Geometric Deep Learning for Healthcare Applications

Gaurang A. Karwande

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Ismini Lourentzou, Co-chair
Creed Jones, Co-chair
Lynn Abbott

April 19, 2023
Blacksburg, Virginia

Keywords: Graph Neural Networks, Medical Imaging, Causal Structure Learning, Bayesian Networks

Geometric Deep Learning for Healthcare Applications

Gaurang A. Karwande

(ABSTRACT)

This thesis explores the application of Graph Neural Networks (GNNs), a subset of Geometric Deep Learning methods, for medical image analysis and causal structure learning. Tracking the progression of pathologies in chest radiography poses several challenges in anatomical motion estimation and image registration as this task requires spatially aligning the sequential X-rays and modelling temporal dynamics in change detection. The first part of this thesis proposes a novel approach for change detection in sequential Chest X-ray (CXR) scans using GNNs. The proposed model `CheXRelNet` utilizes local and global information in CXRs by incorporating intra-image and inter-image anatomical information and showcases an increased downstream performance for predicting the change direction for a pair of CXRs. The second part of the thesis focuses on using GNNs for causal structure learning. The proposed method introduces the concept of intervention on graphs and attempts to relate belief propagation in Bayesian Networks (BN) to message passing in GNNs. Specifically, the proposed method leverages the downstream prediction accuracy of a GNN-based model to infer the correctness of Directed Acyclic Graph (DAG) structures given observational data. Our experimental results do not reveal any correlation between the downstream prediction accuracy of GNNs and structural correctness and hence indicate the harms of directly relating message passing in GNNs to belief propagation in BNs. Overall, this thesis demonstrates the potential of GNNs in medical image analysis and highlights the challenges and limitations of applying GNNs to causal structure learning.

Geometric Deep Learning for Healthcare Applications

Gaurang A. Karwande

(GENERAL AUDIENCE ABSTRACT)

Graphs are a powerful way to represent different real-world data such as interactions between patient observations, co-morbidities, treatments, and relationships between different parts of the human anatomy. They are also a simple and intuitive way of representing cause-and-effect relationships between related entities. Graph Neural Networks (GNNs) are neural networks that model such graph-structured data. In this thesis, we explore the applicability of GNNs in analyzing chest radiography and in learning causal relationships. In the first part of this thesis, we propose a method for monitoring disease progression over time in sequential chest X-rays (CXRs). This proposed model `CheXRelNet` focuses on the interactions within different regions of a CXR and temporal interactions between the same region compared in two CXRs taken at different times for a given patient and accurately predicts the disease progression trend. In the second part of the thesis, we explore if GNNs can be used for identifying causal relationships between covariates. We design a method that uses GNNs for ranking different graph structures based on how well the structures explain the observed data.

Acknowledgments

First and foremost, I am deeply grateful to my advisor, Dr. Ismini Lourentzou, for her guidance, support, and expertise throughout my research. Her invaluable insights, critical feedback, and encouragement have been instrumental in shaping my work. I am also grateful to my committee members, Dr. Creed Jones, and Dr. Lynn Abott. I would also like to express my gratitude to the faculty members at Virginia Tech for providing me with a stimulating and challenging academic environment. I am forever indebted to my parents Nilima and Ajit Karwande for constantly motivating and enabling me to dream big. I am also grateful to my roommates, friends and family for their unwavering support and encouragement throughout my academic journey.

Contents

List of Figures

List of Tables

List of Abbreviations

AI	Artificial Intelligence
AIC	Akaike Information Criterion
AUC	Area Under the Cuve
BIC	Bayesian Information Criterion
BN	Bayesian Network
BP	Belief Propagation
CXR	Chest X-ray
DAG	Directed Acyclic Graph
DL	Deep Learning
EHR	Electronic Health Records
FPN	Feature Pyramid Network
GAT	Graph Attention Network
GNN	Graph Neural Networks
ML	Machine Learning
MPN	Message Passing Network
NLP	Natural Language Processing

PCH Pearl's Causal Hierarchy

RoI Region of Interest

SCM Structural Causal Model

SD Standard Deviation

SEM Structural Equation Model

SOTA State of the Art

Chapter 1

Introduction

The healthcare industry has traditionally relied heavily on manual practices, with clinicians and healthcare providers manually collecting and analyzing patient data, and making decisions based on their own expertise and judgment. However, this approach has become increasingly unsustainable, as the demand for healthcare services continues to rise, while the number of clinicians and healthcare providers remains limited [89]. This has resulted in an increased workload for clinicians, who are often overburdened and unable to keep up with the demands of patient care. At the same time, there is a shortage of doctors and other healthcare professionals, which further exacerbates the problem. The need for automation in healthcare has never been more pressing, as it has the potential to help healthcare providers to manage their workload more efficiently, and reduce the risk of errors and inconsistencies in patient care. Artificial (AI) Intelligence and Machine Learning can help facilitate this.

Over the last decade, the healthcare industry has seen a digital revolution. Today, about 30% of the world's data volume is generated by the healthcare sector, and it is estimated to reach 36% by 2025 [3]. The widespread implementation of Electronic Health Records (EHRs) has enabled relatively easy access to large amounts of patient data across varied settings and demographics. EHRs can include a range of different types of data, including structured data such as patient demographics, diagnoses, and medication records, as well as unstructured data such as clinical notes, imaging data, and other types of multimedia data. The multimodal nature of EHRs is an important aspect of their utility in healthcare,

as it enables healthcare providers to integrate and analyze different types of data to gain a more comprehensive understanding of patient health. The volume of data made available by EHRs provides significant opportunities for leveraging machine learning and other advanced analytical techniques to improve healthcare outcomes. By developing models that can integrate and analyze different types of data, researchers and healthcare providers can develop more accurate predictive models for patient outcomes, identify patients at risk for certain conditions or complications, and develop personalized treatment plans that take into account the full range of patient health data [39, 49, 101].

While most AI techniques are data-intensive, infusion of domain knowledge is essential for accurate and robust predictions [2]. This is especially true in sciences such as physics, chemistry, biology, and economics, where our knowledge about natural processes can help establish strong priors during modeling. Domain Knowledge is particularly important in healthcare, where patient data is often multimodal and complex, and the domain knowledge-based-priors help clinicians identify important relationships between different entities. Models that can learn from the complex relationships between different interacting entities such as patients, radiological scans, diseases, and medical procedures while incorporating domain knowledge can lead to better performance and more accurate predictions [7]. Graph Neural Networks (GNNs) can capture complex relationships between such interacting entities. GNNs are a class of neural networks designed to operate on graphs. Owing to the incredible flexibility of graphs compared to more rigidly structured forms of information like images, videos, and text, many real-world tasks can be represented as graphs.

Deep learning (DL) has made significant advances in biomedical image analysis and assisting radiologists with diagnosis. DL algorithms are being used for classification, prediction, segmentation, and reconstruction tasks with medical images [91]. However, traditional DL techniques such as Convolutional Neural Networks (CNNs) by themselves are not able to

exploit the structural information abstracted within the human body. Imaging techniques such as X-rays, CT scans, and MRIs reveal a lot about not just the specific body parts being imaged, but also the exact position and function of the regions within the entire human anatomy. GNNs inherently work with such structural information and thus can aid in the development of clinically intuitive models. In Chapter 3, we propose a graph-based approach for interpreting Chest X-rays (CXRs) that tries to mimic the workflow of an expert radiologist. This chapter is derived from my recently published work [48] in collaboration with students and faculty at Virginia Tech, and researchers from IBM, the University of British Columbia, and the Massachusetts Institute of Technology.

While existing DL methods can analyse massive datasets of medical images and/or text, most of them work by identifying the correlation between different entities within the data. In healthcare, there are often many factors that can affect a patient's health outcomes, and it can be difficult to determine which factors are causal and which are merely correlations. Being able to identify cause-and-effect relationships within vast volumes of clinical data has the potential to improve patient outcomes, reduce healthcare costs, and advance medical research. Therefore, causal reasoning is essential in healthcare settings. A famous quote by Judea Pearl, a pioneer in Bayesian Networks (BN) and causality theory, is - "As X-rays are to the surgeon, graphs are to causation" [75]. Hence, graphs, or more precisely Directed Acyclic Graphs (DAGs), are core to causality and are a simple way of representing causal relationships among entities. In real-world settings, the complete causal model of a system is rarely available and one needs to learn the most fitting causal structure given the observed data. In Chapter 4 we explore the use of GNNs for evaluating causal structures or DAGs.

In this thesis, we ask the following research questions:

1. Can we use GNNs to exploit the anatomical information abstracted within medical images for accurate and robust monitoring of CXRs?

2. Can we use DL methods with sufficient inductive bias such GNNs for causal discovery?

3. Can message passing in GNNs be related to Belief Propagation in BNs?

Chapter 2

Review of Literature

2.1 Deep Learning and Medical Imaging

Medical imaging is an important tool in healthcare for the diagnosis, treatment planning, and monitoring of various medical conditions. There has been significant progress in applying DL techniques to radiological tasks such as diagnosis, Region of Interest (RoI) detection and segmentation, and image enhancement [6, 35]. These cover variety of radiological modalities such as X-rays, MRIs, CT scans, *etc.* Convolutional Neural Networks (CNNs) have been shown to achieve dermatologist-level accuracy in the classification of skin cancer images. [25]. Similarly, SOTA results have also been produced for breast cancer detection as well [83]. In 2016, a study by Lakhani and Sundaram [55] demonstrated that a deep learning algorithm can detect pulmonary nodules in CT scans with an Area Under the Curve (AUC) of 0.99. Deep learning methods have also been applied to Chest X-rays for enhancing pulmonary nodule detection [68]. Image segmentation is a dense classification task of classifying every pixel within the image. Medical image segmentation is a critical clinical task in various clinical applications such as disease diagnosis, treatment planning, and monitoring. U-Net is a popular architecture for medical image segmentation, which combines the encoder-decoder architecture with skip connections [78], has shown excellent results in segmenting various structures such as the liver, brain, and blood vessels [18].

Despite these advances, complex reasoning tasks remain fairly unexplored. For example,

monitoring changes in longitudinal disease progression has received limited attention from the research community. Previous work tackles change between longitudinal patient visits and evaluates the severity of diseases at each time point on a continuous scope on osteoarthritis in knee radiographs and retinopathy of prematurity in retinal photographs [59]. Other works target longitudinal disease tracking and outcome prediction severity for COVID-19 pulmonary diseases [58], by calculating a severity score for pulmonary X-rays via computing the Euclidean distance between each of the normal images and the image of interest. In addition, geometric correlation maps have been used to study the CXR longitudinal change detection problem [70], in which feature maps are extracted from CXR pairs and their matching scores are used to generate a geometric correlation map that can detect map-specific patterns showing lesion change. Most of these works rely on global image information. To the best of our knowledge, no prior work considers capturing correlations among anatomical regions and findings when modeling change between medical examinations. Yet, localizing pathologies to anatomy is critical for the radiologists' reasoning and reporting process, where correlations between image findings and anatomical regions can help narrow down potential diagnoses.

2.2 Graph Neural Networks

Graph Neural Networks (GNNs) operate on graph-structured data. Message-passing networks (MPNs) are a type of GNN that explicitly define message-passing operations between nodes in a graph. MPNs can be used for a variety of tasks, including graph classification, node classification, and link prediction [24, 52, 110]. The message-passing mechanism in MPNs involves passing messages between neighboring nodes in the graph to update their hidden states. This is done through a series of transformation functions that combine the

previous hidden state of each node with the messages received from its neighbors. This process is repeated iteratively until convergence is achieved or a fixed number of iterations is reached [12]. One advantage of MPNs is their ability to capture both local and global information from the graph structure. By iteratively passing messages between nodes, the network can incorporate information from the entire graph, while still retaining information about local neighborhoods. This makes MPNs well-suited for tasks such as molecule property prediction, where the properties of a molecule are dependent on the interactions between its atoms [29]. Another advantage of MPNs is their ability to handle graphs of varying sizes and structures. This makes them ideal for applications in which the underlying graph structure is not fully known or is subject to change, such as social network analysis or medical diagnosis [37]. One example of an MPN is the graph convolutional network (GCN), which uses a simple form of message passing based on graph convolution [52]. GCNs have been applied successfully to a wide range of tasks, including social network analysis, protein structure prediction, and recommender systems [111]. Another example of an MPN is the graph attention network (GAT), which uses attention mechanisms to weigh the messages sent between nodes [94]. GATs have been shown to achieve state-of-the-art results on several graph-based tasks, including node classification and link prediction [34, 95].

This flexibility of the message-passing framework in GNNs has shown great promise in medical imaging where graphs can be used to represent relationships between regions of interest. In a study by Parisot et al. [73], a GNN-based framework was proposed for the diagnosis of Alzheimer's disease using graph representations of brain networks. In another study by Zhao et al., a GNN-based framework was proposed for airway segmentation in chest CTs [28]. The framework incorporated information based on node connectivity in addition to local features, and hence improve upon the segmentation decisions. GNNs have also been used for segmentation of retinal vessel [27, 84], intracranial arteries [28], and cerebral cortex

[21, 33]. We review more applications of GNNs pertinent to causal inference and structure learning in Section 2.4.

2.3 Causal Inference and Structural Causal Models

Causal inference is a branch of statistics and machine learning that seeks to understand the causal relationships between different variables in a system. To make causal claims, we need to go beyond simple correlations and consider the underlying causal mechanisms that link different variables together. Causation is a subtle concept that cannot be fully described in the language of Boolean logic [80] or that of probabilistic inference; it requires the additional notion of intervention [23, 76]. A variable X is said to cause another variable Y if when all confounders are adjusted, an intervention in X results in a change in Y [75].

