Higher-order Networks of Diabetes Comorbidities: Disease Trajectories that Matter

Steven J. Krieg†, Daniel H. Robertson‡, Meeta P. Pradhan‡, and Nitesh V. Chawla†*
†University of Notre Dame, Notre Dame, IN
{skrieg, nchawla}@nd.edu
‡Indiana Biosciences Research Institute, Indianapolis, IN
{drobertson, mpradhan}@indianabiosciences.org

2020 IEEE International Conference on Healthcare Informatics (ICHI) | 978-1-7281-5382-7/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICHI48887.2020.9374390

***Abstract*—Networks are powerful and flexible structures for modeling relationships in medical and biological systems, but in a traditional first-order network representation, an edge typically expresses a relationship between a single pair of nodes. In order to analyze complex relationships between groups of nodes, researchers rely on combined sets of these pairwise connections, which can misrepresent the true relationships in the underlying data. Higher-order networks, on the other hand, capture the higher-order dependencies that go beyond the pairwise interactions, and thus can encode more complex relationships within a familiar structure. In this study, we created and analyzed higher-order networks of disease trajectories generated from the records of 913,475 type 2 diabetes patients. We show that higher-order networks provide a more accurate representation of the underlying disease trajectories than traditional first-order networks. We also analyze differences in PageRank scores and community structure at higher orders and discuss the implications of these differences for the future study of comorbidity networks.**

***Index Terms*—diabetes, comorbidities, disease trajectories, higher order networks**

I. Introduction

Networks are powerful structures for expressing complex and interdependent relationships among entities. Many studies of biological systems have embraced networks to model problems such as patient similarity, disease progression, protein interactions, drug repositioning, and gene expression [1]. To create a network, researchers have traditionally generated a set of nodes, which represents a collection of real entities, and a set of edges, which connects nodes in the network and represents relationships between the entities. However, such networks are only able to express relationships between a pair of entities at a time. Studying relationships between larger groups of entities requires researchers to assume the **Markov property**, which means that, for a given process, knowledge of its future states depend only on information that is immediately available at the present state. Given two edges $a \rightarrow b$, and $b \rightarrow c$ in a network, the Markov property manifests itself via a random walker that moves from a to b to c, even if the entities represented by a and c are unrelated outside the network. In other words, this network, which we call a **first-order network**, transitively infers a connection between a and c because of their mutual connection to b.

*Corresponding author.

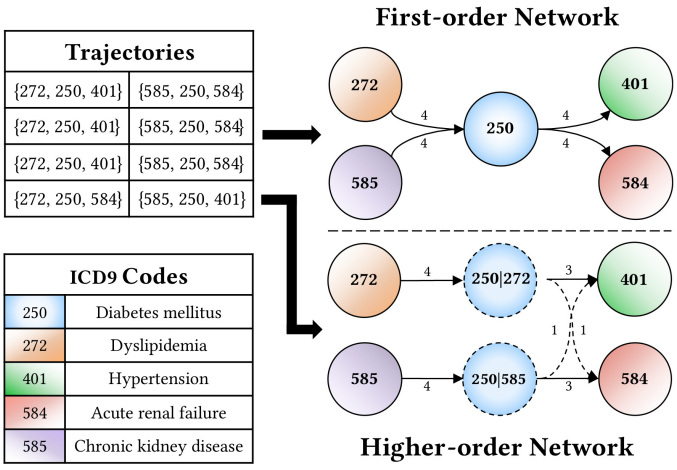


Fig. 1. A toy example of the difference between first-order and higher-order network representations of a set of disease trajectories. In the first-order network each node represents a single entity, but in a higher-order network a node can represent more than one entity. In this example, node 250 is split into 250|272 and 250|585 to better represent the underlying trajectories. Dashed lines around a node or edge indicate topological changes relative to the first-order network.

This Markov assumption has been challenged in studies of invasive species networks [2], human mobility [3], information networks [4], citation networks [5], trade relations [6], and more [7], [8]. Network scientists have recently introduced generalizable models that break the Markovian template [2], [4], [5]. These models, which we call **higher-order networks** (HONs), use directed and weighted networks to represent non-Markovian relationships between entities in a sequence. Experimental results from the aforementioned domains have demonstrated that HONs provide a higher-quality representation of the underlying sequence data than their first-order ancestors.

Figure 1 shows a toy example of the difference between a first-order network and a higher-order network. We begin with a set of eight **trajectories**, or sequences of entities. In this example, the entities are diagnostic codes from the ninth revision of the International Classification of Diseases (ICD9). We call these sequences **disease trajectories**, since they contain diagnosis codes and thus an estimate of disease

progression within a patient. In a first-order network, a node is created for each unique entity, and an edge is created between a pair of nodes if their corresponding entities are adjacent in a trajectory. This produces an underfit network. Consider a random walker at node 250: it has no knowledge of the higher-order pattern that three of the four trajectories that begin with code 272 end with 401, and only one with 584. Rather, it is only aware that the terminal node is 401 four out of eight times, and 584 for the other four.

A HON, on the other hand, allows a single node to represent multiple entities as a mechanism for capturing such dependencies. For example, in Figure 1, the entity 250 is represented by 2 distinct nodes: $250|272$ (read as 250 $given$ 272) and $250|585$. Both nodes represent the same base code, but with a different predecessor. This method of encoding memory into the nodes allows edges to be constructed and weighted in a way that better represents the non-Markovian relationships in the trajectories. We refer to the **order** of a node as the number of entities that it represents; e.g., 585 has an order of 1 and $250|585$ has an order of 2. The goal of a HON is to identify dependencies and encode them as higher-order nodes, without overfitting on nodes that do not have dependencies.

In this study, we used HONs to model comorbidities in type 2 diabetes mellitus (T2D) patients. A number of previous works have taken network-based approaches to modeling comorbidities [9]–[19], which Brunson et al. categorized according to which of the following problem(s) they are trying to solve [20]:

1) Identifying complex dependencies between diseases.
2) Analyzing the contribution of heterogeneous factors to co-occurrence of diseases.
3) Describing patient trajectories as they transition between various states over time.

This work focuses on the potential of HONs to address items 1 and 3 as they relate to comorbidities within T2D, which has a rich and complex comorboditiy space [21]. To demonstrate, we generated first-order and higher-order networks from disease trajectories extracted from the medical records of 913,475 T2D patients. We then examined differences between the networks, including changes in transition probabilities and results from random walking experiments, and conclude that HONs provide a more accurate representation of the underlying disease trajectories than first-order networks. Given this conclusion, we analyzed differences in PageRank scores and community structure at higher orders. Our main contribution is the novel application of the higher-order network framework to a set of T2D disease trajectories. Our experimental results demonstrate that HONs have the potential to greatly enrich the study of disease trajectories and comorbidities.

The rest of this paper proceeds as follows. First, we survey related work on T2D disease trajectories and HONs (Section II). Next, we detail our methods for collecting data, generating disease trajectories, and building higher-order networks (Section III). Then, we report our experimental results and discuss their implications (Section IV). Finally, we conclude and discuss opportunities for future work (Section V).

