes and paths. Using the same random walking procedure, we computed $IG(Y, u')$ (Eq. 5) for all nodes (including conditional nodes) in each graph. To compute the information gain for a path, we calculated the mean $IG(Y, u')$ for each u' in the path. This procedure allows us to validate whether the informative nodes and paths in $\mathcal{G}_4'$ match known clinical relationships between diagnoses and outcomes.

Finally, we compared both the existence and informativeness of paths leading to a positive outcome to assess if the proposed supervision and skip-step resulted in meaningful differences in learned network structure. Utilizing 1,000,000 randomly sampled paths leading to the heart failure outcome, again extracted from the random walker, we identified the most informative trajectory (defined by the multiplicative weight of $IG(Y, u')$, as calculated above, for each node u' in the trajectory) beginning at each of the 907 ICD codes in the dataset. We then calculated: 1) the percentage of paths that existed exactly as sampled from $\mathcal{G}^*$ in each baseline network, and 2) the percentage of paths for which the IG in $\mathcal{G}^*$ was higher than that in the comparison network. For paths that could be not exactly reproduced in a comparison network, we recursively pruned conditions from the higher-order nodes and attempted to continue the path. For example, if an edge from 403 to 414|403 did not exist, we attempted to traverse the baseline network using the edge from 403 to 414 directly. If a path was still unable to be found, no comparison was made to prevent biased denominators. We note that as the multiplication of IG across a trajectory will be biased based on path length, we grouped paths based on their exact length (2 through 6) before performing sensitivity analysis for "top" paths.

2) Outcome Prediction: Second, we explored the use of $\mathcal{G}_4^*$ for a classification task. Although a HON provides increased interpretability of trajectories, it is generated at the population level and not directly applicable to the prediction of an individual's outcome. As such, this analysis sought to determine how trajectories learned by a HON can be integrated into a supervised classification framework as compared to state-of-the-art sequence models (i.e., BERT-based models). In this case, the sequence of diagnoses in each disease trajectory serves as input to a model, whose job is to map the trajectory to a binary outcome (e.g., 1 for heart disease, and 0 for no heart disease). Our baseline, BEHRT, achieves this by 1) mapping each token (entity) from the trajectory to an embedding vector, and 2) passing the token embeddings through a stack of transformer encoder blocks [12]. Following the procedure proposed by the seminal BERT model [11], BEHRT pre-trains the encoder blocks via token masking and a stack of decoder blocks. Slight variants of this approach have been proposed specifically for disease prediction [13], [14], but differences among them are minor, so we consider BEHRT as a representative baseline. Our experiments used the public code and hyperparameters as published by [12].

To classify trajectories using a graph $\mathcal{G}$, we map the problem to a subgraph classification task. Given an input trajectory, we use a graph neural network (GNN) to compute an embedding vector for each node. When $k > 1$, we map tokens to higher-order nodes whenever possible, prioritizing combinations that are temporally close in the trajectory. For example, given an input trajectory $\langle ..., 461, ..., 250, ..., 401, ... \rangle$, if node $401|250 \in \mathcal{V}_4^*$, then we use it instead of node 401 to compute the GNN embedding for token 401. If $401|250 \notin \mathcal{V}_4^*$ and node $401|461 \in \mathcal{V}_4^*$, then we use $401|461$. Following BEHRT, we pass the GNN embeddings forward through a stack of trans-

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

former encoder blocks to compute the outputs. Essentially, this means that we replace BEHRT's basic, fully-connected embedding layer (step 1 above) with a GNN, which learns embeddings mediated by the given graph structure. We refer to this model as Transformer/$\mathcal{G}$. Additionally, we considered a model Transformer/$\mathcal{G}$+Embs, which uses a concatenation of the fully connected embedding layer and the GNN embeddings as input to the encoder block. Intuitively, this approach can be described as a modification of BEHRT that uses embeddings from $\mathcal{G}$ to augment the token embeddings. To evaluate the graph-based models, we used a simple two-layer graph convolutional network (GCN) as the GNN. For all models, we used 128 as the token embedding size. This means that to avoid incurring an unfair advantage due to using more parameters, for Transformer/$\mathcal{G}$+Embs, the output dimensions for the GCN and fully connected embedding layer were 64 each. We trained all models using the Adam optimizer, with a learning rate of 0.0001 and a batch size of 32. We trained and tested each model 10 times, and reported the mean and standard deviation of the area under the precision-recall curve (AUPRC), a standard measure of binary classification performance in class-imbalanced settings [12], [34].