Compared to causal models, statistical models are a rather superficial description of reality as they are only required to model associations [80]. For a given set of input examples X and target labels Y, we are interested in approximating $P(Y|X)$ to answer questions such as "What is the probability that a particular image is of a dog or a cat?" or "What is the probability that the given CXR indicates pneumonia?". Subject to suitable assumptions, these questions can be answered by observing a large amount of independent and identically distributed (i.i.d) data from $P(X, Y)$ [93]. The predictions of a statistical model are only accurate within identical experimental conditions. Performing an intervention results in a change in the data distribution, which may lead to random and inaccurate predictions [75, 87]. However, interventions are quite relevant in real-world settings, where changes in data distributions happen naturally. For example, a drug discovery model trained to find a molecular compound with the most affinity to a prevalent virus. Over time, a mutation within the virus can produce a whole new strain on which this particular compound is no

Table 2.1: Pearl's Causal Hierarchy (PCH). Questions from level 1 can only be answered if information from level 1 or higher is available [76].

Level (Symbol)	**Typical Activity**	**Typical Questions**	**Examples**
1. Association $P(y\|x)$	Seeing	What is? How would seeing X modify my belief in Y?	What does a symptom tell me about a disease?
2. Intervention $P(y\|do(x), z)$	Doing Intervening	What if? What if I do X?	What if I take morphine, will my pain reduce?
3. Counterfactuals $P(y_x\|x', y')$	Imagining Retrospection	Why? Was it X that caused Y? What if I had acted differently?	Was it morphine that reduced my pain? What if I had not taken morphine?

longer effective. Hence, answering questions such as "What is the probability that the drug is effective if a particular mutation happens results in a modified virus strain?". Mathematically this can be represented by the *do* operator introduced by Pearl [76]. $P(Y|do(X = x), z)$ is the probability of Y if we intervened in the data generating process by artificially forcing the variable X to take value x, but otherwise simulating the rest of the variables according to the original data generating process [76].

The three rungs or layers of Causal Inference were introduced by Pearl [76]. This is a hierarchy of three types of problems with increasing difficulty. The first two levels are association and intervention, which we described above. The third and top level is counterfactuals which involve retrospective reasoning. Mathematically, this is given by $P(Y'_{x'}|(x, y))$, that is the probability of Y had the event x' taken place instead of event x having known the probability of the actual event and outcomes x and y respectively. The three rungs of causal inference are illustrated in Table 2.1.

Causality is represented mathematically with Structural Causal Models (SCMs). An SCM $\mathcal{M}$ is a 4-tuple $< \mathcal{U}, \mathcal{V}, P(\mathcal{U}), \mathcal{F} >$ [11], where:

- $\mathcal{U}$ is a set of background variables, also called exogenous variables, that are determined

by factors outside the model.

- $\mathcal{V}$ is a set $V_i, V_2, ..., V_n$ of variables, also called endogenous variables, that are determined by other variables in the model - that is variables in $\mathcal{U} \cup \mathcal{V}$.
- $P(\mathcal{U})$ is the probability function defined over the domain $\mathcal{U}$.
- $\mathcal{F}$ is a set of functions $f_1, f_2, ..., f_n$ such that each f_i is a mapping from (respective domains of) $\mathcal{U}_i \cup \mathcal{T}_i$ to $\mathcal{V}_i$, where $U_i \subset \mathcal{U}$ and $\mathcal{T}_i \subseteq \mathcal{V} \setminus \mathcal{V}_i$ constitutes the set of parent nodes of variables $\mathcal{V}_i$, and the entire set $\mathcal{F}$ forms a mapping from $\mathcal{U}$ to $\mathcal{V}$. That is, for $i = 1, ..., n$ each $f_i \in \mathcal{F}$ is such that:

$$v_i \leftarrow f_i(\mathcal{T}_i, \mathcal{U}_i) \tag{2.1}$$

i.e., f_i assigns a value to variable v_i that depends on the select set of variables in $\mathcal{U} \cup \mathcal{V}$.

Each SCM can be seen as partitioning the variables involved in a certain phenomenon into sets of exogenous (unobserved) and endogenous (observed) variables, respectively, $\mathcal{U}$ and $\mathcal{V}$. The exogenous ones are determined "outside" of the model and their associated probability distribution, $P(\mathcal{U})$, represents a summary of the state of the world outside the phenomenon of interest. Inside the model, the value of each endogenous variable v_i is determined by the causal process $v_i \leftarrow f_i(\mathcal{T}_i, \mathcal{U}_i)$, that maps the exogenous factors $\mathcal{U}_i$ and the specific set of endogenous variables $\mathcal{T}_i$ to $\mathcal{V}_i$.

An SCM can be represented as a Directed Acyclic Graph (DAG) and by a set of equations termed a Structural Equation Model (SEM) [92]. A DAG is a special type of graph for which all edges are directed and there are no cyclic paths. That is, between nodes information can only flow in one direction and the information that leaves a node can never loop back to the same node. The nodes in a causal DAG represent the variables in an SCM and the arrows

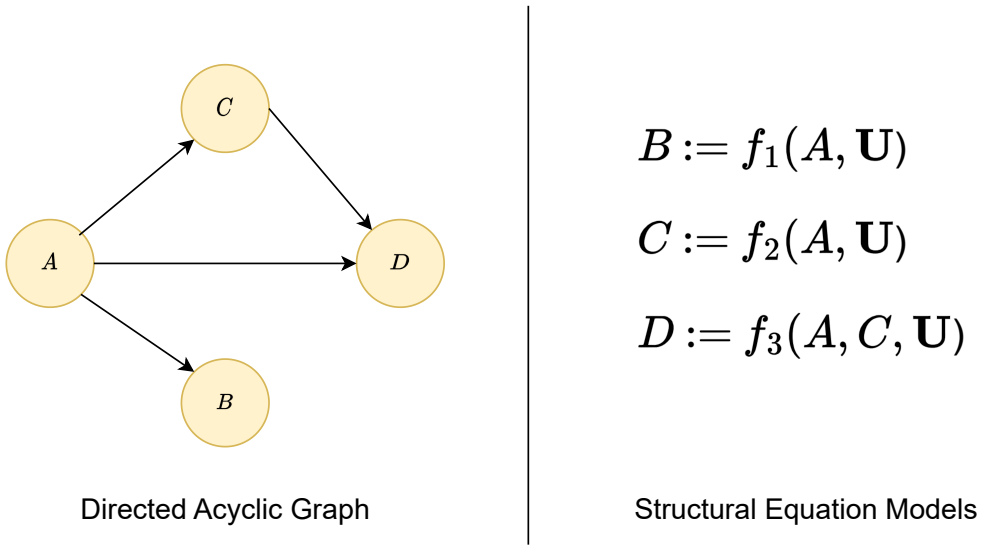


Figure 2.1: Structural Causal Model. The DAG representation is on the left and the SEM representation is on the right.

represent causation. Hence $\mathcal{T}_i$ is the set of all nodes having a directed edge towards the node V_i. SEMs on the other hand represent the exact causal relationships between variables. SEMs are generally asymmetric, meaning the equality only works in one direction. Figure 2.1 shows the DAG and SEM representation of a toy SCM.

2.4 Structure Learning

SCMs thus provide a rigorous framework for understanding causal relationships among variables in a complex system. However, in real-world settings, a true SCM is rarely available or is only partially available. The complete SCM is available only in special settings where laws of nature are understood with high precision, such as in physics or chemistry [105]. Access to the SCM of a dynamic system, particularly in critical domains such as healthcare, would be very helpful to the researchers. Causal Discovery or Structure Learning aims to uncover the causal structure from observational data [32]. Causal Discovery is an inverse

problem. Causal Discovery is an NP-hard problem as it involves searching over a large space of possible causal structures and evaluating the fitness of each structure based on a statistical criterion [17]. The space of possible structures (DAGs) grows exponentially with the number of variables.

Traditional structure learning methods can be categorized into constraint-based methods and score-based methods. Constraint-based methods are based on the idea that the causal relationships between variables can be inferred by examining the conditional independence relations between them [65, 87]. The most commonly used constraint-based method is the PC algorithm [87], which works by first identifying all the pairwise conditional independence relations between the variables, and then using these relations to construct a graph that represents the causal structure. While the PC algorithm is known to be efficient and reliable, it suffers from several limitations, such as the requirement of an acyclic causal graph and the assumption of faithfulness [57]. Score-based methods, such as the Bayesian score, maximum likelihood, and Akaike Information Criterion (AIC) [90], search for the causal structure that maximizes a predefined scoring function [22]. These methods do not rely on conditional independence assumptions and can handle datasets with hidden variables. However, they can be computationally expensive and may suffer from overfitting when the number of variables is large [22]. Structure learning is inherently a discrete optimization problem. DAG with NO TEARS [113] was the first method to recast the combinatoric graph search problem as a continuous optimization problem. They model the weighted adjacency matrix as a linear SEM and then fit observational data to the SEM along with l_1-regularization to enforce the sparsity of the graph. The main contribution of this work is the translation of the combinatorial acyclicity constraint into a continuous penalty derived as:

$$h(A) = tr(e^{A \odot A}) - |V| = 0 \tag{2.2}$$

where $|V|$ is the number of vertices in the graph defined by the adjacency matrix A, $tr(\cdot)$ is the trace operator and $\odot$ is the Hamdard product. Subsequently, this resulted in a growing interest in structure learning within the deep learning community [96]. Another recent method, CASTLE [54], regularizes neural networks by jointly learning causal relationships between variables. They use an auto-encoder-type neural architecture to learn causal DAG as an auxiliary task while training a supervised model. The authors of CausalVAE [107] use a Variational AutoEncoder-based approach to uncover causal relationships from latent factors. They propose a new framework termed causal disentanglement which includes a causal layer that transforms exogenous factors into endogenous factors of the learned DAG. Another auto-encoder type method, GAE [72] extends NO TEARS to facilitate non-linear structural relationships and vector-valued variables. Many of these methods deal with homogenous data only. Real-world signals such as electronic health records are multimodal and include not just continuous measurements but also recordings of patient-specific characteristics such as age, sex, comorbidities, and so on. Hence, there is still a dearth of methods that can be directly applied for practical applications. Also, a lot of the deep learning methods for causal discovery are computationally expensive and can only be used for small to medium-sized graphs and have difficulty scaling up to the interventional layer of Pearls Causal Hierarchy (PCH) [36]. This is not surprising as deep learning methods require significant quantities of data, and interventional data is not easily accessible in practical scenarios. Reinforcement learning techniques do look promising for overcoming this hurdle [42, 97, 114].

In recent times, GNNs have also been used for structure learning. DAG-GNN [108] extends DAG with NO TEARS by utilizing a deep generative model. They use a variational autoencoder parameterized by a GNN. Deep-GMG [60] also introduces a generative model for capturing probabilistic dependencies over the graph's edges and nodes. In their learning paradigm, new structures are sequentially added and GNNs are used for efficient represen-

tation at each step. Some recent works propose graph structure learning as an auxiliary task to downstream prediction tasks. While the main focus of these works is improving upon the downstream performance of the model, they introduce an interesting way in which differential graph structures can be learned during training. Raindrop [112] proposes a GNN architecture for the classification of irregularly sampled time series. Each time series is represented as a fully connected graph, whose edges are pruned during model training aimed at improving the classification accuracy. Their ablation study shows that distinctive graph structures corresponding to separate sample categories are learned during training. Lowe et al. [63] also use a related approach for time-series forecasting. In their work, they use a GNN encoder that receives a fully connected graph, and the encoder's output is used to predict the time series value at the current time step.

So far none of the works has explored the use of GNNs to score DAGs based on observational data. In Chapter 4, we perform experiments on BNs by simulating belief propagation using message passing in GNNs and explore if we can infer causal relationships from the observational data and intervene upon DAG structures.

Chapter 3

Change Detection in Chest Xrays

A chest radiograph or a Chest X-ray (CXR) is an imaging test used to diagnose conditions affecting the heart, lungs, and nearby structures. Chest radiography is one of the most performed diagnostic examinations in the world. It is the foremost imaging test for diagnosing symptoms such as breathing difficulties, persistent cough, chest pain, and fever, and aids the physicians in diagnosis and monitoring of conditions such as pneumonia, heart failure, lung cancer, emphysema, and other medical conditions. The demand for chest radiography has increased the radiologists' workload. As manually interpreting CXRs and radiology reports can be time-consuming, these challenges contribute to the delays in detecting findings and providing exemplary patient clinical management plans. About 129 million CXR were acquired in the United States alone in 2009 [66]. Subsequently, quite a few CXR datasets have been released by the research community specifically for the development of machine learning workflows [43, 46, 98]. A significant portion of research in this domain is focused on detecting and segmenting different anatomical regions within a CXR [50, 64, 85] or on the computer-aided diagnosis of CXR [5, 62, 82].

However, monitoring disease progression in CXRs, a routine task performed by radiologists, has so far attracted limited attention from the Artificial Intelligence (AI) community. Understanding if a patient's condition has deteriorated or improved is crucial to guide the physician's decision-making and determine the patient's clinical management. Automating this process is a challenging task. At times, differences between the X-rays are quite subtle

and to an untrained eye might go undetected. This will hinder early detection of disease progression which would have then required an immediate change in treatment plans. Therefore, there is a need for imaging models that can track disease progression or monitor the changes within CXR findings.

In this chapter, we propose `CheXRelNet`, an anatomy-aware neural model that utilizes the structural information encoded within a CXR to perform a longitudinal relational comparison between CXR exams for a variety of anatomical findings [48]. The proposed model uses Graph Attention Network [94] to capture the intra-image dependencies between different anatomical regions of the chest as well as inter-image temporal relations. `CheXRelNet` combines the localized region features with global CXR-level features to accurately capture anatomical location semantics for tracking disease progression.

The contributions of this work are summarized as follows:

1. We introduce `CheXRelNet`, an anatomy-aware model for tracking longitudinal relations between CXRs. The proposed model utilizes both local and global anatomical information to output accurate localized comparisons between two sequential CXR examinations.

2. We propose a novel graph construction workflow that takes into consideration the correlations between different anatomical regions of the chest as well as temporal relations across CXRs.