II. Related Work

A. Disease Networks and Trajectories

Researchers have previously created networks of disease trajectories by representing disease states as nodes and using edges to connect related states [9], [10]. Hanauer and Ramakrishnan demonstrated the importance of incorporating time and directionality into models of disease states [11]. Chen et al. used networks of cancer patient trajectories to predict metastasis [12], and other works took a similar approach to predicting sepsis mortality [13], cognitive decline in Alzheimer's patients [14], and chronic heart failure [15]. Glicksberg et al. utilized a disease network to identify differences in disease progression between races [16]. Jensen et al. generated a network of population-wide trajectories to identify a set of key diagnoses that are central to disease progression [17]. Their method revealed a cluster of diseases closely associated with the progression from T2D to insulin-dependent diabetes mellitus, and identified retinopathy as a key transition point to other comorbidities. Other work in the diabetes space includes that of Li et al., who utilized a patient similarity network to identify subgroups of T2D patients [18]. All of these works either analyze trajectories independently of a network structure or within a network that assumes the Markov property.

Some recent works addressed the shortcomings of such models by developing more sophisticated trajectories or network structures. Kannan et al. generated a multi-layered network and use directional dependencies between diagnoses to infer causality [22]. Thomas et al. combined diagnoses with demographic and clinical data, and used the resulting heterogeneous graph to predict the rate at which T2D patients will develop comorbidities [23]. However, to the best of our knowledge, none of these models addresses the challenge of treating more complex, non-Markovian interactions between disease states [19], [24]. Oh et al. generated trajectories of variable length [25], but analyzed them outside of a network context. This type of approach cannot take advantage of a network's natural ability to encode complex and interdependent relationships, or the number of proven and robust network analysis methods [1]. To the best of our knowledge, our work is the first to analyze higher-order disease trajectories within the context of a network structure.

B. Higher Order Network Models

Studies of invasive species networks [2], human mobility [3], information networks [4], citation networks [5], trade relations [6], and more [7], [8] share the common conclusion that first-order Markov models are insufficient for representing complex trajectories, but solve the problem via different methodological approaches. We use the higher-order network framework introduced by Xu et al., which encodes variable orders of dependencies in a single-layer network structure [2]. Key advantages of their approach over other models include:

1) It infers an optimal order for each trajectory, rather than inferring a fixed order at the model level and assuming all trajectories should be constrained to that

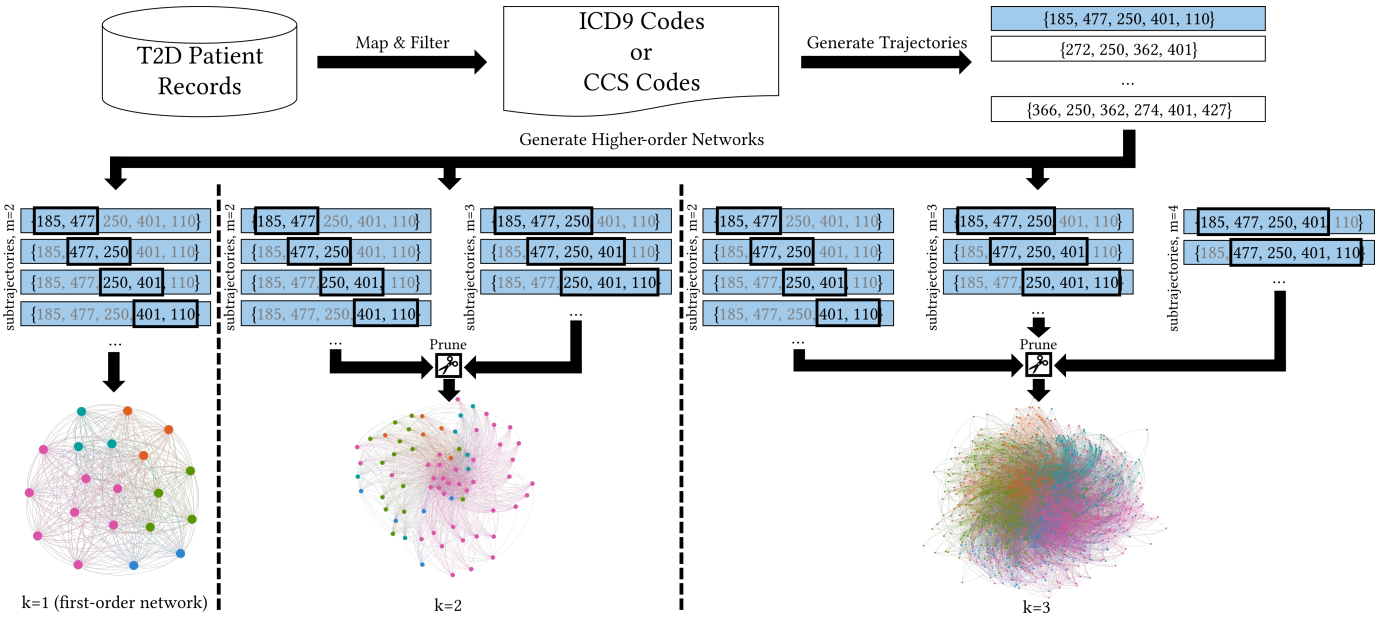


Fig. 2. A visualization of the process for generating higher-order networks of T2D disease trajectories, as described in Section III. We extracted a mapped and filtered set of diagnosis codes from our T2D patient data (Sections III-A and III-B), then used these codes to generate a set of trajectories (Section III-C), which we finally used to generate networks of different k (Sections III-D and III-E). The networks pictured are generated from a coarse mapping of all diagnosis codes to their ICD9 chapter and are not used in any experiments, but are included to illustrate the effects of higher values of k on the size of a network. Nodes are colored according to community structure.

order. This mitigates the exponential complexity and over/underfitting problems incurred by fixed order models, while seeking to create a balance between the expressivity and compactness of the network.

2) Trajectories of variable order share the same space, which means existing network analysis tools can be utilized without further modification. This is in contrast to approaches that use the original trajectories in combination with the first-order network to guide analysis [5], and approaches that generate multiple network layers and project characteristics between layers [4].

III. Methods

In this section we present our methods for generating higher-order networks of T2D disease trajectories. First, we describe our data (Section III-A). Next, we explain our method for mapping and filtering diagnosis codes (Section III-B). Then, we summarize our procedure for generating disease trajectories (Section III-C). Finally, we detail our framework for generating higher-order networks (Sections III-D and III-E). Figure 2 provides an overview of the entire process.

A. Data Description

We utilized a data set of 913,475 T2D patients created from the Indiana Network for Patient Care (INPC) database by the Indiana Biosciences Research Institute, Regenstrief Intistute, and other industrial partners. This study was approved by Indiana University's IRB (Exempt Protocol #: 1608149240 Computational Phenotyping for Type 2 Diabetes). The INPC links major healthcare providers across the state of Indiana and thus provides a rich source of patient data [26]. Included in the T2D cohort was any patient who, while being at least 18 years old, met at least one of the following criteria:

1) The patient was diagnosed with at least one T2D diagnosis code, as detailed in Table I.
2) The patient reported a laboratory glycated hemoglobin (HbA1C) test result of at least 6.5%.
3) The patient was prescribed at least one Medi-Span-defined anti-diabetes medication.

TABLE I
THE ICD9 AND ICD10 CODES USED TO IDENTIFY A T2D PATIENT.

ICD9	ICD10
249.*, 250.*, 357.2, 362.01, 362.01, 362.02, 362.03, 362.04, 362.05, 362.06, 362.07, 366.41	E10.*, E11.*

* is a wildcard symbol.