TABLE III
SUMMARY OF TOY (PATTERN MATCHING) PROBLEMS.

Name	Diagnosis Pattern	P(Y=1)
Toy 1	$\{461, 462, 463, 464, 465, 466\} \rightarrow 401 \rightarrow 250$	0.0475
Toy 2	$401 \rightarrow \{461, 462, 463, 464, 465, 466\} \rightarrow 250$	0.0644
Toy 3	$401 \rightarrow 250 \rightarrow \{461, 462, 463, 464, 465, 466\}$	0.0736

Ultimately, we extended the analysis of the classification task to directly assess performance on known trajectories with noisy labels. To accomplish this using the T2D trajectories, instead of labeling each trajectory according to a clinical outcome, we assigned labels according to whether the trajectory contained a specific pattern of diagnosis codes. For each of these patterns, which are summarized in Table III, we labeled each trajectory as positive (1) if it matched the pattern (while allowing for *any number* of intermediate diagnoses), and negative (0) otherwise. The particular codes we chose to define these patterns were arbitrary and based on prevalence in the data set (i.e., to ensure that each pattern had a sufficient number of positive examples) rather than on known T2D comorbidities or outcomes. This classification problem is thus conceptually simple but contains the kinds of higher-order dependencies that are prevalent in real-world systems yet non-trivial for many sequence models to represent [23]. We also conducted experiments in which the trajectories were subject to various thresholds of label noise. Here, we denote as **noise ratio** the probability that a trajectory matching the pattern was randomly assigned a negative label. For example, at a noise ratio of 0.4, only 60% of the matching trajectories were assigned a positive label, while the remaining 40% were assigned a (noisy) negative label. Label noise is ubiquitous in real-world applications—for example, in disease trajectory studies, a patient may appear to have a negative outcome only because the positive outcome has not yet been diagnosed and recorded—so it is essential to understand the effect of this type of noise on model performance [35], [36].

V. RESULTS

TABLE IV
$IG(Y, \mathcal{G})$ FOR EACH GRAPH AND DATA SET. REPORTED VALUES ARE THE MEANS AND STANDARD DEVIATIONS OF 10 ITERATIONS OF THE RANDOM WALKING PROCEDURE. BOLD FONT INDICATES THE BEST RESULT FOR EACH DATA SET.

	Heart failure	Hypotension	Kidney failure
$\mathcal{G}_1$	0.0001 $\pm$0.003	0.0037 $\pm$0.002	0.0044 $\pm$0.002
$\mathcal{G}_4$	0.0097 $\pm$0.003	0.0138 $\pm$0.001	0.0159 $\pm$0.003
$\mathcal{G}_4'$	0.0046 $\pm$0.002	0.0011 $\pm$0.001	0.0123 $\pm$0.002
$\mathcal{G}_2^*$	0.0349 $\pm$0.002	0.0324 $\pm$0.002	0.0452 $\pm$0.004
$\mathcal{G}_3^*$	0.0782 $\pm$0.003	0.0661 $\pm$0.002	0.0780 $\pm$0.002
$\mathcal{G}_4^*$	**0.0892** $\pm$0.004	**0.0723** $\pm$0.001	**0.0836** $\pm$0.003

A. Information Gain of Graphs and Paths

Table IV presents $IG(Y, \mathcal{G})$ for each graph and outcome. For all outcomes, we found that only those paths sampled from $\mathcal{G}^*$ provided a substantial increase in the amount of information gain, indicating that the overall purity of the paths was highly distinct between those patients found to have the outcome and those who did not. Notably, despite being grown based on the relationship to the outcome, $\mathcal{G}_4'$ did not perform overly well. This further suggests that without the ability to skip irrelevant diagnoses, even an outcome-driven HON model cannot detect and encode the most informative paths.