3. We conduct experimental analysis to demonstrate that our proposed `CheXRelNet` model outperforms baselines.

4. We perform transfer learning experiments to examine the generalization capabilities of our model across pathologies.

3.1 Methodology

3.1.1 Problem Definition

Let $\mathcal{C} = \{(x_i, x_i')\}_{i=i}^{N}$ be the set of CXR image pairs. Each image x_i has K anatomical regions. Each image is associated with a set of labels $\mathcal{Y}_i = \{y_{i,m}\}_{m=1}^{M}$, where $y_{i,m} \in \{0, 1\}$ indicating whether the label for pathology m appears in image x_i or not and each pair (x_i, x_i') is associated with a set of labels $\mathcal{Z} = \{z_{i,m}\}_{m=1}^{M}, z_{i,m} \in \{0, 1\}$ indicating whether the pathology m appearing in the image pair has improved or worsened. Within each CXR the pathology label $y_{i,m}$ is also associated with a specific anatomical Region of Interest (RoI), $k \in \{1, \ldots, K\}$. The goal is to design a model that compares the two images and predicts their labels as accurately as possible for an unseen image pair (x, x') and a wide range of pathologies.

3.1.2 CXR Graph Construction

We construct a graphical representation of the CXR image pair by utilizing (i) the correlation among anatomical region features from the images x_i, x_i', i.e. $R = f(x)$ and $R' = f(x')$, $R, R' \in \mathbb{R}^{K \times d}$, where K is the number of anatomical regions, each embedded into a row vector with dimensionality d (extracted by a pre-trained feature extractor f) and (ii) the correlation among anatomical regions between the two images in the pair. Given the initial training set of anatomical region representations $\{(R_i, R_i')\}_{i=1}^{N}$, we define a normalized adjacency matrix $A \in \mathbb{R}^{2K \times 2K}$ that captures intra-image and inter-image region correlations.

First, we construct the graph for a single CXR. This graph is defined by a normalized adjacency matrix $A_{intra} \in \mathbb{R}^{k \times k}$ that captures the intra-image region correlations. The intra-image correlations are calculated based on the region-disease co-occurrence. The region-

disease co-occurrence matrix is computed by finding the number of times two anatomical regions co-occur with the same pathological finding in the entire set of images $\mathcal{C} = \{(x_i, x_i')\}_{i=i}^{N}$. Each $K \times K$ co-occurrence matrix can be computed via the Jaccard similarity

$$J(r_s, r_t) = \frac{1}{M} \sum_{m=1}^{M} \frac{|\mathcal{Y}_{s,m} \cap \mathcal{Y}_{t,m}|}{|\mathcal{Y}_{s,m} \cup \mathcal{Y}_{t,m}|}. \tag{3.1}$$

Here, r_s represents an anatomical region, $\mathcal{Y}_s^m$ is the set of disease labels for region r_s and pathology m across all images and $\cap, \cup$ denote the intersection and union over multi-sets. To overcome the shortcomings of the label co-occurrence construction tendency to overfit the training data, a filtering threshold τ is adopted, *i.e.*,

$$A_{intra}(s,t) = \begin{cases} 1 & \text{if } J(R_s, R_t) \geq \tau \\ 0 & \text{if } J(R_s, R_t) < \tau \end{cases}. \tag{3.2}$$

This is a constant adjacency matrix and is used to define edge connections for all the CXRs.

The overall adjacency matrix A for the image pair constitutes two identical intra-image adjacency matrices, A_{intra} along the diagonal, and the two off-diagonal $k \times k$ blocks correspond to the relationship between the same anatomical regions of every pair of images. More precisely, we set $A(s,t) = \mathbb{1}\{t = s + k\}$ for $s = 1, \ldots, k$. The rationale of this adjacency matrix definition is that A will be associated with every pair (x_i, x_i') and will capture useful inter-image correlations and local intra-image region-level correlations. More precisely, the upper $k \times k$ diagonal block is associated with image x_i, forming a graph $G_i = (\mathcal{V}_i, \mathcal{E})$ with nodes being the vector representations of the k anatomical regions of image x_i. Similarly, the lower $k \times k$ diagonal block is associated with image x_i', forming a graph $G_i' = (\mathcal{V}_i', \mathcal{E})$ as before. Finally, the $k \times k$ off-diagonal blocks indicate a set of directed edges $\tilde{\mathcal{E}}$ between the same regions of images x_i, x_i'. This graph construction is also depicted in Figure 3.1.

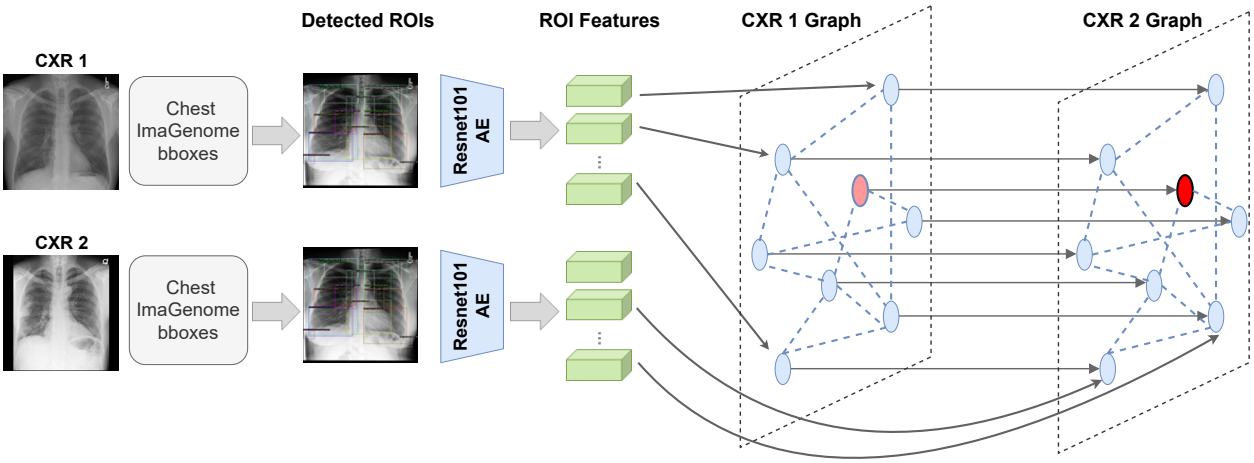


Figure 3.1: Graph Construction overview. Detected anatomical regions of interest (ROIs) are fed into a ResNet101 pre trained autoencoder to extract their corresponding visual features, formulating initial node representations $\mathcal{V}_i, \mathcal{V}_i'$ for CXR graphs $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E})$ and $\mathcal{G}_i' = (\mathcal{V}_i', \mathcal{E})$. Here, $\mathcal{E}$ is constructed based on intra-image region-disease co-occurrence. Moreover, the nodes of the two graphs are connected via a set $\tilde{\mathcal{E}}$ of directed edges indicating inter-image relations.

3.1.3 Graph Representation Learning

To capture global and local dependencies between anatomical regions, we utilize a graph attention network (GAT) [94] $Z_i = g(R_i, A) \in \mathbb{R}^{k \times d}$ to update R_i as follows:

$$R_i^{(t+1)} = \alpha_{i,i}^{(t)} W_1^{(t)} R_i^{(t)} + \sum_{j \in \mathcal{N}(i)} \alpha_{i,j}^{(t)} W_1^{(t)} R_j^{(t)}, \tag{3.3}$$

where $W_1 \in \mathbb{R}^{d \times d}$ is a learned weight matrix, $\mathcal{N}(i)$ denotes the neighborhood of x_i, t is the number of stacked GAT layers, and $\alpha_{i,j}$ are the attention coefficients computed as

$$\alpha_{i,j}^{(t)} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}^\top \left[W_1^{(t)} R_i^{(t)}; W_1^{(t)} R_j^{(t)}\right]\right)\right)}{\sum_{k \in \mathcal{N}(i) \cup \{i\}} \exp\left(\text{LeakyReLU}\left(\mathbf{a}^\top \left[W_1^{(t)} R_i^{(t)}; W_1^{(t)} R_j^{(t)}\right]\right)\right)} \tag{3.4}$$

Here, $\mathbf{a}$ is a learned weight vector, and ; denotes concatenation. The final region represen-

tations are computed by a weighted combination of the neighbour vector representations, scaled by their attention scores

$$R_i^{(t+1)} = \phi\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(t)} R_j^{(t)}\right), \tag{3.5}$$

where $\phi(\cdot)$ is a non-linear transformation. Given the history of the patient, a medical expert has enough information to direct the majority of their focus on a particular region within a CXR. In Figure 3.1, the node highlighted in red corresponds to the physician-designated focus region $k^* \in [1, k]$ for the particular CXR examination. We extract the node embedding corresponding to the focus region of x_i' for each CXR image pair and forward this embedding to the final dense classification layer. Specifically, for a focus-region $k^* \in [1, k]$, the extracted node embedding $R_i' \in \mathbb{R}^d$ is given by,

$$R_i' = R_i'^{(t+1)} \mathbb{1}\{k = k^*\} \tag{3.6}$$

To capture global image-level information, each image in pair (x_i, x_i') is encoded into two d-dimensional vectors by utilizing the pretrained feature extractor f, *i.e.*, $Q_i = f(x_i)$ and $Q_i' = f(x_i')$, $Q_i, Q_i' \in \mathbb{R}^d$. The final prediction is computed via

$$\hat{y} = [R_i'; Q_i; Q_i']\, W_2^T, \tag{3.7}$$

where ; denotes the concatenation of the local region-level and global image-level features, $W_2 \in \mathbb{R}^{3d \times M}$ is a fully connected layer that obtains the label predictions. The network is trained with a multi-label cross-entropy classification loss

$$L = \frac{1}{N} \sum_{i=1}^{N} \sum_{m=1}^{M} z_{i,m} log(\sigma(\hat{z}_{i,m})) + (1 - z_{i,m}) log(1 - \sigma(\hat{z}_{i,m})), \tag{3.8}$$

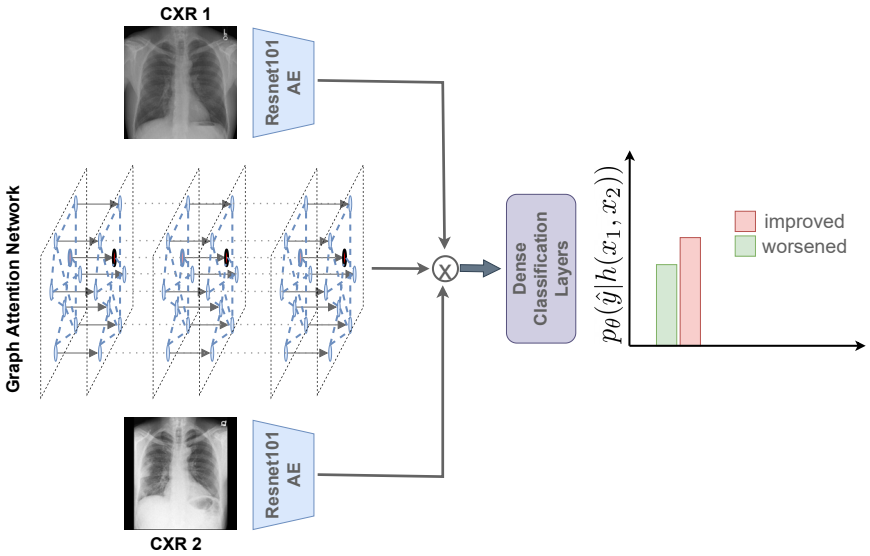


Figure 3.2: Classification module. The constructed graph, *i.e.*, the anatomical regions of interest (ROIs) vector representations and the corresponding adjacency matrix, is passed through a graph attention network that learns ROI inter-dependencies, essentially capturing local ROI information. Global image-level representations extracted from a pretrained ResNet101 autoencoder model are concatenated with the ROI learned representations and are passed through a final dense classification layer. The model is trained end-to-end with a cross-entropy classification loss.

where σ is the sigmoid function and $\{\hat{y}_i^m, y_i^m\} \in \mathbb{R}^M$ are the model prediction and the ground truth for example x_i, respectively. Figure 3.2 presents an overview of the model architecture.

3.1.4 Improving the Feature Extraction Pipeline

The two main components of `CheXRelNet` are the convolutional feature extractor and the graph neural network module. The feature extractor takes each cropped RoI as input and encodes it into a d-dimensional feature vector. If there are k anatomical regions within a single CXR image, the forward pass through the feature extractor will happen k times. Hence for a CXR pair, the forward pass happens $2k$ times. Quite unsurprisingly, experimentation revealed the feature extractor to be the main bottleneck in terms of training and inference

times.

The seminal R-CNN model [31] also faced similar computational inefficiencies which stopped them from achieving real-time object detection. The R-CNN model first performed a selective search over the input image to extract region proposals. These region proposals are then cropped, resized, and fed into a CNN-based feature extractor. The extracted features are then used to identify objects. This workflow is quite similar to `CheXRelNet`. Instead of selective search, we already have the RoI coordinates from the Chest ImaGenome dataset [104]. We crop these regions from the original image and then forward the encoded feature maps to the message-passing network for downstream comparison.