The patient records are multi-modal and contain information about demographics, procedures, prescriptions, and clinical laboratory results, but for this work we only used diagnosis codes. Despite limitations in their ability to accurately represent states of disease, diagnosis codes are still a rich source of historical information. In total we considered 228,652,937 diagnosis codes across 913,475 patients (an average of 250.31 codes per patient) between the years 1995 and 2018. 205,977,594 of these codes were valid under ICD9 and 20,248,072 were valid under ICD10. 2,427,271 were not valid under either classification and were likely the result of scribal error.

B. Mapping and Filtering Diagnosis Codes

We next applied two distinct mapping schemas to our data. This resulted in two sets of patient records, one defined by each schema.

1) ICD Major Codes: Because our longitudinal data is dominated by ICD9 codes (due to its timespan) and the difficulty of mapping these codes forward to the more granular ICD10 system, we preserved the ICD9 codes and attempted to map any ICD10 codes to their corresponding ICD9 code. Our procedure addressed the following cases:

- If the code was valid under ICD9, we preserved the section-level code (first 4 characters for E and V codes; first 3 characters for all others), which, for simplicity, we refer to as the major code. For example, codes 250.00 and 250.01 both mapped to 250.
- If the code was valid under ICD10, we used General Equivalence Mappings [27] to attempt a one-to-one mapping to an ICD9 code. If such a mapping existed, we replaced the ICD10 code with its ICD9 equivalent and preserved the major code as above. If there was more than one possible target, we did not attempt to identify the correct one and instead, to minimize bias, discarded the entire patient record. For example, ICD10 E11.42 can map to ICD9 250.60 or 357.2, so we discarded the record containing that code.
- If the code was not valid under ICD9 or ICD10, we discarded the code but kept the rest of the record.

We additionally discarded all E and V codes (supplementary), as well as major codes 780-799 (ill-defined conditions), which do not provide useful information about disease states and would have introduced noise into the trajectories. This process resulted in a set of records containing 908 distinct ICD major codes.

2) Clinical Classification Software (CCS) Codes: CCS was developed as a means of collapsing ICD9 and ICD10 codes into 283 "clinically meaningful categories" [28]. We performed this mapping for each valid ICD9 and ICD10 code and discarded any invalid codes. Because each ICD9 and ICD10 code maps to a single CCS code, we did not need to discard any patient records. After mapping, we discarded the CCS codes that were comprised entirely of E and V codes, as well as codes 780-799. We additionally discarded the CCS codes that were comprised of more than 50% of ICD codes that we discarded above: 196 (pregnancy otherwise unclassified), 218 (liveborn), 259 (residual codes; unclassified), and 663 (screening for mental health and substance abuse). This resulted in a set of records containing 243 unique CCS codes.

C. Generating Disease Trajectories

We next used the following procedure to generate two distinct sets of disease trajectories, one for each of the mapping schemas:

1) Sorted each patient's diagnoses in chronological order.
2) Split the trajectory into two separate trajectories at any gap of more than 1,095 days (3 years) between diagnoses. This is to minimize the risk of patients who have time lapses in their records due to missing data.
3) Removed all but the first occurrence of each code. In this study we simply assumed the first occurrence of a disease is the most informative about the progression between states, but acknowledge this assumption may not always hold.
4) Removed any remaining codiagnoses, which introduce ambiguity into the true sequence of states. We treat two or more codes as codiagnoses if they were recorded for the same patient on the same day. If this step removes any codes that have another occurrence that was removed in step 3, we re-insert the next occurrence at its appropriate chronological position in the trajectory. We then repeated this step until there were no codiagnoses in the trajectory.

The final set of ICD9 major code trajectories, which we call **ICD**, contained 1,005,931 trajectories, with 86,132 trajectories discarded due to having at least one ICD10 code that did not have a one-to-one backwards mapping. The set of CCS code trajectories, which we call **CCS**, contained 1,104,274 trajectories. We used both sets of t rajectories to build networks according to the methods described in the following sections.

D. Building First-order Networks

We formally define a **trajectory** as a sequence $S = (s_0, s_1, ..., s_n)$. Each $s_i \in S$ represents a discrete **entity**. Intuitively, a trajectory is a path with a entity s_0, a destination entity s_n, and intermediate entities $(s_1, ..., s_{n-1})$. Each pair of entities $(s_i, s_{i+1}) \in S$ represents a transition from s_i to s_{i+1}. In our case each entity is a diagnosis code, so the disease trajectory S estimates a sequence of disease states for a given patient. We also define $S' = (s'_0, s'_1, ..., s'_m)$ as a **subtrajectory** of S, denoted as $S' \sqsubseteq S$, if and only if all the entities in S' also appear in S in exactly the same order, i.e.

$$S' \sqsubseteq S \iff \exists j \leq n - m : \forall s'_i \in S', s'_i = s_{i+j}. \quad (1)$$

We construct a first-order network of a set of trajectories $\mathcal{S} = \{S_0, S_1, ..., S_N\}$ by creating a graph $G_1 = (V_1, E_1)$ where the set of nodes $V_1 = \{\bigcup \mathcal{S}\}$ is the set of unique entities across all trajectories, and the set of edges $E_1 = \{(u, v) : \exists S \in \mathcal{S}, (u, v) \sqsubseteq S\}$ is the set of adjacent pairs of entities across all trajectories. Edges are directed such that $(u, v) \neq (v, u)$, and weighted such that $w(u, v) \in \mathbb{Z}_0^+$ is the number of occurrences of the subtrajectory (u, v) across all trajectories. In the first-order network, each node represents a single entity, and each edge represents a Markovian transition from one entity to the next.

E. Building Higher-order Networks

A higher-order network encodes one or more entities in a single node. Let $G_k = (V'_k, E'_k)$ be a higher-order network, where $k \in \mathbb{Z}^+$ is the maximum order of the network, i.e. the maximum amount of history each node can encode. We first

TABLE II
SUMMARY OF THE HONS GENERATED FROM THE DISEASE TRAJECTORIES USING THE METHOD DESCRIBED IN SECTION III.

	N	$\overline{n}^a$	k	Build Time (s)	$\lvert\mathbf{V}\rvert$	$\lvert\mathbf{E}\rvert$	$\frac{\lvert\mathbf{E}\rvert}{\lvert\mathbf{V}\rvert}$	$\mathbf{H(G_k)}$ (bits)^b	# Clusters	Clustering Time (s)
CCS	1,104,274	12.20	1	49	242	50,851	210.13	6.643	2	1
			2	190	48,796	2,121,789	43.48	6.296	423	17
			3	533	2,315,331	11,244,163	4.86	3.299	115,081	1,469
			4	902	4,678,276	14,921,000	3.19	2.081	251,493	2,211
			5	1,093	4,700,681	14,958,602	3.18	**2.072**	253,220	2,239
ICD	1,005,931	11.81	1	44	908	312,410	344.06	7.509	3	1
			2	254	306,735	5,388,506	17.57	5.970	11,491	87
			3	486	4,569,841	13,601,271	2.98	2.042	266,059	1,682
			4	620	5,364,759	14,738,097	2.75	1.618	312,981	1,477
			5	796	5,375,585	14,752,165	2.74	**1.614**	313,727	1,505

[a]The mean trajectory length.
[b]Defined in Equation 6.

define an unpruned set of higher-order nodes V_k as the set of all subtrajectories of length $k+1$ or less:

$$V_k = \bigcup_{j=1}^{k+1} \{(s_{i-j+1}, ..., s_i) : \exists S \in \mathcal{S} \wedge \exists i, (s_{i-j+1}, ..., s_i) \sqsubseteq S\}. \quad (2)$$

Each node $u = (s_0, s_1, ..., s_m) \in V_k$ encodes a current entity s_m and a sequence of $m-1$ preceding entities. However, because the subtrajectories vary in length, V_k contains nodes of different orders. Intuitively, the order of a node is the length of the subtrajectory it represents, and a node's order increases with the number of previous entities it encodes. We use cardinality to denote order, such that $|u| = m+1$. We say that a node $u' = u \setminus \{s_0\}$ is the **lower-order counterpart** of u, since it represents the same subtrajectory but with the first (and oldest, assuming the sequence is chronological) entity truncated. Likewise, u is a **higher-order counterpart** to u' since it represents the same subtrajectory but predicated on one additional entity. A node can have up to $|V_1|$ higher-order counterparts, but only one lower-order counterpart.