TABLE V
CLASSIFICATION PERFORMANCE (AUPRC). BOLD FONT INDICATES THE BEST RESULT FOR EACH DATA SET.

Model	Heart failure	Hypotension	Kidney failure
BEHRT	0.372 $\pm$0.00	0.133 $\pm$0.00	0.222 $\pm$0.00
BEHRT/$\mathcal{G}_1$	0.305 $\pm$0.02	0.109 $\pm$0.01	0.172 $\pm$0.01
BEHRT/$\mathcal{G}_1$+Embs	0.375 $\pm$0.00	0.134 $\pm$0.00	0.228 $\pm$0.00
BEHRT/$\mathcal{G}_4^*$	0.321 $\pm$0.01	0.115 $\pm$0.01	0.190 $\pm$0.01
BEHRT/$\mathcal{G}_4^*$+Embs	**0.386** $\pm$0.01	**0.144** $\pm$0.01	**0.237** $\pm$0.01

Looking next to the comparison of sampled heart failure paths in $\mathcal{G}_4^*$ with other graphs, Figure 2A, demonstrates that, on average, well below 50% of paths leading to heart failure in the proposed graph could be replicated in any of the comparison graphs. Note that a path of length 1 (i.e., a single node) will always have a 100% match for all networks, as there is always a non-zero probability of reaching the outcome of interest from any given starting point. Unsurprisingly, shorter trajectories had a higher prevalence of overlap across the various network configurations. However, a small subset of the best paths for rare and/or unrelated root nodes were found to contain all first-order steps, allowing for some exact matches to the FON beyond a path length of 2. Critically, even after pruning higher-order dependencies at each step, we were only able to recover matches for just over 74% of the paths in

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

$\mathcal{G}_1$, 62% in $\mathcal{G}_4'$, and only 40% in $\mathcal{G}_4$ across all sensitivity analyses, indicating that edges learned by $\mathcal{G}_4^*$ never received enough support to even be grown at a first-order in these earlier methods. Additionally, the low match rate for the unsupervised HON $\mathcal{G}_4$, even compared to the FON $\mathcal{G}_1$, demonstrates how the dependencies detected by an unsupervised model cause the network to overfit on fragmented trajectories instead of encoding meaningful paths.

Moving beyond the simple existence of paths, Figure 2B, illustrates that the individual paths in $\mathcal{G}_4^*$ as compared to those extracted from baseline networks (evaluating exact matches when possible and, alternatively, pruned matches), as denoted by the average purity of the path in reaching a given outcome. Again, shorter paths have more similar performance across multiple networks, because the statistical support for paths that only include 1 or 2 nodes before an outcome is likely to be also found in the baseline networks. However, for paths with a length of at least 3, $\mathcal{G}_4^*$ can generate paths with superior purity in over 90% of cases.

B. Outcome Prediction

As Table V shows, on the classification task, we observed consistent performance trends across all outcomes. First, we note that the transformer with only GNN embeddings (Transformer/$\mathcal{G}$) significantly underperformed relative to BEHRT using either $\mathcal{G}_1$ or $\mathcal{G}_4^*$. On the other hand, augmenting the inputs with graph embeddings consistently improved transformer performance, especially with $\mathcal{G}_4^*$. With a more expressive GNN implementation, these improvements would likely be even more substantial [29].

Figure 3 summarizes the results of the classification experiments on the toy pattern matching task, which illuminate a key reason for the effectiveness of $\mathcal{G}_4^*$. At a noise ratio of 0, the task was trivial due to the ability of the transformer to match the higher-order patterns, and both models achieved AUPRC $\geq$0.999. While the noise ratio remained below 0.5, the performance of both models was virtually identical, so these are not shown to preserve scale. As the noise ratio increased, AUPRC consistently decreased for both models. While this was expected, we also noticed that a performance gap emerged as the noise ratio surpassed 0.5. At the noisiest ratio of 0.9, the differences in AUPRC were relatively large in favor of Transformer/$\mathcal{G}_3^*$+Embs over BEHRT: 0.112 and 0.053, respectively, on Toy 1 (+113%); 0.152 and 0.096, respectively, on Toy 2 (+58%), and 0.122 and 0.075, respectively, on Toy 3 (+62%). These results suggest that at least one of the reasons that Transformer/$\mathcal{G}_3^*$+Embs performs well on the outcome prediction tasks is that the embeddings from $\mathcal{G}_3^*$ help improve the robustness of the encoder against label noise.