Fast R-CNN [30] addresses the computational drawbacks of the earlier method. In Fast R-CNN the expensive convolution operation is shared amongst all the RoI proposals. Here, the input image is fed to the CNN feature extractor, and RoI proposals are then cropped from the encoded feature maps. Hence, the feature map for all region proposals is generated via a single pass through the CNN feature extractor as opposed to generating a feature map for each region proposal individually by separate forward passes. We employ a similar feature extraction pipeline where the features for all anatomical regions within the CXR are extracted in a single forward pass. To realize this, we utilize Feature Pyramid Network (FPN) [61] and the RoIAlign layer [38]

FPN is a widely used technique for computer vision object detection tasks. FPNs address the problem of detecting objects at different scales by creating a multi-scale feature pyramid. The pyramid is created by combining feature maps from a feature extractor network, such as a Convolutional Neural Network (CNN), at different levels of abstraction and spatial resolution, with higher-level feature maps capturing more abstract and global features and lower-level feature maps capturing fine-grained details [61]. To pick feature maps for different resolutions of objects, FPNs use a technique called "pyramid pooling". In pyramid pooling,

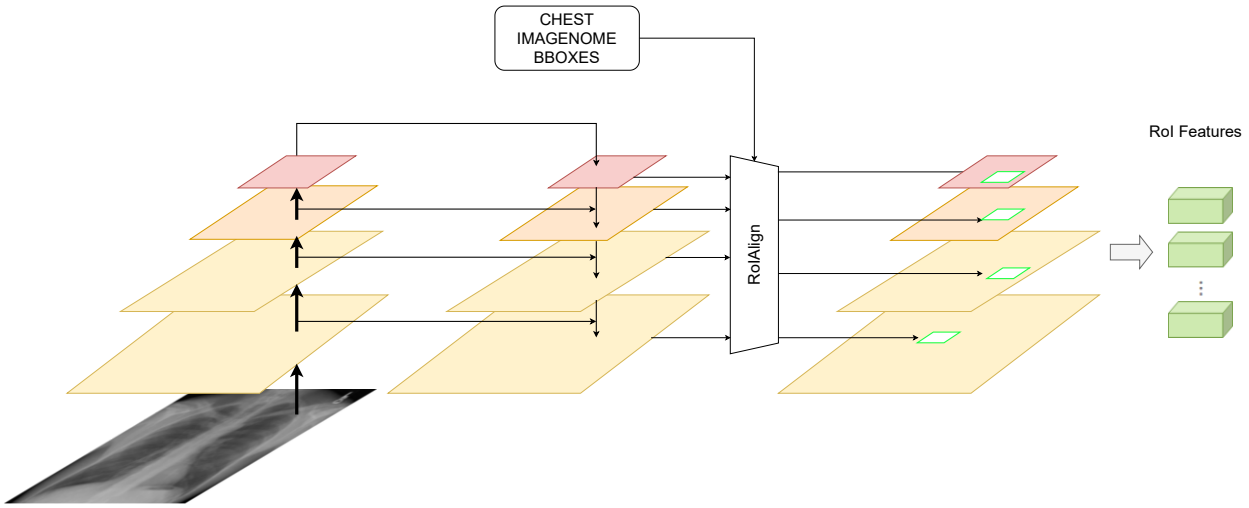


Figure 3.3: The FPN-based feature extraction pipeline

feature maps are pooled over different levels of the pyramid to generate a fixed-length representation for each region of interest. This representation captures both the global context and fine-grained details of the region of interest, allowing for accurate object detection at different scales.

Figure 3.3 shows the adaptation of FPN for extracting features from CXR regions. Using FPN we first extract four feature maps from a single CXR. We use these feature maps to then extract high-dimensional feature vectors for each anatomical region within the CXR. Depending upon the spatial sizes of the regions in the original image and the feature map sizes, different regions are mapped onto different feature maps using the RoIAlign layer. The RoIAlign layer takes the original bounding box coordinates as input and rescales them with respect to the dimensions of the encoded feature maps. It uses bilinear interpolation [1] while rescaling the coordinates to avoid loss of information due to quantization. We assign an RoI of width w and height h to the level P_k of our feature pyramid as [61]:

$$k = \lfloor k_0 + \log_2(\sqrt{wh}/224) \rfloor \tag{3.9}$$

Here 224 is the input CXR size, and k_0 is the feature map level on which an RoI with $w \times h = 224^2$ should be mapped onto. Hence, according to Equation 3.9, RoIs with smaller scales are mapped on higher-resolution feature maps. Put simply, instead of cropping the RoIs from the original image and then feeding each one separately to a CNN-based feature extractor, we first extract feature maps and then using RoI Align directly crop the RoIs from these feature maps. Overall using FPNs with the RoI Align layer results in more efficient computation and solves the issue of the ResNet encoder being the computational bottleneck.

3.2 Experiments

3.2.1 Dataset

The proposed `CheXRelNet` model is trained and evaluated on the Chest ImaGenome dataset [104]. This dataset was generated by locally labelling 242,072 frontal MIMIC-CXRs [46] (AP or PA view) automatically through a combination of rule-based text analysis and atlas-based bounding box extraction techniques [102, 103]. Chest ImaGenome represents the connections of each CXR annotation as an anatomy-centred scene graph, following a radiologist-constructed CXR ontology. The dataset contains 1,256 combinations of relation annotations between 29 CXR anatomical locations and their attributes structured as one scene graph per image, and about 670,000 localized comparison relations between the anatomical locations across sequential exams. In this work, we utilize the localized comparison relations data that involves cross-image relations for the 9 pathologies. Each comparison relation in the Chest ImaGenome dataset consists of the DICOM identifiers of the two CXRs being compared, the particular pathological finding observed in those two CXRs, the anatomical region of interest on which the radiologist's comparison is focused, and the corresponding comparison label. In addition to comparison relations, the Chest ImaGenome dataset also provides bounding box information for extracting individual anatomical regions from the CXRs, viz. "Left Lung", "Cardiac Silhouette", etc. For each of the 242,072 frontal MIMIC-CXRs, a list of anatomical regions (bboxes) is provided, as well as the corresponding Euclidean coordinates for each bounding box. We utilize these coordinates to crop different anatomical regions within a CXR. There are a total of 122,444 unique comparisons in the dataset, of which 79,902 have at least one of the nine selected pathology labels, in addition to regions detected by the object detection pipeline and the overall comparison relation. Table 3.1 shows high-level data statistics. For each image, except for those with the pathology label as "Enlarged Cardiac

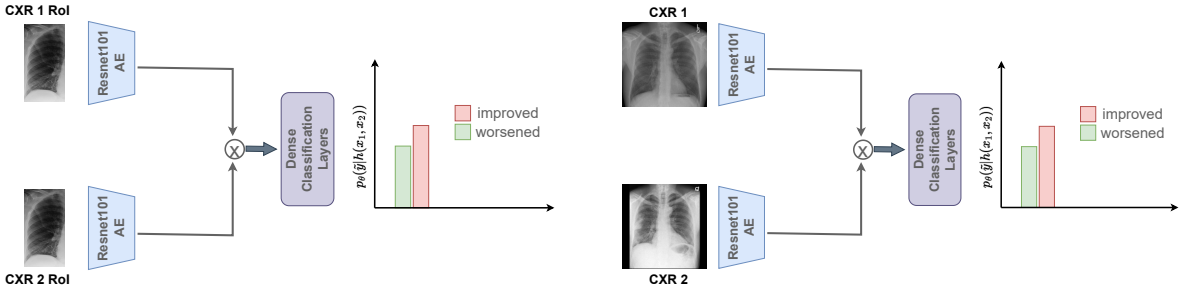
Table 3.1: Dataset Characteristics. # Image Pairs (number of comparison CXR pairs) and # Bboxes (number of bounding boxes) and # Training Pairs (number of training comparison CXR pairs) per pathology label. Each pathology is indexed with a pathology ID (first column).

Pathology ID	Description	# Image Pairs	# Bboxes	# Training Pairs
D1	Lung Opacity	32,524	455,336	22,620
D2	Pleural Effusion	13,122	183,708	9,192
D3	Atelectasis	9,660	135,240	6,922
D4	Enlarged Cardiac Silhouette	1,958	3,916	1,384
D5	Pulmonary Edema/Hazy Opacity	12,090	169,260	8,424
D6	Pneumothorax	2,728	38,192	1,930
D7	Consolidation	3,332	46,648	2,310
D8	Fluid Overload/Heart Failure	674	9,436	132
D9	Pneumonia	3,814	53,396	2,590
All 9 Pathologies	**Total**	79,902	1,095,132	55,504

Silhouette", 7 of the most frequently occurring anatomical regions were extracted. For the pathology label "Enlarged Cardiac Silhouette", the dataset provides only one corresponding bounding box.

3.2.2 Baselines

We compare the `CheXRelNet` model against the following baselines: 1) **Local** model: we utilize a previously proposed siamese network trained on cropped RoIs, encoded with a pre-trained ResNet101 autoencoder and passed through a dense layer and a final classification layer [104]. This model essentially only looks at the corresponding anatomical regions and considers neither global information nor intra-region dependencies. 2) **Global** model: we also design a Siamese architecture that encodes the entire CXR as opposed to only the cropped RoIs in the Local model. Apart from the input being a full image rather than an RoI, the model architecture is the same as the Local model. Hence, this baseline incorporates the global information but does not take into consideration the anatomical region of interest nor explicitly models inter-region dependencies. These two siamese models serve as baseline



(a) Local Baseline (b) Global Baseline

Figure 3.4: The Local and Global baseline siamese networks

methods to contrast the effectiveness of `CheXRelNet`, which not only is location-aware but can also explicitly model both inter-region and intra-image CXR dependencies. Figure 3.4 shows the two baseline model architectures.

3.2.3 Implementation Details

To train each model, we use the train/validation/testing splits and detected RoIs provided by Chest ImaGenome. The two different feature extraction pipelines result in slightly distinct training workflows. The base feature extractor is a pre-trained ResNet101 autoencoder [19]. This autoencoder is trained on Padchest [14], NIH [44], CheXpert [43], and MIMIC datasets [46]. For each image within the comparison pair, we crop the image RoIs and resize them to 224×224, and feed the cropped RoIs separately to the ResNet101 encoder. When using the FPN-based feature extraction pipeline, the entire image is directly fed to the ResNet101 encoder, and the RoIs are cropped from encoded feature maps. We extract 4 feature maps from different layers of the encoder, their sizes are $(256 \times 56 \times 56)$, $(512 \times 28 \times 28)$, $(1024 \times 14 \times 14)$, and $(2048 \times 7 \times 7)$. Each cropped RoI is then finally embedded into a 2048-dimensional vector. The co-occurrence matrix threshold is set to 0.5.

Our model is a 2-layer graph neural network with 2048 and 1024 neurons per layer, in the

Table 3.2: Comparison against baselines (accuracy)

Method	D1	D2	D3	D4	D5	D6	D7	D8	D9	**All**
Local	0.59	0.53	0.60	0.47	0.56	0.46	0.61	0.47	0.63	0.60
Global	0.67	**0.69**	0.64	0.74	0.71	0.50	0.65	0.69	0.67	0.67
`CheXRelNet`	0.67	0.68	**0.66**	**0.75**	0.71	**0.52**	**0.67**	**0.73**	0.67	**0.68**

first and second layers, respectively. There are 5 and 3 multi-head-attentions in each respective layer. The output from the graph attention network is concatenated with the global information and then passed through two dense layers of sizes 768 and 128, respectively. We train the network using Adam [51] optimizer for 200 epochs, with a $0.8e^{-3}$ initial learning rate [100] and a batch size of 32. To avoid overfitting, we utilize early stopping with patience set to 11 epochs and gradient clipping set to 0.1. In addition, we use 0.5 Dropout [88] and a learning rate decay factor of 0.3 with the patience threshold set to 4. The model is implemented by utilizing the PyTorch [74] and pytorch_geometric [26] deep learning frameworks. The evaluation metric is accuracy and results are reported over six experimental trials.

3.3 Results

3.3.1 Binary Classification

Results are summarized in Table 3.2. `CheXRelNet` achieves a mean accuracy of 0.683 (SD=0.0024), while the Local model has 0.602 mean accuracy (SD=0.0059) and the Global model has 0.672 mean accuracy (SD=0.0046) over six trials. We observe that the Local model is generally underperforming, and it is most likely limited because it focuses on a specific anatomical region and completely neglects global information. In contrast, radiologists often take into consideration more than one anatomical region when drawing inferences from CXRs. The Global model is a lot more effective than the Local one, and incorporating

Table 3.3: Transfer learning evaluation against baselines (accuracy). Models are trained on D6-D9 and tested on unseen pathologies (D1-D5). SetA consists of unseen pathologies {D1, D2}. SetB consists of unseen pathology labels, {D3, D4}. Set C consists of all unseen pathology labels {D1,D2,D3,D4,D5}.

Method	D1	D2	D3	D4	D5	SetA	SetB	SetC
Local	0.56	0.49	0.54	0.49	0.55	0.54	0.55	0.54
Global	0.61	**0.63**	0.60	0.65	0.63	0.61	0.63	0.62
`CheXRelNet` (ours)	**0.64**	0.60	**0.61**	**0.68**	**0.67**	**0.63**	**0.64**	**0.64**

global information boosts the prediction accuracy. Yet, the Global model is also limited as it focuses on the entire image but fails to consider the relationships among anatomical regions. We additionally perform statistical significance tests, i.e. an unpaired t-test ($p = 0.049$) and a one-tailed t-test ($p = 0.018$) comparing `CheXRelNet` and the Global baseline. These t-test results verify that the `CheXRelNet` and Global baseline predictions follow distinct distributions and that the improvement in accuracy is significant at $p < 0.05$. Overall, `CheXRelNet` improves upon the Global model's prediction accuracy by modelling the inter-image and intra-image region correlations and attending to the anatomical regions of interest.

3.3.2 Transfer Learning

We also perform a transfer learning experiment wherein we train `CheXRelNet` on a set of diseases and test performance on a different set of diseases [16, 106]. Specifically, we train `CheXRelNet` on a subset of the data with 'Pneumothorax', 'Consolidation', 'Fluid Overload/-Heart Failure', 'Pneumonia' (D6-D9) pathologies, and test on the following pathology labels that are unseen during training: 'Lung Opacity', 'Pleural Effusion', 'Atelectasis', 'Enlarged Cardiac Silhouette', and 'Pulmonary Edema/Hazy Opacity' (D1-D5). Results are reported in Table 3.3. We perform this experiment on individual unseen pathology labels as well as on sets of multiple unseen pathology labels. We observe that our model can generalize well to

Table 3.4: Pathology-specific comparison of `CheXRelNet` against baselines.