We next define an unpruned set of edges E_k in a manner similar to E_1, but generalized to accommodate the fact that nodes represent variable-length subtrajectories rather than individual entities:

$$E_k = \{(u, v) : [(u, v) \in V_k] \wedge [\exists S \in \mathcal{S}, u + (v_m) \sqsubseteq S]\}, \quad (3)$$

where $+$ is the concatenation operator and v_m is the last entity in the subtrajectory v. Each edge in E_k is directed and weighted in the same way as in E_1. While E_1 is the set of adjacent pairs of entities across all trajectories, E_k is the set of adjacent pairs of subtrajectories (of length k or less) across all trajectories.

Two significant problems arise from this approach:

1) From a statistical perspective, some subtrajectories may be insignificant, so incorporating them would cause overfitting.
2) From a computational perspective, the cost of network generation and analysis grows exponentially with increasing values of k.

Following Xu et al. [2], we address these problems by pruning higher-order nodes from V_k if they do not contribute sufficiently to reducing the entropy of the network. We measure this by calculating the relative entropy (Kullback-Leibler divergence) of a higher-order node u with respect to its lower-order counterpart u' and comparing the result to a dynamic threshold. Recalling that in V_k a given node u is a tuple $(a_0, ..., a_m)$, and that $w(u, v)$ is the weight of the edge (u, v), we define a transition probability function $P(u \to v)$ as the probability that a random walker would move from node u to v, i.e.

$$P(u \to v) = \frac{w(u, v)}{outdeg(u)}, \quad (4)$$

where $outdeg(u) = \sum_{x \in V_k(u)} w(u, x)$ is the weighted out-degree of u. We then utilize Shannon entropy to measure the level of uncertainty for a given transition from u to v:

$$H(u \to v) = P(u \to v) log P(u \to v). \quad (5)$$

We can also measure the entropy rate of the entire network as a weighted sum of $H(u \to v)$ across all u and v:

$$H(G_k) = -\frac{\sum_{(u,v) \in E_k} w(u, v) H(u \to v)}{\sum_{u \in V_k} outdeg(u)}. \quad (6)$$

In constructing a higher-order representation, we want to reduce the entropy of the network. To determine which subtrajectories should be represented at higher orders, we utilize relative entropy. Given that each higher-order node u has exactly one lower-order counterpart u', we calculate the entropy of u as follows:

$$H(u) = \sum_{v \in \mathcal{N}(u)} P(u \to v) log \frac{P(u \to v)}{P(u' \to v')}, \quad (7)$$

where $\mathcal{N}(u)$ is the set of u's outgoing neighbors. Because u is of higher-order than u', we assume that higher relative entropy indicates that the u's additional historical information is valuable and is likely to reduce the total entropy of the network. However, relative entropy can be biased at higher orders, where edges are sparser and the transition probabilities are noisier. To mitigate this problem, we utilize the following threshold function from [29]:

$$f(u, \tau) = \frac{\tau \times (1 + |u|)}{log(1 + indeg(u))}, \quad (8)$$

TABLE III
A SAMPLE OF CCS CODES AND THEIR REPRESENTATION IN HIGHER-ORDER NETWORKS.

Code	Description	Frequency	# Nodes[a]					PageRank Scores				
			$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$
49	Diabetes mellitus without complication	513,546	1	192	27,138	139,051	141,160	.0378	.0371	.0452	**.0513**	**.0513**
53	Disorders of lipid metabolism	370,398	1	186	23,368	112,074	116,085	.0279	.0277	.0309	**.0357**	**.0357**
87	Retinal detachments, defects; retinopathy	65,535	1	221	13,543	23,070	23,110	**.0052**	.0049	.0047	.0047	.0047
98	Essential hypertension	435,269	1	185	25,443	129,597	133,143	.0323	.0380	**.0438**	.0437	.0437
103	Pulmonary heart disease	27,935	1	230	10,692	12,016	12,016	**.0024**	**.0024**	.0019	.0017	.0017
108	Congestive heart failure; nonhypertensive	90,107	1	211	16,626	34,406	34,412	.0075	**.0076**	.0074	.0072	.0072
109	Acute cerebrovascular disease	48,687	1	224	13,521	17,152	17,152	**.0040**	**.0040**	.0038	.0035	.0035
157	Acute renal failure	309,240	1	230	14,208	19,301	19,301	.0042	**.0043**	.0035	.0032	.0032
158	Chronic kidney disease	82,684	1	215	15,985	29,651	29,676	.0068	.0068	**.0072**	.0068	.0068
653	Delirium, dementia, cognitive disorders	247,913	1	224	12,926	16,320	16,320	**.0037**	**.0037**	.0033	.0030	.0030

[a] Indicates the number of nodes that represent the same CCS code, but with different predecessors.

where $\tau \in \mathbb{R}$ is a free parameter and $indeg(u) = \sum_{v \in V_k} w(v,u)$ is the weighted in-degree of u. The value of $f(u,\tau)$ increases with higher τ and node order $|u|$, but decreases for nodes that have a higher in-degree, which have more stable transition probability distributions.

Next, we create a pruned set of nodes V'_k and edges E'_k by keeping only the nodes from V_k whose relative entropy exceeds the threshold:

$$V'_k = \{u \in V_k : H(u) > f(u,\tau)\}, \tag{9a}$$

$$E'_k = \{(u,v) \in E_k : u,v \in V'_k\}. \tag{9b}$$

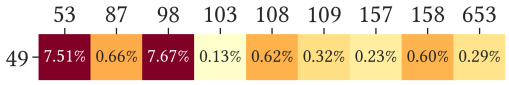
We found that this raw pruning of nodes and edges can isolate some higher-order nodes and increase the number of strongly-connected components. To avoid this, as a postprocessing step we identify any isolated higher-order nodes or components and merge them with their lower-order counterparts by combining their edge weights and removing the higher-order nodes.

The final network $G'_k = (V'_k, E'_k)$ thus addresses the statistical problem identified above by only keeping the trajectories that are expected to reduce the entropy of the network. Pruning has the additional computational benefit of significantly reducing the network size. However, it does not address the fact that larger values of k incur an exponentially increasing cost to enumerate and process the longer trajectories. We mitigate this problem by embedding all the subtrajectories of length $k+1$ into a tree structure during a preprocessing step, then performing pruning in-place on the tree structure. When each node is written to the edge list, it is converted from a tuple $(s_0, s_1, ..., s_n)$ to a delimited string "$s_n|s_{n-1}|...|s_0$" (as in Figure 1) for ease of interpretation. Further details and code for this implementation can be found online[1].