VI. Discussion

This work has introduced a novel implementation of higher-order networks able to extract meaningful trajectories towards an outcome of interest even in the presence of noise commonly found in real-world data. As evidenced by the results above, simply replacing the objective function for a HON model to enforce the growth of paths that maximize purity in reaching an outcome is not sufficient for detecting the most informative paths. This is hardly surprising in a healthcare context; disease trajectories encompass a variety of diagnoses elicited from various encounters within the health system, many of which are unlikely to be relevant to the outcome of concern. In this case, these noisy intermediate diagnoses only function to reduce the statistical support for the relevant combinations of diagnoses, making it less likely that a model can identify them.

We see strong evidence of this phenomenon in the *Information Gain* analysis (Section V-A). As expected, the supervised HON without a skip step ($\mathcal{G}_4'$) had the highest percent of matching paths to our proposed model, as both are designed to grow paths and create higher-order dependencies using the same objective function. However, less than 20% of paths $\geq$4 matched exactly without the inclusion of the skip-step, strongly illustrating how noise from unrelated diagnoses can fragment trajectory growth. Looking deeper, we note that 60% of the paths extracted from the proposed network could be reconstructed in this non-skip step network by pruning higher-order edges; i.e. edges between relevant diagnoses were created, but conditional higher-order paths were not constructed given noise. As it is clear these reduced paths are not as informative (Figure 2B), investigation of higher-order nodes can provide additional insight into what information is lost.

In addition to analyzing paths in $\mathcal{G}_4'$, we can also study individual nodes. As Table II shows, for the heart failure, hypotension, and renal failure outcomes, the proposed method created 17,325, 44,648, and 44,307 conditional nodes, respectively, in each $\mathcal{G}_4^*$. In the case of heart failure, 27% of these were second-order nodes, 62% were third-order nodes, and the remaining 11% were fourth-order nodes. The five most common conditions were chronic ischemic heart disease (ICD9 code 414), essential hypertension (401), hypercholesterolemia (272), cardiac dysrhythmia (427), and anemia (285)—all of which are known to be strongly correlated with heart failure [37], [38]. Of the 17,325 conditional nodes, 8,372 (48%) were conditioned one at least one of these five diagnoses. We observed similar patterns by analyzing the individual nodes with the highest IGs. Almost all these nodes used a base diagnosis of cardiomyopathy (IC9 code 425) in conjunction with various conditions; for example, 425|511|427 (cardiomyopathy given pleurisy given cardiac dysrhythmia) produced the highest IG (0.051). Besides validating that the discovered nodes are truly informative of the heart disease outcome, examining these nodes highlights certain combinations of risk factors and drive further investigation.

Turning next to the utilization of the identified trajectories, the *Outcome Prediction* analysis (Section V-B) provided a novel approach to improve the interpretability of sequence prediction tasks while maintaining predictive performance. The interplay between these factors is particularly relevant in health-centric applications, where the utility and adoption of computational models rely not only on a measure of accuracy but increasingly on an ability to investigate the factors underlying output, which may be used to guide clinical