Method	D1	D2	D3	D4	D5	D6	D7	D8	D9	AVG
Local	0.63	0.55	0.59	0.62	0.68	**0.53**	0.60	0.45	0.63	0.59
Global	**0.68**	0.64	**0.61**	0.69	**0.70**	0.49	0.59	**0.69**	0.58	0.63
`CheXRelNet` (ours)	0.67	**0.69**	0.61	**0.71**	**0.70**	0.49	**0.67**	0.65	**0.65**	**0.65**

unseen pathology labels. We can attribute this to the incorporation of both local and global information during training. The model is learning associations between different anatomical regions and therefore can identify complex bio-markers associated with the progression of pathologies.

3.3.3 Pathology-specific Models

In this experiment, we train models that are specific to a given pathology label. Hence, unlike Table 3.2, where the model is trained jointly for all nine pathology labels and later tested over each individual pathology, here we train and test our model on the same pathology label. Results are shown in Table 3.4. From these results, we can infer that `CheXRelNet` is comparable to or outperforms the Local and Global baselines for five out of nine pathologies. For the pathologies 'Pleural Effusion' (D2), 'Consolidation' (D7), and 'Pneumonia' (D9), the difference in accuracy is greater than or equal to 5%. Compared to the results in Table 3.2, another interesting observation is that when our model is trained for all pathologies, the classification accuracy is higher. We attribute this observation to the presence of a consistent pattern of disease progression across all pathologies. Hence, the model trained on all pathologies can generalize better and is more consistent than the one trained on a single pathology.

Table 3.5: Ablation study on model structure and capacity.

Model	Local			Global			CheXRelNet			
	Type A	Type B	Type C	Type A	Type B	Type C	Type A	Type B	Type C	Type D
#Parameters (M)	25.6	34.7	41.4	25.6	34.7	41.4	38.6	27.8	54.3	28.9
Accuracy	0.60	0.64	0.63	0.67	0.67	0.67	0.68	0.68	0.68	0.67

3.3.4 Ablation Study: Model Architectures and Capacity

We perform an ablation study to investigate if there exists a correlation between the performance and the model capacity (number of trainable parameters). Results are shown in Table 3.5. For all three models, the architectures named *Type A* are the ones used throughout the study. The architectures named *Type B* and *Type C*, for the Local and Global baselines, have more dense layers and neurons. As for the graph models, *Type B*, *Type C* are the shallower and deeper versions of `CheXRelNet` having 1 and 3 Graph Attention (GAT) layers, respectively, whereas *Type A* is the actual `CheXRelNet` that has 2 GAT layers. *Type D* is the version of our `CheXRelNet` model without attention and in that we replace the GAT layers with simpler Graph Convolution layers. The number of trainable parameters and corresponding accuracy is reported in Table 3.5. We can infer that the model performance is less influenced by the model capacity and that the graph neural network is the prominent differentiating factor.

3.3.5 Improvement in Computing Efficiency with FPNs

The use of FPN and RoI-Align layers greatly speeds up the training and inference phase. Table 3.6 compares the training and inference times, and the overall prediction accuracy of `CheXRelNet` and `CheXRelNet` with FPN. Without FPN, `CheXRelNet` has a training time of 13.9 hours (SD = 1.07) and an inference time of 1.28 seconds (SD = 0.49) on GPU, over six trials. With FPN, `CheXRelNet` has a training time of 6.7 hours (SD = 0.82) and an

Table 3.6: Speedup with FPN and RoIAlign layers

Model	Training (↓)	Inference (↓)	Accuracy (↑)
ChexRelNet	13.9 h	1.28 s	0.68
ChexRelNet with FPN	6.7 h	0.6 s	0.68

Table 3.7: Comparison against baselines with 'no change' samples.

Method	D1	D2	D3	D4	D5	D6	D7	D8	D9	**All**
Local	0.41	0.37	0.41	0.29	0.37	**0.37**	**0.49**	0.29	0.42	0.43
Global	0.45	**0.47**	**0.44**	0.48	0.48	0.36	0.47	**0.50**	0.43	0.45
`CheXRelNet` (ours)	**0.49**	**0.47**	**0.44**	**0.49**	**0.49**	0.36	0.47	0.44	**0.47**	**0.47**

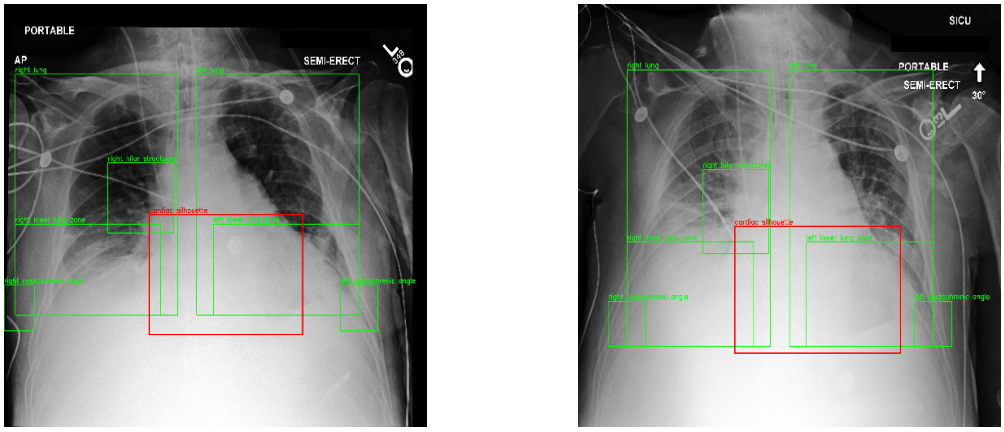
inference time of 1.02 seconds (SD = 0.3) on GPU, over six trials. However, we do not see any improvement in the model performance over the change classification task. Overall accuracy with FPN is 0.680 (SD = 0.016) and without FPN is 0.683 (SD = 0.0024) Hence, we can conclude that the addition of FPN and RoI-Align results in almost 2x speedup in training and inference.

3.3.6 Adding "No Change" samples

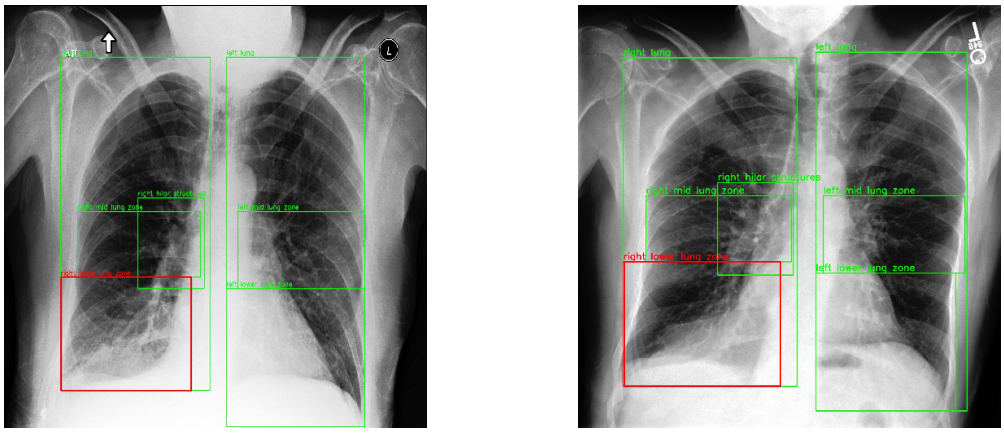
Table 3.7 shows the results when we repeat the original experiment with samples wherein there is no change in the disease progression. Albeit the accuracy drops considerably for all models, `CheXRelNet` outperforms baselines. Future work can target improving the model performance and making the model robust to "No Change" samples.

3.3.7 Qualitative Results

We visualize the model predictions for different pathologies. Figure 3.5a showcases an input image pair for the pathology label 'Fluid Overload/ Heart Failure' where there has been a worsening in the patient's condition. For this particular pair, the anatomical region of



(a) Image pair for pathology D8, class: Worsened



(b) Image pair for pathology D9, class: Improved

Figure 3.5: Qualitative Results. Figure (a) shows the image pair for pathology *Fluid Overload/Heart Failure*. Figure (b) shows the image pair for pathology *Pneumonia*

interest (ROI) is 'Cardiac Silhouette', which is depicted with a red bounding box. Other anatomical regions that our model takes into consideration when making predictions are shown in green bounding boxes. The previous and current CXRs are named 'CXR 1' and 'CXR 2', respectively. Upon close inspection, we can see there are subtle changes within the ROI as well as in other parts of the CXR. There is increased haziness in the Left and Right Lungs, and minute changes in the Cardiac Silhouette. Similarly, Figure 3.5b, depicts the

input image pair for the pathology label 'Pneumonia', and the case where there has been an improvement in the patient's condition. For this particular pair, the anatomical ROI is the 'Right Lower Lung Zone'. In addition to changes within the ROI, there are significant improvements in the regions 'Left Mid Lung Zone' and 'Left Lower Lung Zone'. The Local model focuses only on the ROI, whereas the Global model focuses only on the entire image. Hence, both of these fail in making correct predictions about these images. Our `CheXRelNet` model builds associations between various regions and hence is able to factor in the minute changes across the entire anatomy while making predictions.

3.4 Discussion

CXRs are repeatedly requested in the clinical workflow to assess for a myriad of attributes. Diagnosis and monitoring are typically performed through comparisons of sequential CXR images, both in in-patient and outpatient settings. Given a patient with two sequential CXR exams, the goal of this work is to automatically evaluate disease change. To this end, we describe a methodology for localized relation comparisons between CXR images. This is a rather complex task because

1. There is no ideal frame of reference in chest radiography. *i.e.*, there is no CXR that represents the absolutely perfect health condition and hence enables us to compare a new CXR exam and directly infer whether the medical condition of the patient has improved or worsened.

2. The biomarkers for change are specific to a given patient and for a given pathology. i.e. two patients sharing the same condition will show different features of change even if the disease progression trend for both of them is the same.

3. Finally, slight changes between CXR examination environments caused by changes in lighting conditions, patient movements, and deviations in the angle of imaging, make it even more difficult to extract quality visual features from the CXRs.

The proposed `CheXRelNet` makes initial strides in overcoming these hurdles by fusing global image-level information, local intra-image region-level correlations, and inter-image correlations. Experimental results show that `CheXRelNet` outperforms baselines in both traditional and transfer learning settings. As a result, our method provides the necessary components for monitoring the progression of pathologies that are visualized through chest imaging. With hopes of igniting future research in this direction, we have open-sourced the code at https://github.com/PLAN-Lab/ChexRelNet.

In the future, we hope to expand our work to model disease progression among several sequential CXRs, incorporate additional temporal context information and physiological data [47, 104] and account for the time interval variability found in longitudinal imaging records. Also, the net gain in performance over the global baseline model is limited. This could be improved upon by constructing more informative graphs. In the current graph construction methodology the edges are static. A possible future work could transform the edges into more stochastic nature and infuse edge features into them. Expanding the model to cover the "No Change" samples without a drop in performance is also something we plan to further investigate.

Chapter 4

Causal Structure Learning

The previous chapter introduced a novel way to analyse chest radiographs and track anatomical changes over time. While we utilized only image data, there are abundant other works which make use of medical data in different modalities - images, text, speech, biological signals, *etc.*, for diagnosis and treatment recommendation tasks [77]. Neural methods can take in, process, and output multimodal information. Overall the universal applicability of neural networks is one of the most widely recognized and decorated findings in the study of intelligence [41]. The universal approximation theorem of neural methods along with the insight that most tasks can be simplified and represented as input/output, that is, as functions, results in the belief that neural networks under the correct set of conditions can solve the majority of tasks in AI. In this chapter, we investigate this belief in the context of Causal Inference, Structural Causal Models (SCMs), and Graph Neural Networks. More precisely we explore whether neural methods with inductive bias such as GNNs can be utilized for causal discovery based on observational data.

Correlation seldom reveals causation. This is especially significant in medical environments. For example, consider a machine learning model that predicts hospital readmission rates based on EHRs. If the model detects a strong correlation between a particular medication and readmission rates, it may be alluring to say that the medication is the cause behind patient re-admissions. However, without considering other contributing factors such as patient demographics, the severity of underlying conditions, and so on, this reasoning may be

incorrect. Hence, by incorporating causal reasoning into deep learning methods we can augment their accuracy and reliability, and ultimately improve clinicians' trust in traditionally black-box deep learning techniques.

Graphs, or more precisely Directed Acyclic Graphs (DAGs), are core to causality and are a simple and intuitive way of representing SCMs. Graph-structured data is widely prevalent in the real world and healthcare settings. It is thus quite interesting to think about the usability of GNNs in uncovering causal insights in data. From a high-level perspective, one can intuitively relate Belief Propagation (BP) in BNs to the message-passing framework in GNNs. In this chapter,

1. We explore the applicability of Graph Neural Networks for causal inference.
2. We relate interventions in causality theory to graph manipulations.
3. We try to simulate BP with message passing in GNNs and hence explore the utility of GNNs in structure learning.
4. We conduct experimental analysis on nine expert-curated Bayesian networks, with six of these originating from the medical domain.