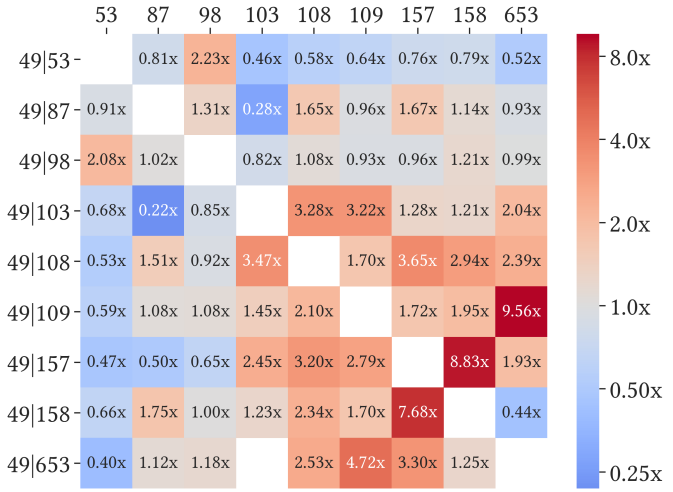
IV. EXPERIMENTAL RESULTS

In this section we present and discuss our experimental results. We first describe the networks and show examples of changes to the transition probability distribution between $k=1$ and $k=2$ (Sections IV-A and IV-B). We then use random walking to quantify the representational quality of each network with respect to the original trajectories (Section IV-C). Next, we use PageRank to show changes in node importance as the value of k increases (Section IV-D). Finally, we discuss differences in community structure detected in the higher-order networks (Section IV-E).

[1] https://github.com/sjkrieg/growhon



(a) Heatmap of first-order transition probabilities, as defined by Equation 4. Each column label represents a possible destination, and the value in cell is the probability (%) of moving from 49 to the corresponding destination at $k=1$. Darker shading corresponds to higher probabilities.



(b) Heatmap of second-order transition probabilities, relative to the first-order probabilities in Figure 3a, for CCS node 49. Each row label $49|x$ represents the second-order node 49 conditioned on the previous step x. Darker reds and blues indicate, respectively, greater increases and decreases to the transition probabilities. Blank cells represent non-existent transitions, i.e. edges that do not exist in the network at $k=2$.

Fig. 3. An example of the changes in transition probabilties from $k=1$ to $k=2$ for CCS node 49 (diabetes mellitus) to the sample nodes listed in Table III.

A. Network Generation

We generated HONs for both the ICD and CCS trajectories using $k = 1..5$ (we set $\tau = 1.0$ for all cases). This resulted in 10 total networks, which we describe in Table II. In all cases, the build time t was measured as process time on a

single Intel® Xeon® E5-2686 v4 2.30GHz CPU. For both CCS and ICD, the number of vertices and edges increased rapidly at lower orders but saturated at $k = 5$, which means that $k = 4$ was sufficient to capture most of the statistically significant subtrajectories. Additionally, the average degree $\frac{|E|}{|V|}$ decreased monotonically as k increased, which means that fewer distinct trajectories were detected at higher orders than would be predicted by inferring transitive connections at the first order. We also note that, as expected, the network entropy rate $H(G_k)$ decreased monotonically with higher values of k.

B. Changes to Transition Probabilities

As the order of a network increases, the transition probability between nodes changes. Figure 3 shows an example of transition probabilities between $k = 1$ and $k = 2$ for the sample of CCS codes described in Table III. Figure 3a shows the probabilities at $k = 1$, and Figure 3b shows the changes from $k = 1$ to $k = 2$. At $k = 2$, CCS code 49 (diabetes mellitus) was split into 184 nodes, each with a unique predecessor $(49|x)$. In some cases, the addition of a predecessor shifted the transition probabilities by an order of magnitude. For example, the transition $49|157 \rightarrow 158$ (T2D, given renal failure, to chronic kidney disease) was 8.83x more likely to occur than the corresponding first-order transition $49 \rightarrow 158$. The higher-order network is thus able to capture the known relationship between kidney diagnoses, even though in the original trajectories they are separated by a diagnosis of T2D. Also of note are changes to codes with high first-order baselines: e.g., the probability of $49|53 \rightarrow 98$ (T2D, given a lipid disorder, to hypertension) increases by 2.23x from an already-high 7.67% at the first order to 17.10% at the second order—another significant change that is not captured by the first-order network.

C. Using Random Walks to Measure Representational Quality

Random walks are foundational to many network methods, including PageRank, network embedding, and clustering methods. In these cases, a random walker's ability to simulate trajectories relies entirely on the representational quality of the network with respect to the underlying data. To quantify the effects of higher-order networks on random walking, we performed two experiments. In the first, we used a random walker to predict future disease states. In the second, we used a random walker to generate synthetic trajectories, which we then compared to the original set of trajectories.

1) Predicting Disease States: In the first random walking experiment, we predicted the final three entities in each trajectory. To do this, we first removed the last three codes of each trajectory and treated them as labels for the prediction task. We considered the rest of each trajectory as an observation, and used the set of observations to build a new set of networks (used only in this experiment) for $k = 1..5$. Then, for each trajectory, we instructed a random walker to follow the entire sequence of observed diagnoses before taking an additional 3 steps, which represent its predictions. At each step, the walker moved to a random neighbor based on probabilities

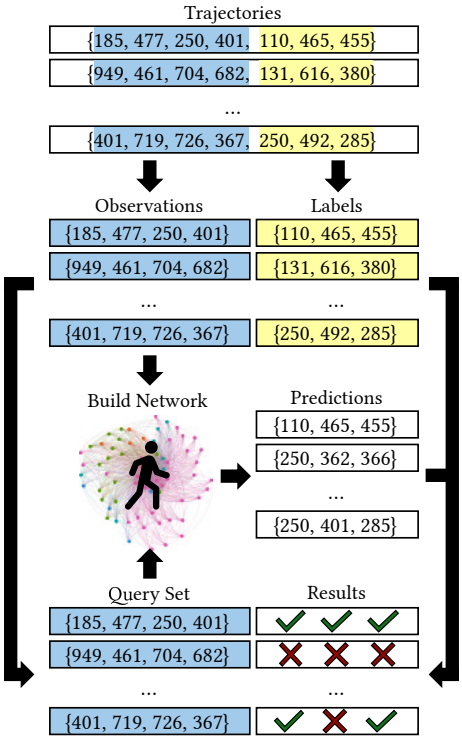


Fig. 4. Overview of the first random walking experiments used to quantify the representational quality of higher-order networks. Results of this experiment are shown in Figure 5.

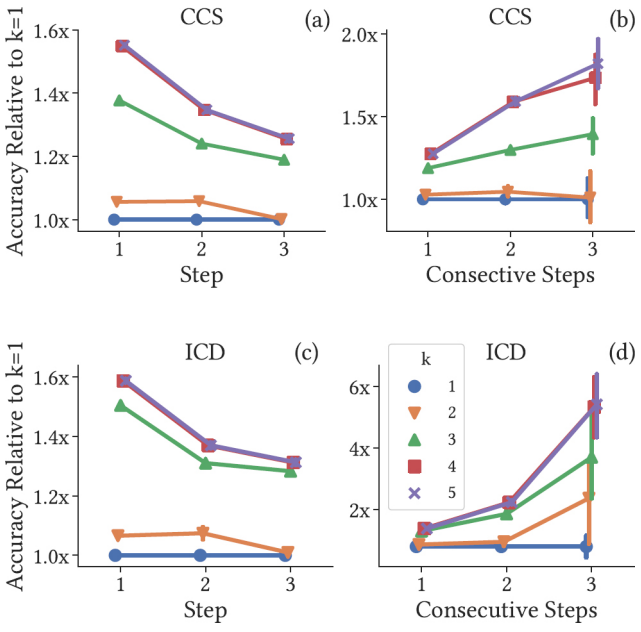


Fig. 5. Relative accuracy of random walk predictions on networks of various k. The y-axis of each plot represents accuracy relative to $k = 1$, with error bars representing standard deviation over 10 iterations. In (a) and (c), the x-axis represents each of the 3 steps predicted for each test trajectory. In (b) and (d), the x-axis represents the number of consecutive steps for each test trajectory. In this case, a prediction for steps 2 or 3 can only be correct if the predictions for all previous steps were correct.