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

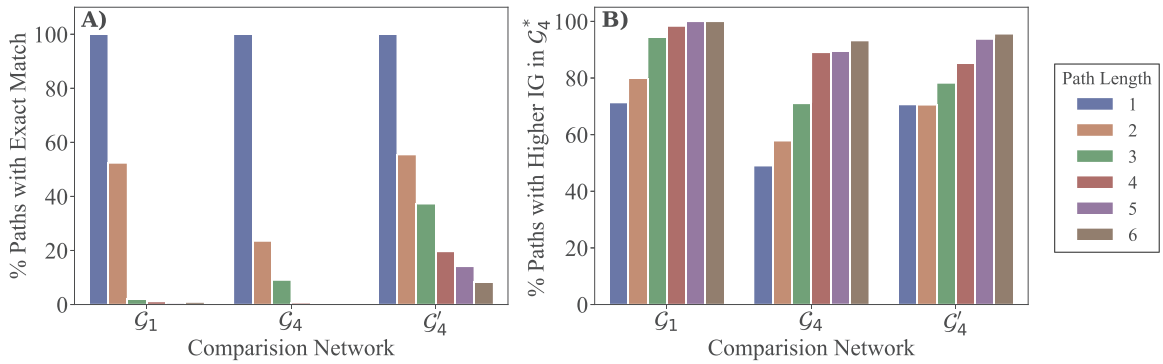

Fig. 2. Comparison of paths towards heart failure (ICD9 code 428) sampled from $\mathcal{G}_4$. Panel (A) shows the percentage of exact matches between paths of length 1-6, while panel (B) shows the percentage of paths for which the Information Gain (IG) was higher in $\mathcal{G}_4$.

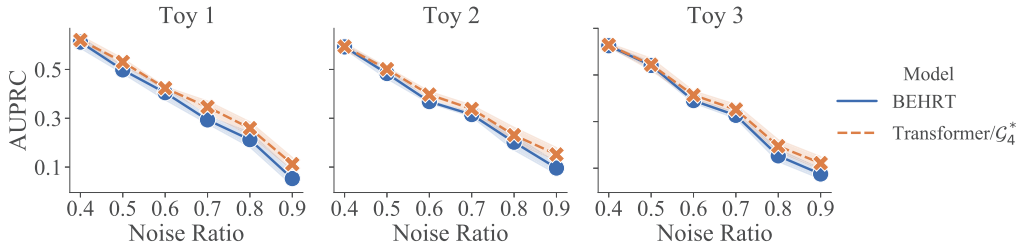

Fig. 3. Classification results on toy trajectories at various levels of label noise. Shading represents the standard deviation of 10 runs for each.

analyses and recommendations [39]–[41].

Given the established state-of-the-art predictive performance and widespread availability in software packages, deep learning transformer architectures are unlikely to be replaced as a standard model for sequence prediction tasks. Yet, it is clear there is no straightforward way for BEHRT or similar models to isolate the trajectories that drive its outputs. For example, while we can inspect the attention weights to identify that a diagnosis code 414 has a probability of transitioning to 250, and that 250 then has some probability of reaching a positive outcome, this transitive interpretation cannot account for the complex dependencies learned by the model over all *combinations* of diagnoses and may be misleading [15]. As a result, we sought to augment these models rather than replace them entirely.

By utilizing a graph to augment the embeddings used for prediction, we can easily and directly examine the relationships between learned edges to compute, for example, the probability distribution over $\pi_{\mathcal{G}}$ (Eq. 2). We can use this distribution to understand how different risk factors could contribute to a prediction. We know by our definition of Eq. 6 that grown edges must provide some information about the probability of reaching the outcome. In the context of a higher-order network, the fact that each node can represent the conditional probability across multiple entities helps us then to understand how these combinations of comorbidities can affect outcome probabilities in a way that transcends the previously-discussed limitations of transformers [30]. For example, the node 425|511|427 produced the highest IG with respect to heart failure in $\mathcal{G}_4'$ (0.051). On their own, we observed IGs of 0.022 and 0.004 for cardiomyopathy (425) and pleurisy (511), respectively, both ranking within the top 15 most informative first-order nodes. For cardiac dysrhythmia (427), on the other hand, we only observed an IG of 0.001—rank 76 of the 906 first-order nodes. And while we saw an increase in IG for the second-order nodes 425|511 (0.026) and 511|427 (0.010) when compared to their first-order counterparts 425 and 511, respectively, these values are still much smaller than the information provided by 425|511|427. The progression of a patient over this third-order sequence of diagnoses thus seems to increase their probability of reaching heart failure in such a way that is more than the sum of its first or second-order parts. This means that the ability to examine such higher-order combinations and probabilities directly is a critical piece of a truly interpretable model.