4.1 Methodology

Zecevic et al. [109] establish theoretical proofs and reasoning relating GNNs to SCMs. They theoretically prove that any GNN can be seen as a neural SCM variant. Additionally, they extend this relation to the Neural Causal Model (NCM) introduced in [105]. The concept of intervention is central to causality as suggested by Holland et al., -"No causation without manipulation" [40]. Hence in Zecevic et al. [109], equation (9) introduces the process of intervention within a GNN computational layer. They define intervention as follows:

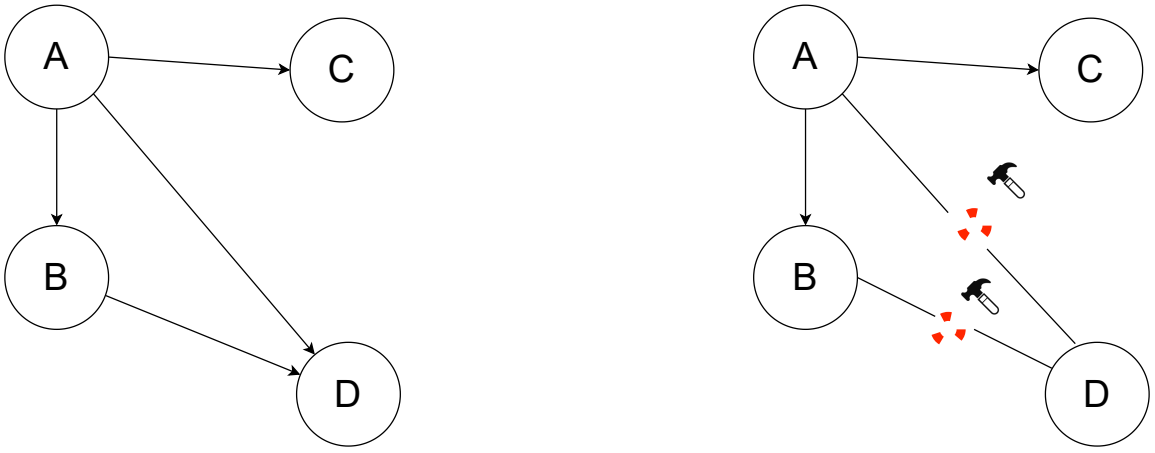
Definition 1 (Intervention): An intervention x on the corresponding set of variables $\mathcal{X} \subseteq \mathcal{V}$ within a GNN layer $f(\mathbf{D}, \mathbf{A}_G)$ denoted by $f(\mathbf{D}, \mathbf{A_G}|do(\mathbf{X} = x))$, where $\mathbf{D}$ is some dataset considered to be vector-valued samples of our variables $\{v_i\}_{i=1}^n \in \mathbb{R}^d$ and $\mathbf{A_G} \in \mathbb{R}^{n \times n}$ is the adjacency matrix representation of our graph G, is defined as a modified layer computation,

$$h_i = \phi(v_i, \bigoplus_{j \in M_i^G} \psi(v_i, v_j)), \tag{4.1}$$

where the localized local neighbourhood is given by,

$$\mathcal{M}_i^G = \{j | j \in \mathcal{N}_i^G, j \notin \mathcal{T}_i \iff i \in X\} \tag{4.2}$$

where $\phi(\cdot)$ is permutation invariant function on each of the variable features d_i and their respective neighbourhoods, $\mathcal{N}_i^G$ denotes the regular graph neighborhood, $\mathcal{T}_i$ denotes the set of parent nodes of the i_{th} variable, and h_i represents the updated information of node i aggregated ($\bigoplus$) over its neighborhood in the form of messages ψ. Such GNN layers are said to be interventional. This definition stresses the local nature of intervention in SCM. Simply put intervention with the *do* operator in an SCM is analogous to manipulating the



(a) Original causal DAG (b) Intervened DAG

Figure 4.1: Doing interventions on causal DAGs

graph. After intervening or manipulating the graph the modified neighbourhood of a given node is the subset of the original neighbourhood. This alters the adjacency matrix. This is illustrated in Figure 4.1. In our experiments, instead of manipulating the graphs within GNN layers, we do the interventions outside the GNN. This change simplifies the GNN training methodology and enables us to use a more standard GNN architecture such as Graph Convolution Network (GCN). It also allows us to realize a direct one-to-one mapping between interventions and the downstream prediction performance of the model.

Let $\mathcal{C}$ be an SCM represented by the causal DAG $G(\mathcal{V}, \mathcal{E})$, where $v_i \in \mathcal{V}$, for $i = 1, ..., n$ is the set of d endogenous variables or graph vertices. Given, this SCM $\mathcal{C}$ we can simulate a dataset $\mathbf{D}_\mathcal{V}$ of N samples by forward sampling. Each sample $d_i \in \mathbb{R}^n$ in $\mathbf{D}_\mathcal{V}$ constitutes of n variables $\{v_i\}_{i=1}^n \in \mathbb{R}^d$. We select a target variable $v_t \in \mathbb{R}^d$, for $t \in 1, ..., n$, and define a neural model $v_t \leftarrow f_\theta(D_{\mathcal{V}'}, A_G)$, where $\mathcal{V}' = \mathcal{V} \setminus v_k$, A_G is the adjacency matrix defining the DAG G, and $\theta = \arg\max P(v_t | D_\mathcal{V}, G(\mathcal{V}, \mathcal{E}))$. Given the model f_θ, the sampled data $D_\mathcal{V}$, and the causal DAG G, we aim to evaluate how the interventions on the DAG affect the downstream prediction performance of v_k, and hence (indirectly) infer the causal relationships. The interventions are caused by altering A_G. The underlying intuition is

given an SCM and an accompanying causal DAG, the model performance on a downstream inference task will vary with the structure of the DAG. When the DAG structure deviates from the actual DAG structure of the data-generating SCM, the downstream performance will decrease. We can thus use a neural model conditioned on the adjacency matrix to rank different causal DAG structures using the downstream prediction performance as a proxy. Formally the null and alternative hypotheses of our experiments are

Null Hypothesis, $\mathbf{H_0}$: The downstream prediction performance does not vary with the amount of intervention in the DAG structure. Hence it cannot be used to rank DAGs.

Alternative Hypothesis, $\mathbf{H_a}$: The downstream prediction performance varies with the amount of intervention in the DAG structure. Hence it can be used to rank DAGs.

The model f_θ compromises of message passing the GNN layer and a final dense layer to perform the prediction (classification) task. We first embed all the endogenous variables in $\mathcal{V}' = \mathcal{V} \setminus v_k$ in a high-dimensional space. Each variable acts as a node in the graph defined by the adjacency matrix $A_{G_{\mathcal{V}'}} \in \mathbb{R}^{(n-1)\times(n-1)}$. To simulate belief propagation via message passing, we utilize a Graph Convolutional Network (GCN) [52], $R = g(v_i, \mathcal{A}_{G'_V})$, as follows:

$$R^{(t+1)} = \sigma(\tilde{O}^{-1/2}\tilde{A}\tilde{O}^{-1/2}R^tW^t) \tag{4.3}$$

$W^{(t)}$ is the trainable weight matrix for GCN layer t. $\tilde{A} = A_{G_{\mathcal{V}'}} + I_{n-1}$ is the adjacency matrix with added self-connections and $O_{ii} = \sum_j \tilde{A}_{jj}$. $\sigma(.)$ is the activation function such as ReLU [67]. $R^{(t)} \in \mathbb{R}^{(n-1)\times d}$ are the node embeddings for all n nodes after the L^{th} GCN layer. After L GCN layers, we extract node embeddings corresponding to the terminal nodes. Terminal nodes are defined as follows:

Definition 2 (Terminal Nodes). An endogenous node is said to be a terminal node in the DAG if it is the direct parent of the designated target node. *i.e.*, $\mathcal{V}_{tn}$ is a set of terminal nodes with respect to the target node v_t if and only if $\mathcal{V}_{tn} \in \mathcal{V} \setminus v_t$ and $\mathcal{V}_{tn} = \mathcal{T}_{v_t}$, where $\mathcal{T}_{v_i}$ are the parents of node $v_i \in \mathcal{V}$.

The extracted node embedding $R_{tn} \in \mathbb{R}^{(|\mathcal{V}_{tn}| \times d)}$ is given by,

$$R_{tn} = {R_{v_i}}^{(t+1)} 1\{v_i \in V_{tn}\}. \tag{4.4}$$

The final prediction of the target node state is computed as,

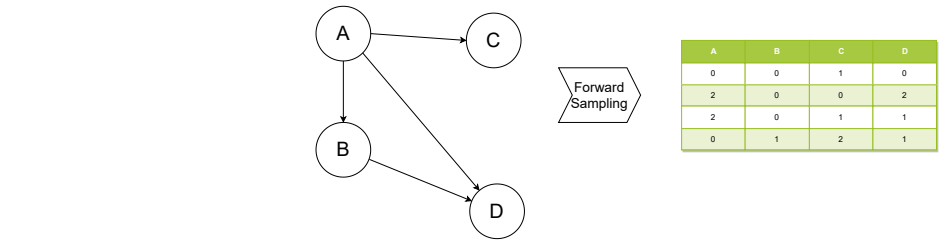
$$\hat{v}_t = R_{tn} W_2^T, \tag{4.5}$$

where, $W_2 \in \mathcal{R}^{(|\mathcal{V}_{tn}|d) \times M}$ is a fully connected layer and M is the total number of states the target node v_t takes. The overall network is then trained with binary cross-entropy loss,

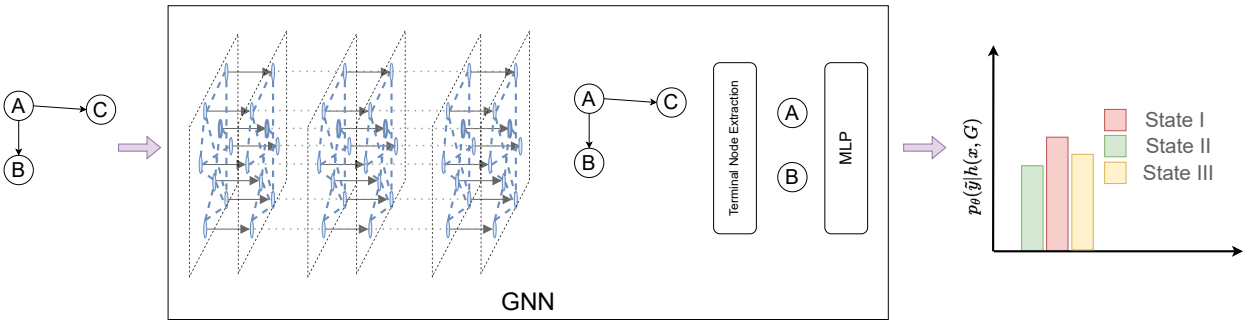
$$L = \frac{1}{N} \sum_{i=1}^{N} \sum_{m=1}^{M} v_{t_{i,m}} log(\sigma(\hat{v}_{t_{i,m}})) + (1 - v_{t_{i,m}}) log(1 - \sigma(\hat{v}_{t_{i,m}})), \tag{4.6}$$

where, $\sigma(.)$ is the softmax operator. The entire model and the methodology are illustrated in Figure 4.2.

We follow the same training procedure for all intervention operations. That is, each intervention creates a new DAG structure as shown in Figure 4.1. Then this manipulated DAG is fed into the GNN model. We can then evaluate the structural integrity of DAGs based upon the GNN performance on the simulated data generated by forward sampling the original SCM $\mathcal{C}$. We use prediction accuracy to score and rank each DAG. This is illustrated in Figure 4.3.



(a) Simulating Data by forward sampling the BN



(b) GNN model

Figure 4.2: Method Design. Figure (a) shows the DAG representation of a data-generating SCM or BN. We simulate data by forward sampling this BN. Figure (b) shows the entire GNN pipeline. The node embedding from the graph minus the target node (here D) is fed to the GCN layers. Message passing happens within the GCN layers and the updated node embeddings are passed ahead. Based upon the original DAG structure (refer Figure (a)) only the terminal node embeddings (A and B) are passed on to the final classification layer, which makes predictions for the target node (D).

4.2 Experiments

4.2.1 Dataset

The proposed GNN model is trained and evaluated on 9 diverse Bayesian Networks (BNs) from the *bnlearn* network repository [81]. The BNs are summarized in Table 4.1. We run experiments over 3 small BNs, 3 medium BNs, 2 large BNs, and 1 very large BN. The number of nodes in the network characterizes the size of the BNs. All BNs are composed of discrete variables. Hence, the final prediction task is multi-class classification. Below we describe each network in detail.