TABLE IV
RESULTS OF USING RANDOM WALKS TO REPRODUCE THE ORIGINAL TRAJECTORIES.

		Jaccard Similarity				Weighted Jaccard Similarity			
	k	**n = 2**	**n = 3**	**n = 4**	**n = 5**	**n = 2**	**n = 3**	**n = 4**	**n = 5**
CCS	1	.9839	.3626	.0378	.0013	.3680	.2614	.0349	.0012
	2	.9867	.7163	.0508	.0020	.3678	.3203	.0464	.0019
	3	.9868	.7222	.4096	.0801	.3679	.3265	.2399	.0635
	4	.9865	.7218	.4096	.1790	.3680	.3265	.2400	.1311
	5	.9871	.7221	.4098	.1790	.3680	.3264	.2400	.1311
ICD	1	.9328	.1396	.0087	.0002	.3552	.1351	.0086	.0002
	2	.9329	.5508	.0287	.0025	.3551	.2762	.0259	.0022
	3	.9326	.5511	.3492	.1450	.3551	.2762	.2199	.1107
	4	.9328	.5510	.3493	.1816	.3552	.2763	.2199	.1346
	5	.9325	.5506	.3493	.1816	.3552	.2761	.2199	.1346

calculated using Equation 4. We considered a step correct if it matched the corresponding label, and incorrect otherwise. Figure 4 provides an overview of the process. Because our trajectories are defined to not include duplicate codes, we forced the walker to ignore neighbors that represent codes already observed in the trajectory. We also excluded any trajectories that contained fewer than eight total codes in order to minimize bias against higher values of k, which rely on historical information. In total, we tested 540,090 trajectories from CCS and 469,383 from ICD.

Figure 5 shows the results of 10 iterations on each network. In all cases the HONs outperformed the first-order network, and performance increased monotonically with higher k until saturating at $k = 5$. The y-axis for each plot represents accuracy relative to $k = 1$, rather than raw accuracy. This is because we are interested not in using random walks as a predictive model, but in quantifying the network's representational ability at higher values of k. We note that the raw accuracy values are quite low due to the difficulty of the task; for example, in CCS at $k = 1$, the accuracy of step 1 was only 1.9% across the 540,090 trajectories. In (a) and (c), the x-axis represents each independent step. In this case, the higher-order networks consistently outperformed the first-order networks, except for $k = 2$ on step 3. We suggest that the decrease in relative accuracy for $k > 1$ at steps 2 and 3 is because in a higher-order network, each of the walker's node choices are more consequential than in a first-order network. The first-order networks have high density, so if the walker makes an incorrect choice at step 1, it has a chance to correct its course at steps 2 or 3. In the higher-order networks, an incorrect choice directs the walker into a sparse neighborhood in which it is unlikely to correct itself. To test this hypothesis, we recomputed the results to count each step as correct only if the walker also correctly predicted all previous steps. These results are shown in (b) and (d) of Figure 5. As expected, for both CCS and ICD the increased accuracy compounds with each consecutive step. On ICD at $k = 4$ and $k = 5$, after 3 steps the accuracy is almost an order of magnitude higher than $k = 1$.

2) Reproducing the Original Trajectories: In the second random walking experiment, we tested the ability of each network to generate synthetic trajectories that accurately reproduce the original set of trajectories. To generate a synthetic trajectory, we selected a random starting node u, using the out-degree distribution to weigh the selection probabilities. In the HONs we only considered first-order nodes for u, since the first node in the trajectory should be free from historical information. At each step, the walker moved to a random neighbor based on probabilities calculated using Equation 4. We used this method to generate sets of synthetic trajectories on each of the networks described in Table II. Given a fixed trajectory length n, we enumerated all subtrajectories of length n from the set of real trajectories, then generated an equal number of synthetic samples. We evaluated the results using the Jaccard similarity coefficient, which measures the similarity of two sets $\mathcal{S}_n$ (real trajectories of length n) and $\mathcal{S}'_n$ (synthetic trajectories of length n) according to the following:

$$J(\mathcal{S}_n, \mathcal{S}'_n) = \frac{|\mathcal{S}_n \cap \mathcal{S}'_n|}{|\mathcal{S}_n \cup \mathcal{S}'_n|}. \tag{10}$$

Because Jaccard measures similarity only with respect to set membership and is thus agnostic to frequency, we additionally utilized weighted Jaccard to measure similarity with respect to the frequency distribution of each trajectory. To do this, we first created a set $\mathcal{T}_n = \mathcal{S}_n \cup \mathcal{S}'_n$ of all trajectories (real and synthetic) with length n. We then constructed two vectors, $X_n = \{x_n^0, x_n^1, ..., x_n^{|\mathcal{T}_n|-1}\}$ and $Y_n = \{y_n^0, y_n^1, ..., y_n^{|\mathcal{T}_n|-1}\}$, such that x_n^i and y_n^i correspond to the frequency that trajectory $i \in \mathcal{T}_n$ occurs in $\mathcal{S}_n$ and $\mathcal{S}'_n$, respectively. We then calculated weighted Jaccard as follows:

$$J_{\mathcal{W}}(X_n, Y_n) = \frac{\sum_{i=0}^{|\mathcal{T}_n|-1} min(x_n^i, y_n^i)}{\sum_{i=0}^{|\mathcal{T}_n|-1} max(x_n^i, y_n^i)}. \tag{11}$$

For each network with maximum order k and trajectory length n we generated 10 sets of synthetic trajectories and report the average similarity in Table IV. The standard deviation was less than 0.001 in all cases. For both CCS and ICD, the first-order network's performance decayed rapidly with increasing n, which reflects the exponentially increasing difficulty of the task. However, the HONs were much more resilient to increasing n: increasing the value of k consistently increased performance on longer trajectories while maintaining the same performance for shorter ones. Because a HON of a given k incorporates subtrajectories of up to length $k+1$,