Further, we are often led to believe there must be a trade-off between accurate, complex models and those that are interpretable. As a result, a simple non-inferiority of the augmented network in terms of predictive performance would be valuable. However, as we examine the results of this augmentation, we find it promising that results not only equal but slightly exceed that of the existing transformer. Other studies have noted that a transformer's dense embedding layer is equivalent to a fully connected graph and characterized the learning process of a transformer as learning the edge weights of an unspecified graph [42]. Therefore, it is not surprising

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

to see that such a model would overfit on noisy patterns, where the additional information gained by the more targeted graph provides useful information in learning weights for the transformer model. This idea is supported by the results of our analysis of the toy paths. We see a clear pattern in which the augmented graph+transformer model improves performance as the proportion of noise injected into a path increases. Such noise is abundant in healthcare data because of causes like time (e.g., a T2D patient has not *yet* been diagnosed with heart disease) and false negatives, so robustness to it is an extremely useful characteristic of any model.

Limitations: There are several limitations to the work presented here. First, although a large dataset was used to assess the performance of the proposed methodology on three distinct disorders and three toy examples, we recognize the data used only claims from a single state. Second, this model evaluated only trajectories of diagnoses. It is unlikely diagnosis history is the only factor driving a patient to reach a specific clinical outcome. As such, performance may be improved with the inclusion of a more comprehensive set of clinical factors (procedures, laboratory results, etc.). Current work is underway by the study team to extend the methodology to account for sequences of heterogeneous nodes. Finally, it is becoming increasingly important to consider the computational cost of the models we build. In some cases, the marginal improvements of the proposed model on the classification performance (Table V) may not be worth the additional computation cost of constructing $\mathcal{G}_3^*$ and using a GNN.

VII. Conclusion

Given the way healthcare data are captured and utilized, noise is the norm rather than the exception. Drawing on the strengths of state-of-the-art work in EHR data analysis, disease trajectory modeling, and higher-order networks, this manuscript proposed a new method for constructing a HON in a supervised learning context. Different from prior approaches, the proposed method identifies the higher-order dependencies that are most informative of an outcome while skipping noisy or irrelevant steps. Experiments demonstrated that HONs constructed with the proposed method can provide significantly more information about an outcome than other graphs, and can successfully detect and encode dependencies that correspond to known comorbidities and disease pathways in T2D patients. Additionally, the proposed method's robustness to noise can be used to augment transformer-based models, marginally improving their performance on a disease prediction task when label noise is present. These results suggest several fruitful avenues for future work, such as improving the scalability of the proposed method to higher orders and longer trajectories, incorporating more granular information from trajectories (e.g., time intervals and heterogeneous/multimodal data) into the HON structure, and designing more sophisticated methods for using information from the HON to augment neural sequence models.

References

[1] S. Shilo, H. Rossman, and E. Segal, "Axes of a revolution: challenges and promises of big data in healthcare," *Nature medicine*, vol. 26, no. 1, pp. 29–38, 2020.

[2] W. Raghupathi and V. Raghupathi, "Big data analytics in healthcare: promise and potential," *Health information science and systems*, vol. 2, pp. 1–10, 2014.

[3] C. Burton-Jeangros, S. Cullati, A. Sacker, and D. Blane, "A life course perspective on health trajectories and transitions," *Life Course Research and Social Policies*, 2015.

[4] S. J. Henly, J. F. Wyman, and M. J. Findorff, "Health and illness over time: The trajectory perspective in nursing science," *Nursing research*, vol. 60, no. 3 Suppl, p. S5, 2011.

[5] D. W. Hollar and D. W. Hollar, "Understanding the evolutionary historical background behind the trajectories in human health and disease," *Trajectory Analysis in Health Care*, pp. 211–230, 2018.

[6] A. Allam, S. Feuerriegel, M. Rebhan, and M. Krauthammer, "Analyzing patient trajectories with artificial intelligence," *Journal of medical internet research*, vol. 23, no. 12, p. e29812, 2021.

[7] A. Amirahmadi, M. Ohlsson, and K. Etminani, "Deep learning prediction models based on ehr trajectories: A systematic review," *Journal of Biomedical Informatics*, p. 104430, 2023.