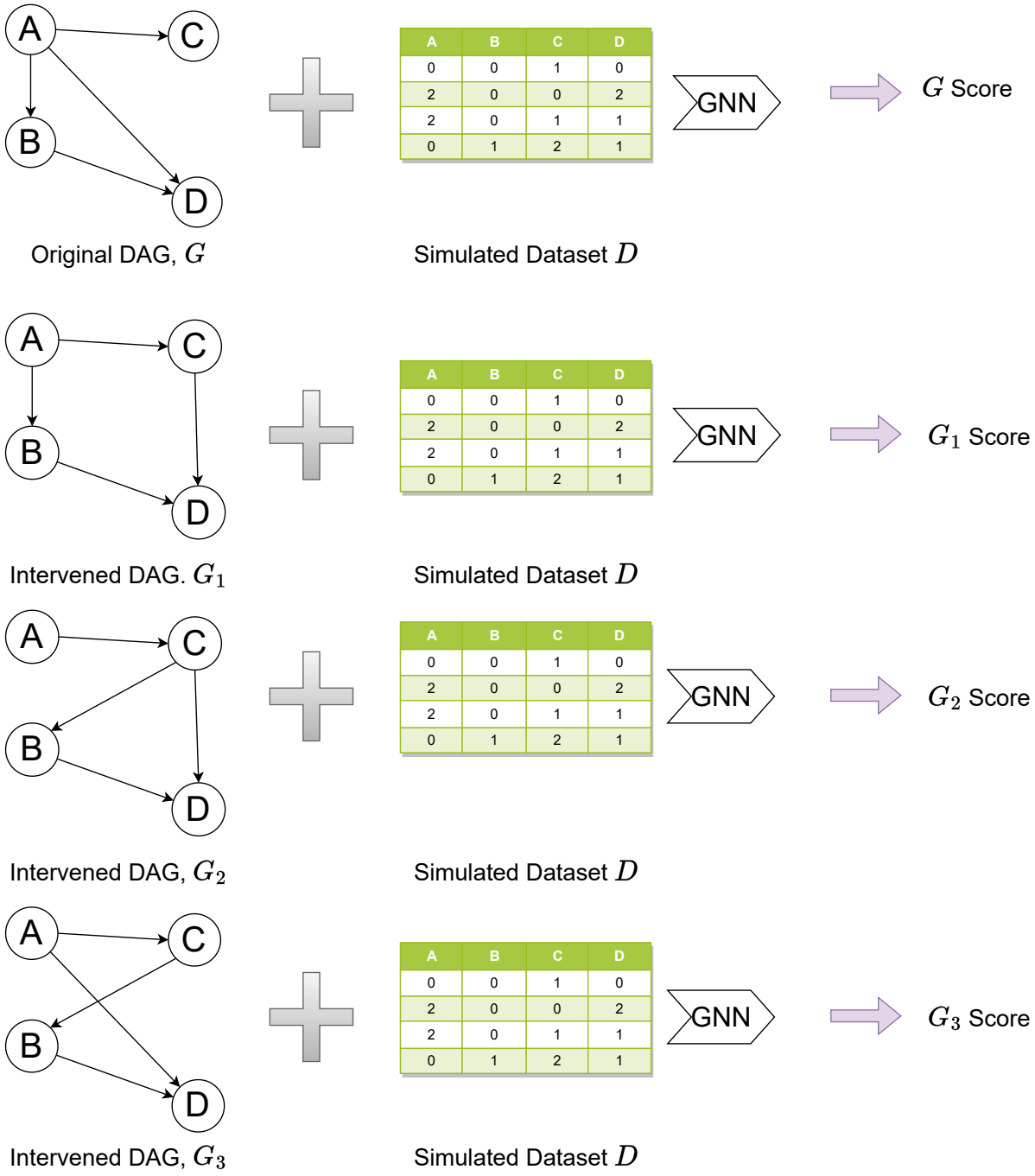


Figure 4.3: The DAG scoring methodology. DAG G is the original DAG for an arbitrary SCM $\mathcal{C}$ and dataset $\mathbf{D}$ is generated by forward sampling from $\mathcal{C}$. DAGs G_1, G_2, G_3 are the resultant DAGs after introducing interventions. The same dataset D is used to evaluate the performance of all DAGs after message passing.

Table 4.1: Dataset Charachteristics

Network Size	Network Name	# Nodes	# Arcs	# Parameters	Target Node
Small	Cancer [53]	5	4	10	"Dyspnoea"
Small	Asia [56]	8	8	18	"dysp"
Small	Sachs [79]	11	17	178	"Akt"
Medium	Child [86]	20	25	230	"GruntingReport"
Medium	Alarm [13]	37	46	509	"BP"
Medium	Water [45]	32	66	10,083	"CNON_12_45"
Large	Hailfinder [4]	56	66	2,656	"R5Fcst"
Large	Hepar2 [71]	70	123	1,453	"consciousness"
Very Large	Diabetes [9]	413	602	429,409	"bg_24"

Cancer: The Cancer BN, also known as the "Cancer" model, is a commonly used example of a BN in causal modelling [53]. This network represents a set of variables related to pollution, smoking, X-ray, dyspnoea, and cancer. The target node is the binary variable "dyspnoea", and the task is to predict its presence.

Asia: The Asia BN, also known as the "Asia" model, is a well-known example of a BN used in causal modelling. The Asia model was first proposed by Lauritzen and Spiegelhalter in 1988 [56] and has since been used as a benchmark dataset in various studies of BN inference and Causal Discovery algorithms. The network represents a set of variables related to lung cancer, including smoking, pollution, tuberculosis, and cancer, among others. The Asia model serves as a good illustration of the challenges in causal inference, as it involves variables with complex interdependencies and intricate causal relationships. The target node is binary variable "dysp" or "dysponea" and the task is to predict its presence.

Sachs: The Sachs BN consists of 11 nodes, representing different variables related to gene regulation and cell signalling pathways [79]. These nodes are named after the biological factors they represent: ERK, JNK, p38, Akt, PKA, PKC, p70S6K, MSK, CREB, ATF2, and Elk1. The interactions among these nodes are represented by directed edges in the BN,

with each edge indicating a causal relationship between the corresponding variables. The target node is "Akt", and can have 3 distinct states. Hence the network task is a 3-way classification task.

Child: The Child BN models the relationships among a set of variables that are thought to influence respiratory illness within children [86]. The nodes in the BN include variables such as the child's age, symptoms, and exposure to pollution, chest X-ray reports, as well as environmental factors such as the time of year and the presence of other respiratory illnesses in the community. The target node is "GruntingReport" and the prediction task is to binary prediction between "yes" and "no".

Alarm: The Alarm BN models a medical alarm system in an intensive care unit, with nodes representing different physiological variables such as blood pressure, heart rate, and respiratory rate, as well as alarms that can be triggered based on the values of these variables [13]. The Alarm network has been widely used for testing and evaluating probabilistic inference algorithms and decision-making strategies, as well as for exploring the challenges and complexities of modeling real-world systems using probabilistic graphical models [15]. The target node is "BP" or relating to "Blood Pressure". The network task is to predict whether the blood pressure is low, normal, or high.

Water: The Water BN models an expert-designed system for wastewater management [45]. The nodes represent the state of the system at different stages of the water treatment process. The target node here is "CNON_12_45", which takes 4 different states relating to the influent flow.

Hailfinder: The Hailfinder BN is a model designed to predict hailstorms [4]. It was developed by the National Center for Atmospheric Research (NCAR) and uses data from weather stations, radars, and satellites to make its predictions. The network consists of nodes representing different weather variables, such as temperature, humidity, and wind direction. The model is trained on historical data of hailstorms and their associated weather conditions, allowing it to learn the complex relationships between different variables and how they contribute to the weather forecast. The target node is "5Fcst" or the forecast for Region 5 (Denver), which takes 3 states - Nil, Significant or Severe hailstorms.

Hepar2: The Hepar2 is BN for diagnosis of liver disorders [71]. The model was generated based on 570 patient records at the Gastroenterological Clinic of the Institute of Food and Feeding in Warsaw. The variables can be divided into three groups: symptoms reported by patients, objective evidence by physicians, and laboratory results. The target node is "consciousness" and the task is to predict its presence.

Diabetes: The Diabetes BN is a differential equation model of carbohydrate metabolism [9]. This network is a rule-based system for insulin therapy to control blood glucose levels. The base model is a discrete-time model with a one-hour time step, that describes the factors affecting the patient blood glucose. A 24-hour model is then constructed by concatenating the one-hour models. The target node is "bg__24" or the blood glucose levels over a 24-hour time slice. This variable takes 11 states relating to the glucose level.

4.3 Implementation Details

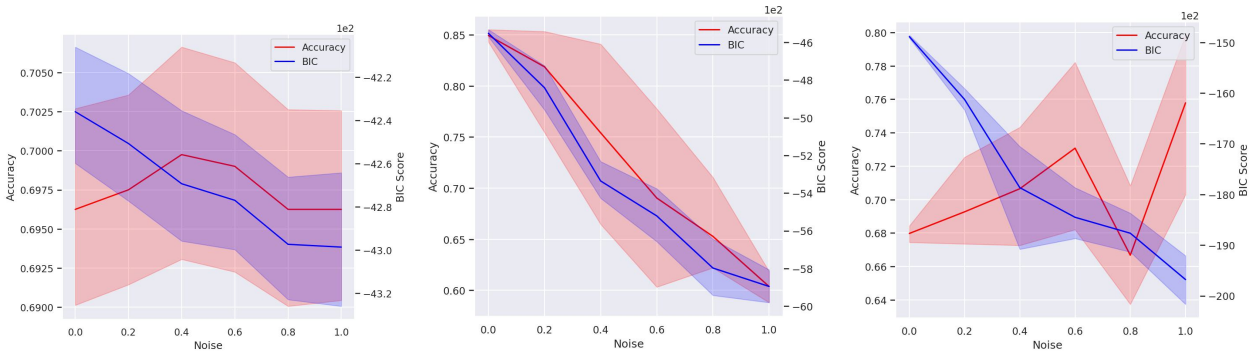
The target node for each BN described above is listed in the last column of Table 4.1. The target node is chosen based upon topological ordering of the DAG [99] (refer Appendix A for more details). For each BN, we generate 10,000 samples by forward sampling. Out of these, 6,000 samples constitute the training set, and the testing and validation set comprises 2,000 samples each. For each BN, we create additional interventional DAGs. These are characterized by increasing amounts of noise or perturbation. As the noise increases, the DAG structure deviates farther from the original. The perturbation is designed to keep the number of edges constant as well as to maintain acyclicity in the graph. Throughout our experiments, we utilize six noise levels ranging from 0 to 1. At noise level 0.2, 20% of the edges in the original DAG are replaced by an equivalent number of random edges, and so on. We train a new model from scratch for each BN and each noise level. The evaluation metric is classification accuracy. Overall we train and evaluate $9 \times 6 = 54$ models. Our model is a 2-layer GCN followed by MLP consisting of two dense layers with ReLU [67] activation. The number of neurons in the GCN varies as per the size of the BN. Similarly, the number of neurons in the MLP varies with the number of terminal nodes in the DAG and the number of states the target node takes. The classes for the classification task correspond to the target node/variable states. Before feeding into the model, each discrete variable is mapped onto a 16-dimensional feature space, by an embedding layer.

We train the networks using Adam [51] optimizer for 50 epochs, with an initial learning rate set to $1e^{-2}$, and a batch size of 256. We also use 0.5 dropout in between the GCN and MLP layers and a learning rate scheduling paradigm that reduces the learning rate by a decay factor of 0.3 if there has been no improvement in the classification accuracy for 4 epochs. To avoid overfitting we implement early stopping with the patience of 10 epochs. For each DAG (original and intervened) we compute the Bayesian Information Criteria (BIC) score

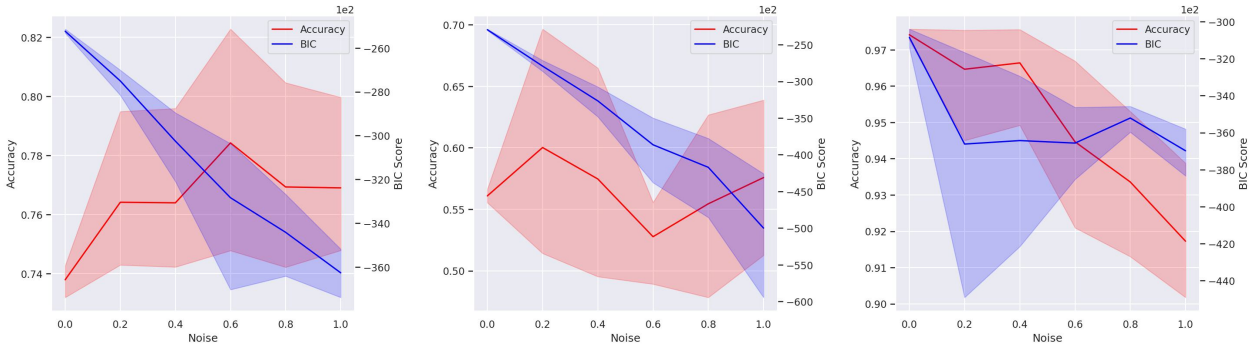
[69]. BIC is widely used for evaluating structure learning algorithms. The GNN model is implemented by utilizing PyTorch [74] and pytorch_geometric [26] deep learning frameworks, and data generation for BNs, perturbation operations, and BIC evaluation is implemented with the pgmpy library [10]. In Section 4.4 we report the BIC as well as the testing accuracy for all DAGs and also compute the correlation between noise, BIC, and accuracy.

4.4 Results

The results are illustrated in Figure 4.4 and Figure 4.5. We plot the accuracy and the BIC score against increasing noise levels over eight trials. The solid red line represents the mean downstream prediction accuracy and the shaded red region is the 95% confidence interval. Similarly, the solid blue line is the BIC score and the shaded blue region is its 95% confidence interval. A higher BIC score (less negative) indicates a better model fit. Hence, ideally, the BIC score should decrease as we increase the noise. Figure 4.4 shows how the trend of the metrics for small and medium-sized BNs as delineated in Table 4.1. We can observe that the BIC score is downward trending for five out of the six BNs. Pearson correlation test reveals that the BIC score is negatively correlated for Cancer, Asia, Sachs, Child, and Alarm and is significant at $p < 0.05$ (refer to Table A.1 in Appendix A). However, there seems to be no such trend in the accuracy curves. Only Asia and Water BNs showcase a negative correlation between accuracy and noise that is statistically significant at $p < 0.05$. Figure 4.5 shows how the metrics trend for the remaining large and very large BNs. For these BNs there exists no correlation between the BIC score and noise levels. The accuracy curves follow the same trend as in Figure 4.4 and they are uncorrelated with the noise except only for Diabetes. While the BIC score trends are surprising, considering this to be a conventional metric for evaluating structure learning methods, one can notice that BIC becomes increasingly

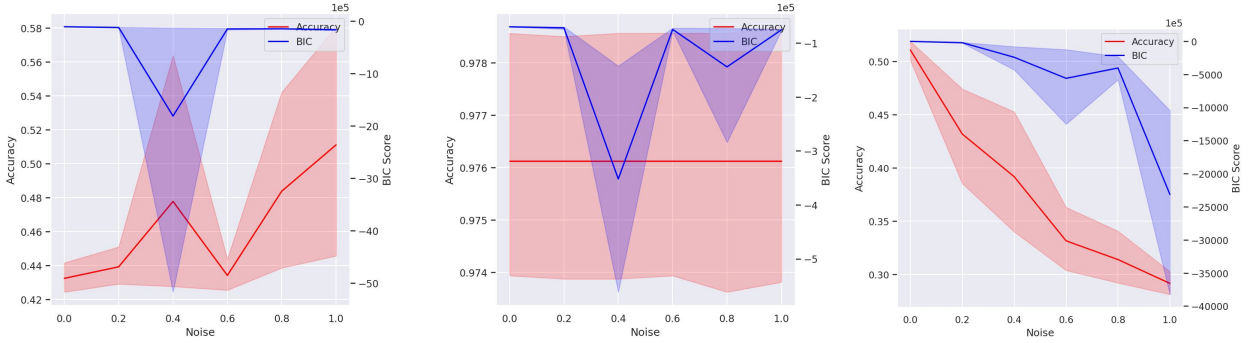


(a) Cancer (b) Asia (c) Sachs



(e) Child (f) Alarm (g) Water

Figure 4.4: Accuracy and BIC score on small and medium sized BNs



(a) Hailfinder (b) Hepar2 (c) Diabetes

Figure 4.5: Accuracy and BIC score on large and very large BNs

unreliable with an increase in the network size. Based on these results, we can conclude that downstream accuracy is not a good metric for scoring DAGs or the evaluation of structure learning methods. Hence, we cannot reject the null hypothesis (H_0) defined earlier.