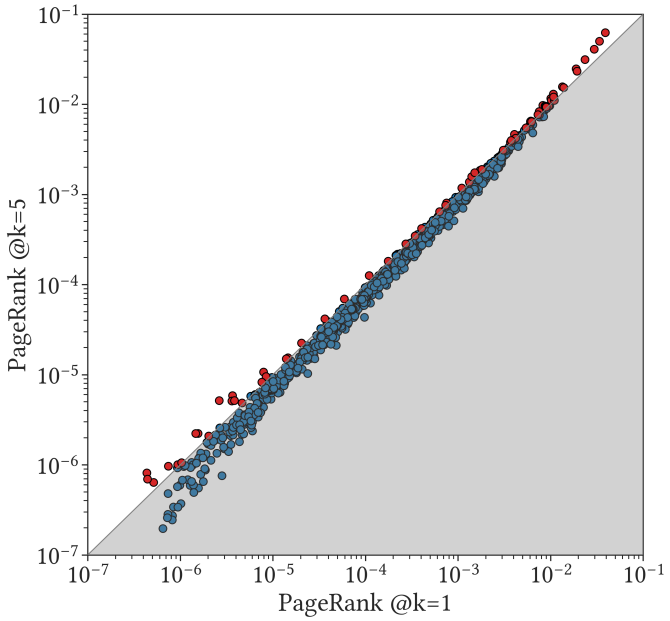


Fig. 6. Differences in PageRank scores for nodes in ICD between $k = 1$ (x-axis) and $k = 5$ (y-axis), in log-log scale. Nodes are colored blue if their score decreases and red if their score increases.

it is expected that the performance for a given value of n peaks when $k + 1 \geq n$. For the hardest task ($n = 5$), $k = 4$ and $k = 5$ outperformed $k = 1$ by two and three orders of magnitude for CCS and ICD, respectively. This demonstrates that random walks longer than two nodes are much more likely to correspond to the distribution of the underlying data when performed on higher-order networks than on their first-order counterparts.

D. Using PageRank to Measure Node Importance

PageRank was developed to measure the relative importance of web pages [30], and is used similarly in other networks to quantify node importance. PageRank scores are calculated by assessing the importance of a node as a function of both its degree and the importance of its neighbors, so the results depend heavily on the network structure. To investigate the effects of higher-order representation on PageRank, we calculated PageRank scores in each of our networks and compared the change in each entity's score at higher values of k. In each HON, we calculated an entity's PageRank score as the sum of the scores of all its higher-order splits. Because the out-degree of a given entity often increases with higher k, we calculated PageRank on a set of HONs constructed with an additional postprocessing step: for each preserved higher-order node, we decreased the out-degree of its lower-order counterparts by the higher-order node's out-degree. This ensures that each entity has the same representation (with respect to total out-degree) at any value of k, and thus avoids biasing the PageRank scores toward entities whose out-degree increases more at higher k.

Table III reports the scores for a sample of CCS codes across all k, and Figure 6 depicts the changes to scores in ICD between $k = 1$ and $k = 5$. For ICD, most nodes (836) decrease in importance at higher-orders, while a much smaller number (72) increase in importance. These changes can be used to understand the significance of a node as a transition point between other nodes when we consider longer trajectories. For example, Jensen et al. identify retinopathy as a gatekeeper to other diabetic comorbidities because of the number of other diagnoses it precedes in their disease trajectory network [17]. In our case, retinopathy (CCS code 87, ICD9 code 362) has a PageRank score of 0.0052 in both CCS and ICD at $k = 1$. At $k = 5$, its score decreases to 0.0047 in CCS and 0.0051 in ICD. While in many cases retinopathy connects other comorbidities, its significance decreases as we consider more history. By contrast, the score of CCS code 211 (disorders of soft and connective tissue) increases from 0.0282 at $k = 1$ to 0.0371 at $k = 5$, suggesting that such a diagnosis could be a more important indicator of progression to other comorbid conditions.

E. Detecting Communities of Related Diseases

Clustering, or community detection, algorithms are used to identify groups of densely connected nodes within a network [31]. Within a higher-order network, which uses nodes to represent trajectories rather than single entities, clustering offers even more analytical potential than in a first-order network. We applied the Infomap algorithm to cluster each network according the map equation, which seeks to minimize the number of bits required to accurately describe the flow of information, measured by PageRank score, between communities [32]. Table II shows the number of clusters found by Infomap for each network. For ICD at $k = 1$, the nodes were trivially divided into a dominant cluster with 855 nodes (responsible for 99% of information flow), and two smaller clusters: one with 44 nodes (0.1% of flow) from ICD codes 630-679 (complications of pregnancy, childbirth, and the puerperium) and 760-779 (certain conditions originating in the perinatal period), and the other cluster with 9 nodes (0.01% of flow) from ICD codes 630-640 (complications of pregnancy). Thus, the only major difference revealed by the flow of information at $k = 1$ was between pregnancy and non-pregnancy-related codes. As k increased, so did the number of clusters and the complexity of their distinguishing features. Figure 7 shows a summary of the diseases represented in the most influential cluster at each value of k for ICD. There were clear differences in node representation, especially at $k > 2$, where the flow is dominated by endocrine diseases (codes 240-279). At $k = 3$, the rest of the cluster was dominated almost entirely by diseases of the circulatory system (codes 390-459). At $k = 4$ and 5 we observed high representation from other codes such as 401 (essential hypertension), 477 (allergic rhinitis), 724 (back disorders), and 733 (bone disorders). These structural relationships were important for describing flow in the higher-order networks, but were not distinguishable in the first-order network. Further study of these higher-order clusters could greatly enrich our understanding of complex relationships between diseases and thus further one of network

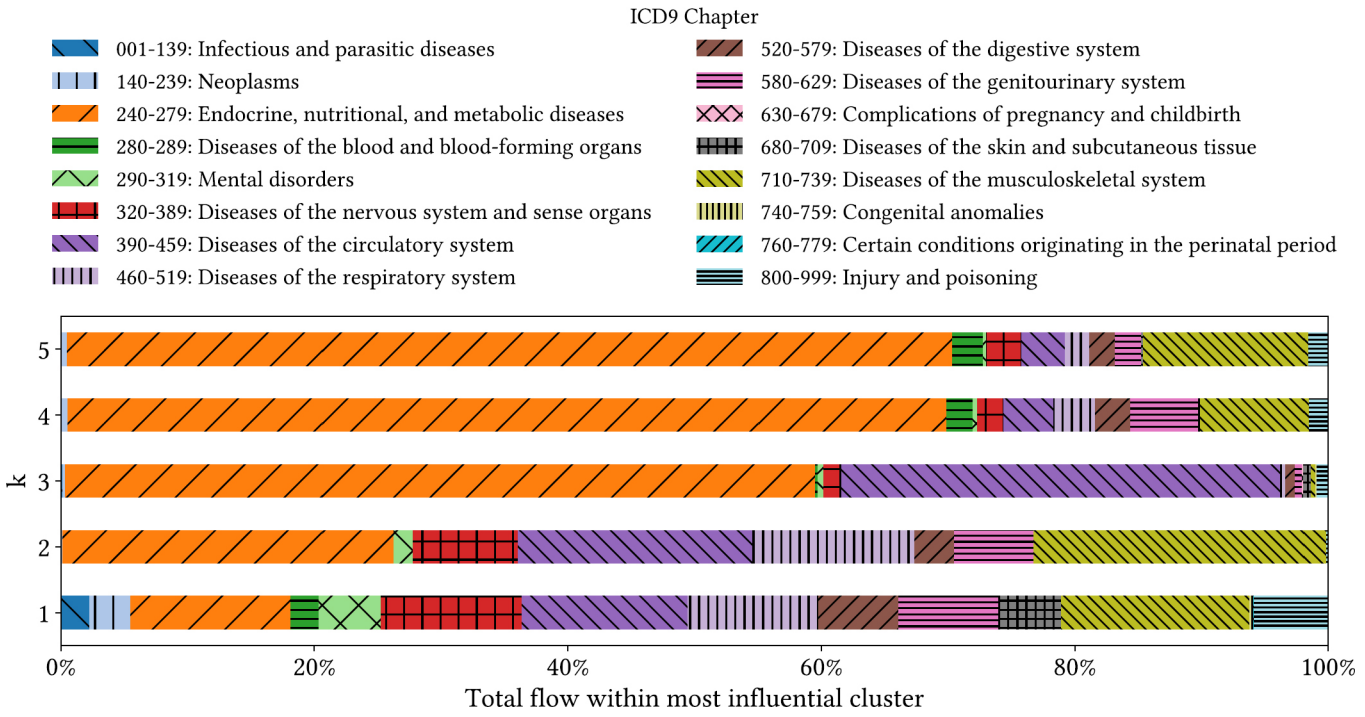


Fig. 7. Summary of differences in the most influential cluster at each value of k for ICD, as discussed in Section IV-E. The most influential cluster is the one whose nodes have the highest combined PageRank score. In this figure, representation is defined as the contribution of each node to the cluster's combined PageRank score. Nodes are additionally grouped by ICD9 chapter.

science's main contributions to the study of comorbidities [20].