[8] G. van der Nest, V. L. Passos, M. J. Candel, and G. J. van Breukelen, "An overview of mixture modelling for latent evolutions in longitudinal data: Modelling approaches, fit statistics and software," *Advances in Life Course Research*, vol. 43, p. 100323, 2020.

[9] H. Nguena Nguefack, M. Pagé, J. Katz, M. Choinière, A. Vanasse, M. Dorais, O. Samb, and A. Lacasse, "Trajectory modelling techniques useful to epidemiological research: A comparative narrative review of approaches. clin epidemiol 2020 oct; 12: 1205-22," *J Sleep Res*, vol. 29, no. 3, p. e12898, 2020.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint:1810.04805*, 2018.

[12] Y. Li, S. Rao, J. R. A. Solares, A. Hassaine, R. Ramakrishnan, D. Canoy, Y. Zhu, K. Rahimi, and G. Salimi-Khorshidi, "Behrt: transformer for electronic health records," *Scientific reports*, vol. 10, no. 1, pp. 1–12, 2020.

[13] L. Rasmy, Y. Xiang, Z. Xie, C. Tao, and D. Zhi, "Med-bert: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction," *NPJ digital medicine*, vol. 4, no. 1, p. 86, 2021.

[14] Y. Meng, W. Speier, M. K. Ong, and C. W. Arnold, "Bidirectional representation learning from transformers using multimodal electronic health record data to predict depression," *IEEE journal of biomedical and health informatics*, vol. 25, no. 8, pp. 3121–3129, 2021.

[15] H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 782–791.

[16] P. B. Thomas, D. H. Robertson, and N. V. Chawla, "Predicting onset of complications from diabetes: a graph based approach," *Applied network science*, vol. 3, no. 1, p. 48, 2018.

[17] V. Kannan, F. Swartz, N. A. Kiani, G. Silberberg, G. Tsipras, D. Gomez-Cabrero, K. Alexanderson, and J. Tegnèr, "Conditional disease development extracted from longitudinal health care cohort data using layered network construction," *Scientific reports*, vol. 6, p. 26170, 2016.

[18] L. Li, W.-Y. Cheng, B. S. Glicksberg, O. Gottesman, R. Tamler, R. Chen, E. P. Bottinger, and J. T. Dudley, "Identification of type 2 diabetes subgroups through topological analysis of patient similarity," *Science translational medicine*, vol. 7, no. 311, pp. 311ra174–311ra174, 2015.

[19] A. B. Jensen, P. L. Moseley, T. I. Oprea, S. G. Ellesøe, R. Eriksson, H. Schmock, P. B. Jensen, L. J. Jensen, and S. Brunak, "Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients," *Nature communications*, vol. 5, p. 4022, 2014.

[20] W. J. Vlietstra, R. Vos, M. van den Akker, E. M. van Mulligen, and J. A. Kors, "Identifying disease trajectories with predicate information from a knowledge graph," *Journal of biomedical semantics*, vol. 11, pp. 1–11, 2020.

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.

[21] E. Capobianco and P. Liò, "Comorbidity networks: beyond disease correlations," *Journal of Complex Networks*, vol. 3, no. 3, pp. 319–332, 2015.

[22] J. X. Hu, C. E. Thomas, and S. Brunak, "Network biology concepts in complex disease comorbidities," *Nature Reviews Genetics*, vol. 17, no. 10, p. 615, 2016.

[23] J. Xu, T. L. Wickramarathne, and N. V. Chawla, "Representing higher-order dependencies in networks," *Science advances*, vol. 2, no. 5, p. e1600028, 2016.

[24] S. J. Krieg, D. H. Robertson, M. P. Pradhan, and N. V. Chawla, "Higher-order networks of diabetes comorbidities: Disease trajectories that matter," in *2020 IEEE International Conference on Healthcare Informatics (ICHI)*. IEEE, 2020, pp. 1–11.

[25] H. Niu, O. Omitaomu, Q. Cao, O. Ozmen, H. Klasky, M. M. Olama, L. Pullum, T. Kuruganti, M. Ward, A. Laurio *et al.*, "Anomaly detection in sequential health care data using higher-order network representation," Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), Tech. Rep., 2020.

[26] M. Rosvall, A. V. Esquivel, A. Lancichinetti, J. D. West, and R. Lambiotte, "Memory in network flows and its effects on spreading dynamics and community detection," *Nature communications*, vol. 5, no. 1, p. 4630, 2014.

[27] R. Lambiotte, M. Rosvall, and I. Scholtes, "From networks to optimal higher-order models of complex systems," *Nature physics*, vol. 15, no. 4, pp. 313–320, 2019.

[28] M. Saebi, J. Xu, L. M. Kaplan, B. Ribeiro, and N. V. Chawla, "Efficient modeling of higher-order dependencies in networks: from algorithm to application for anomaly detection," *EPJ Data Science*, vol. 9, no. 1, p. 15, 2020.

[29] S. Krieg, W. Burgis, P. Soga, and N. Chawla, "Deep ensembles for graphs with higher-order dependencies," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=hZftxQGJ4Re

[30] S. J. Krieg, P. M. Kogge, and N. V. Chawla, "Growhon: A scalable algorithm for growing higher-order networks of sequences," in *International Conference on Complex Networks and Their Applications*. Springer, 2020, pp. 485–496.

[31] W. Oh, E. Kim, M. R. Castro, P. J. Caraballo, V. Kumar, M. S. Steinbach, and G. J. Simon, "Type 2 diabetes mellitus trajectories and associated risks," *Big data*, vol. 4, no. 1, pp. 25–30, 2016.

[32] S. Maule, G. Papotti, D. Naso, C. Magnino, E. Testa, and F. Veglio, "Orthostatic hypotension: evaluation and treatment," *Cardiovascular & Haematological Disorders-Drug Targets (Formerly Current Drug Targets-Cardiovascular & Hematological Disorders)*, vol. 7, no. 1, pp. 63–70, 2007.

[33] E. Jeong, N. Park, Y. Kim, J. Y. Jeon, W. Y. Chung, and D. Yoon, "Temporal trajectories of accompanying comorbidities in patients with type 2 diabetes: a korean nationwide observational study," *Scientific Reports*, vol. 10, no. 1, pp. 1–10, 2020.

[34] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 233–240.

[35] B. Frénay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 5, pp. 845–869, 2013.

[36] B. Han, Q. Yao, T. Liu, G. Niu, I. W. Tsang, J. T. Kwok, and M. Sugiyama, "A survey of label-noise representation learning: Past, present and future," *arXiv preprint:2011.04406*, 2020.

[37] V. M. van Deursen, R. Urso, C. Laroche, K. Damman, U. Dahlström, L. Tavazzi, A. P. Maggioni, and A. A. Voors, "Co-morbidities in patients with heart failure: an analysis of the european heart failure pilot survey," *European journal of heart failure*, vol. 16, no. 1, pp. 103–111, 2014.

[38] A. Groenewegen, F. H. Rutten, A. Mosterd, and A. W. Hoes, "Epidemiology of heart failure," *European journal of heart failure*, vol. 22, no. 8, pp. 1342–1356, 2020.

[39] K. Rasheed, A. Qayyum, M. Ghaly, A. Al-Fuqaha, A. Razi, and J. Qadir, "Explainable, trustworthy, and ethical machine learning for healthcare: A survey," *Computers in Biology and Medicine*, p. 106043, 2022.

[40] J. Amann, A. Blasimme, E. Vayena, D. Frey, V. I. Madai, and P. Consortium, "Explainability for artificial intelligence in healthcare: a multidisciplinary perspective," *BMC medical informatics and decision making*, vol. 20, pp. 1–9, 2020.

[41] C. C. Yang, "Explainable artificial intelligence for predictive modeling in healthcare," *Journal of healthcare informatics research*, vol. 6, no. 2, pp. 228–239, 2022.

[42] P. Veličković, "Everything is connected: Graph neural networks," *Current Opinion in Structural Biology*, vol. 79, p. 102538, 2023.

Authorized licensed use limited to: UNIVERSITY NOTRE DAME. Downloaded on March 08,2025 at 19:37:09 UTC from IEEE Xplore. Restrictions apply.