V. Conclusion

In this study, we demonstrated the ability of higher-order networks to model complex relationships between cormorbid diseases more effectively than traditional first-order networks. Because higher-order networks can use a single node to represent one or more entities, they are able to capture more complex dependencies from a set of disease trajectories than first-order networks. To show this, we first extracted diagnosis codes from the medical records of 913,475 type 2 diabetes patients and used them to generate sets of disease trajectories. We then used these trajectories to generate a set of higher-order networks of various k, where k represents the maximum order, or amount of history encoded by each node in the network. We showed that increasing k reduces entropy in the network, and discussed changes to the transition probability distribution for a sample of disease states. We then demonstrated that random walks on higher-order networks are better able to predict additional steps and reproduce the original set of trajectories. We next analyzed PageRank scores for nodes across all networks and discussed implications for nodes that increase or decrease in importance at higher orders. Finally, we clustered the networks and discussed that the ability of higher-order networks to encode complex relationships can produce more informative communities.

From the results of this study, we conclude that the higher-order network framework, which provides a solution to the Markovian limitations of other network models, is important for the continued research of complex dependencies between diseases and their progression over time. Future work could include incorporating heterogeneous information like medication history, laboratory results, and procedure history into higher-order networks, which are currently limited to representing sequences of the simple type used in this study. Additionally, more sophisticated methods for representing the disease trajectories, especially in regards to time between diagnoses, could enable even higher-quality network representations.

Acknowledgments

The authors wish to thank Dr. Titus K. Schleyer, Regenstrief Institute, and Indiana University School of Medicine for their assistance with access to the Indiana Network of Patient Care (INPC) data.

References

[1] F. Conte, G. Fiscon, V. Licursi, D. Bizzarri, T. D'Antò, L. Farina, and P. Paci, "A paradigm shift in medicine: a comprehensive review of network-based approaches," *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms*, p. 194416, 2019.

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

[3] I. Scholtes, N. Wider, R. Pfitzner, A. Garas, C. J. Tessone, and F. Schweitzer, "Causality-driven slow-down and speed-up of diffusion in non-markovian temporal networks," *Nature communications*, vol. 5, p. 5024, 2014.

[4] I. Scholtes, "When is a network a network?: Multi-order graphical model selection in pathways and temporal networks," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 1037–1046.

[5] 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, p. 4630, 2014.

[6] A. Koher, H. H. Lentz, P. Hövel, and I. M. Sokolov, "Infections on temporal networks—a matrix-based approach," *PloS one*, vol. 11, no. 4, p. e0151209, 2016.

[7] T. P. Peixoto and M. Rosvall, "Modelling sequences and temporal networks with dynamic community structures," *Nature communications*, vol. 8, no. 1, p. 582, 2017.

[8] 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.

[9] C. A. Hidalgo, N. Blumm, A.-L. Barabási, and N. A. Christakis, "A dynamic network approach for the study of human phenotypes," *PLoS computational biology*, vol. 5, no. 4, p. e1000353, 2009.

[10] K. Steinhaeuser and N. V. Chawla, "A network-based approach to understanding and predicting diseases," in *Social computing and behavioral modeling*. Springer, 2009, pp. 1–8.

[11] D. A. Hanauer and N. Ramakrishnan, "Modeling temporal relationships in large scale clinical associations," *Journal of the American Medical Informatics Association*, vol. 20, no. 2, pp. 332–341, 2012.

[12] L. Chen, N. Blumm, N. Christakis, A. Barabasi, and T. S. Deisboeck, "Cancer metastasis networks and the prediction of progression patterns," *British journal of cancer*, vol. 101, no. 5, p. 749, 2009.

[13] M. K. Beck, A. B. Jensen, A. B. Nielsen, A. Perner, P. L. Moseley, and S. Brunak, "Diagnosis trajectories of prior multi-morbidity predict sepsis mortality," *Scientific reports*, vol. 6, p. 36624, 2016.

[14] P. A. Wilkosz, H. J. Seltman, B. Devlin, E. A. Weamer, O. L. Lopez, S. T. DeKosky, and R. A. Sweet, "Trajectories of cognitive decline in alzheimer's disease," *International Psychogeriatrics*, vol. 22, no. 2, pp. 281–290, 2010.

[15] S. Nagrecha, P. B. Thomas, K. Feldman, and N. V. Chawla, "Predicting chronic heart failure using diagnoses graphs," in *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*. Springer, 2017, pp. 295–312.

[16] B. S. Glicksberg, L. Li, M. A. Badgeley, K. Shameer, R. Kosoy, N. D. Beckmann, N. Pho, J. Hakenberg, M. Ma, K. L. Ayers *et al.*, "Comparative analyses of population-scale phenomic data in electronic medical records reveal race-specific disease networks," *Bioinformatics*, vol. 32, no. 12, pp. i101–i110, 2016.

[17] 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.

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

[20] J. C. Brunson and R. C. Laubenbacher, "Applications of network analysis to routinely collected health care data: a systematic review," *Journal of the American Medical Informatics Association*, vol. 25, no. 2, pp. 210–221, 2017.

[21] H. D. Nickerson and S. Dutta, "Diabetic complications: current challenges and opportunities," *Journal of cardiovascular translational research*, vol. 5, no. 4, pp. 375–379, 2012.

[22] 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.

[23] 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.

[24] 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.

[25] 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.

[26] J. M. Overhage, W. Tierney, and C. McDonald, "Design and implementation of the indianapolis network for patient care and research." *Bulletin of the Medical Library Association*, vol. 83, no. 1, p. 48, 1995.

[27] R. R. Butler, "Icd-10 general equivalence mappings: Bridging the translation gap from icd-9," *Journal of AHIMA*, vol. 78, no. 9, pp. 84–86, 2007.

[28] Healthcare Cost and Utilization Project (HCUP), "Clinical classification software," Agency for Healthcare Research and Quality, Mar. 2017, accessed Jan. 8, 2020. [Online]. Available: http://www.hcup-us.ahrq.gov/toolssoftware/ccs/ccs.jsp

[29] J. Xu, M. Saebi, B. Ribeiro, L. M. Kaplan, and N. V. Chawla, "Detecting anomalies in sequential data with higher-order networks," *arXiv preprint arXiv:1712.09658*, 2017.

[30] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.

[31] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[32] M. Rosvall, D. Axelsson, and C. T. Bergstrom, "The map equation," *The European Physical Journal Special Topics*, vol. 178, no. 1, pp. 13–23, 2009.