4.5 Discussion

GNNs require well-defined graph structures for accurate graph representational learning, and structure learning aims to generate these graph structures. Thus in this chapter, we have explored whether GNNs can collectively address both these problems. We developed a workflow that tries to simulate Belief Propagation in BNs with message-passing networks and ran experiments to check if this workflow can be used for structure learning. The experimental results show that downstream model performance is not a good proxy for evaluating the correctness of causal structures. Possible reasoning for the observed results is the modularity of interventions or the *do* operation [75]. An SCM can be said to be a collection of local partial mechanism f_{ij} of any structural equation [109]. An SCM thus considers a particular mechanism for each variable or node in the graph. This directly relates to the atomic property of the *do* operator. GNNs however have the property of shared computations which might violate the localized nature of interventions. The update function (which updates the node embeddings) is shared amongst all nodes in the graph. This thus creates a discrepancy when trying to simulate belief propagation in BNs. Additionally, GNNs use message passing to learn representations of the graph, while BP in BNs uses message passing to perform probabilistic inference. Hence even though message passing and belief propagation schemes both work on graph-structured data it might not be prudent to directly equate them.

When we manipulate the DAG with noise, we are causing interventions in the SCM. Consider the DAGs G and G_2 in Figure 4.3. The intervened DAG G_2 has additional edges between nodes C and D and between nodes C and B, and it is missing an edge between nodes A and D. *i.e.*, our intervention is causing node C to send a message or propagate belief to nodes D and B and it is obstructing the message from node A to node D. However, the dataset is generated based on the original DAG G and the data has no observations for

this new scenario created by us. Feeding such data to a GNN has the potential to result in stochastic behaviour since we are essentially trying to answer interventional questions from observational data. This violates PCH [76] and could be another possible explanation for our results.

We also see that the BIC score, while quite accurate for smaller networks, becomes inconsistent as the size of the BNs grows. It is also computationally expensive to calculate the BIC score for large networks. Our experiments on a massive BN, MUNIN, [8] with 1041 nodes and 80592 parameters were unfeasible to run. The BIC score has a time complexity of $O(NM)$ where N is the number of data points and M is the number of parameters in the network. Hence there is a need to find more efficient metrics for structure learning [20].

Chapter 5

Conclusions

In this thesis, we explore the utility of geometric deep learning for healthcare applications. Our motivation for using GNNs is the flexibility they offer in representing real-world information as graphs, where nodes and edges can represent different entities and their relationships. The learnings from our studies may facilitate the development of novel graph representation techniques for a variety of real-world settings. The foremost conclusive deliverables and prospects of our work are listed as follows:

1. We develop `CheXRelNet`, a novel graph-based deep learning model for identifying changing pathology in longitudinal CXR examinations. `CheXRelNet` utilizesz structural information that is crucial to diagnosis. Representing CXR regions as nodes within the graph and allowing message propagation within is clinically intuitive, in the sense that any organ within the human body is influenced by the changes occurring in the surrounding regions. This work transforms the biomedical image analysis problem into a scene understanding task and opens up interesting possibilities in graph representation learning for medical images. Our novel graph construction methodology can be further expanded to other problems wherein there are multiple interacting entities and the interactions between them are significant for efficient representation learning.

2. We propose a novel GNN-based workflow for causal structure learning that tries to score DAGs by using downstream prediction accuracy as a proxy for the structural

score. Although the results of this study did not support our hypothesis, we believe that the findings have provided valuable insights into the limitations and challenges of the proposed approach. The lack of correlation between prediction accuracy and the DAG structure suggests that it is not prudent to directly equate belief propagation in BNs to message passing in GNNs. Despite this setback, we believe that our study adds to the growing body of literature in this area and lays the groundwork for future research in the intersection of GNNs and SCMs.

Bibliography

[1] Bilinear interpolation. URL https://en.wikipedia.org/wiki/Bilinear_interpolation.

[2] The convergence of healthcare and technology. https://blog.ml.cmu.edu/2020/08/31/1-domain-knowledge/. Accessed: 2023-03-29.

[3] The convergence of healthcare and technology. https://www.rbccm.com/en/gib/healthcare/episode/the_healthcare_data_explosion. Accessed: 2023-03-29.

[4] Bruce Abramson, John Brown, Ward Edwards, Allan Murphy, and Robert L Winkler. Hailfinder: A bayesian system for forecasting severe weather. *International Journal of Forecasting*, 12(1):57–71, 1996.

[5] Anuja Kumar Acharya and Rajalakshmi Satapathy. A deep learning based approach towards the automatic diagnosis of pneumonia from chest radio-graphs. *Biomedical and Pharmacology Journal*, 13(1):449–455, 2020.

[6] Ravi Aggarwal, Viknesh Sounderajah, Guy Martin, Daniel SW Ting, Alan Karthikesalingam, Dominic King, Hutan Ashrafian, and Ara Darzi. Diagnostic accuracy of deep learning in medical imaging: a systematic review and meta-analysis. *NPJ digital medicine*, 4(1):65, 2021.

[7] David Ahmedt-Aristizabal, Mohammad Ali Armin, Simon Denman, Clinton Fookes, and Lars Petersson. Graph-based deep learning for medical diagnosis and analysis: past, present and future. *Sensors*, 21(14):4758, 2021.

[8] Steen Andreassen, Finn V Jensen, Stig Kjær Andersen, B Falck, U Kjærulff, M Woldbye, AR Sørensen, A Rosenfalck, and F Jensen. Munin: an expert emg assistant. In *Computer-aided electromyography and expert systems*, pages 255–277. Pergamon Press, 1989.

[9] Steen Andreassen, Roman Hovorka, Jonathan Benn, Kristian G Olesen, and Ewart R Carson. A model-based approach to insulin adjustment. In *AIME 91: Proceedings of the Third Conference on Artificial Intelligence in Medicine, Maastricht, June 24–27, 1991*, pages 239–248. Springer, 1991.

[10] Ankur Ankan and Abinash Panda. pgmpy: Probabilistic graphical models using python. In *Proceedings of the 14th Python in Science Conference (SCIPY 2015)*. Citeseer, 2015.

[11] Elias Bareinboim, Juan D Correa, Duligur Ibeling, and Thomas Icard. On pearl's hierarchy and the foundations of causal inference. In *Probabilistic and causal inference: the works of judea pearl*, pages 507–556. 2022.

[12] Peter Battaglia, Jessica Blake Chandler Hamrick, Victor Bapst, Alvaro Sanchez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Caglar Gulcehre, Francis Song, Andy Ballard, Justin Gilmer, George E. Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Jayne Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. Relational inductive biases, deep learning, and graph networks. *arXiv*, 2018. URL https://arxiv.org/pdf/1806.01261.pdf.

[13] Ingo A Beinlich, Henri Jacques Suermondt, R Martin Chavez, and Gregory F Cooper. The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks. In *AIME 89: Second European Conference on Artificial Intelligence*

in Medicine, London, August 29th–31st 1989. Proceedings, pages 247–256. Springer, 1989.

[14] Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria de la Iglesia-Vayá. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797, 2020.

[15] Micheline Chalhoub-Deville. The intersection of test impact, validation, and educational reform policy. *Annual Review of Applied Linguistics*, 29:118–131, 2009.

[16] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68. Springer, 2016.

[17] Max Chickering, David Heckerman, and Chris Meek. Large-sample learning of bayesian networks is np-hard. *Journal of Machine Learning Research*, 5:1287–1330, 2004.

[18] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*, pages 424–432. Springer, 2016.

[19] Joseph Paul Cohen, Joseph D. Viviano, Paul Bertin, Paul Morrison, Parsa Torabian, Matteo Guarrera, Matthew P Lungren, Akshay Chaudhari, Rupert Brooks, Mohammad Hashir, and Hadrien Bertrand. TorchXRayVision: A library of chest X-ray datasets and models. In *Medical Imaging with Deep Learning*, 2022. URL https://github.com/mlmed/torchxrayvision.

[20] Nicandro Cruz-Ramírez, Héctor-Gabriel Acosta-Mesa, Rocío-Erandi Barrientos-Martínez, and Luis-Alonso Nava-Fernández. How good are the bayesian information criterion and the minimum description length principle for model selection? a bayesian network analysis. In *MICAI 2006: Advances in Artificial Intelligence: 5th Mexican International Conference on Artificial Intelligence, Apizaco, Mexico, November 13-17, 2006. Proceedings 5*, pages 494–504. Springer, 2006.

[21] Guillem Cucurull, Konrad Wagstyl, Arantxa Casanova, Petar Veličković, Estrid Jakobsen, Michal Drozdzal, Adriana Romero, Alan Evans, and Yoshua Bengio. Convolutional neural networks for mesh-based parcellation of the cerebral cortex. In *Medical Imaging with Deep Learning*, 2018.

[22] Mathias Drton and Marloes H Maathuis. Structure learning in graphical modeling. *Annual Review of Statistics and Its Application*, 4:365–393, 2017.

[23] Frederick Eberhardt and Richard Scheines. Interventions and causal inference. *Philosophy of science*, 74(5):981–995, 2007.

[24] Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. A fair comparison of graph neural networks for graph classification. *arXiv preprint arXiv:1912.09893*, 2019.

[25] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017.

[26] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *Proceedings of the ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[27] Hongyang Gao and Shuiwang Ji. Graph u-nets. In *international conference on machine learning*, pages 2083–2092. PMLR, 2019.

[28] Antonio Garcia-Uceda Juarez, Raghavendra Selvan, Zaigham Saghir, and Marleen de Bruijne. A joint 3d unet-graph neural network-based method for airway segmentation from chest cts. In *Machine Learning in Medical Imaging: 10th International Workshop, MLMI 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Proceedings 10*, pages 583–591. Springer, 2019.

[29] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[30] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[31] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[32] Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

[33] Karthik Gopinath, Christian Desrosiers, and Herve Lombaert. Graph convolutions on spectral embeddings for cortical surface parcellation. *Medical image analysis*, 54: 297–305, 2019.

[34] Weiwei Gu, Fei Gao, Xiaodan Lou, and Jiang Zhang. Link prediction via graph attention network. *arXiv preprint arXiv:1910.04807*, 2019.

[35] Ruihua Guo, Kalpdrum Passi, and Chakresh Kumar Jain. Tuberculosis diagnostics and localization in chest x-rays via deep learning models. *Frontiers in Artificial Intelligence*, page 74, 2020.

[36] Xiaojie Guo and Liang Zhao. A systematic survey on deep generative models for graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[37] William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*, 2017.

[38] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[39] Katharine E. Henry, David N. Hager, Peter J. Pronovost, and Suchi Saria. A targeted real-time early warning score (trewscore) for septic shock. *Science Translational Medicine*, 7(299):299ra122–299ra122, 2015. doi: 10.1126/scitranslmed.aab3719. URL https://www.science.org/doi/abs/10.1126/scitranslmed.aab3719.

[40] Paul W Holland. Statistics and causal inference. *Journal of the American statistical Association*, 81(396):945–960, 1986.

[41] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[42] Xiaoshui Huang, Fujin Zhu, Lois Holloway, and Ali Haidar. Causal discovery from incomplete data using an encoder and reinforcement learning. *arXiv preprint arXiv:2006.05554*, 2020.

[43] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al.

Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 33, pages 590–597, 2019.

[44] Stefan Jaeger, Sema Candemir, Sameer Antani, Yì-Xiáng J Wáng, Pu-Xuan Lu, and George Thoma. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative imaging in medicine and surgery*, 4(6):475, 2014.

[45] FV Jensen, U Kjærulff, KG Olesen, and J Pedersen. Et forprojekt til et ekspertsystem for drift af spildevandsrensning. Technical report, Technical report, Judex Datasystemer A/S, Aalborg, Danmark, 1989.

[46] Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, et al. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, pages 1–8, 2019.

[47] Alexandros Karargyris, Satyananda Kashyap, Ismini Lourentzou, Joy T Wu, Arjun Sharma, Matthew Tong, Shafiq Abedin, David Beymer, Vandana Mukherjee, Elizabeth A Krupinski, et al. Creation and validation of a chest x-ray dataset with eye-tracking and report dictation for ai development. *Scientific data*, 8(1):1–18, 2021.

[48] Gaurang Karwande, Amarachi B Mbakwe, Joy T Wu, Leo A Celi, Mehdi Moradi, and Ismini Lourentzou. Chexrelnet: An anatomy-aware model for tracking longitudinal relationships between chest x-rays. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part I*, pages 581–591. Springer, 2022.

[49] Rohan Khera, Julian Haimovich, Nathan C Hurley, Robert McNamara, John A Spertus, Nihar Desai, John S Rumsfeld, Frederick A Masoudi, Chenxi Huang, Sharon-Lise

Normand, et al. Use of machine learning models to predict death after acute myocardial infarction. *JAMA cardiology*, 6(6):633–641, 2021.

[50] Minki Kim and Byoung-Dai Lee. Automatic lung segmentation on chest x-rays using self-attention deep neural network. *Sensors*, 21(2):369, 2021.

[51] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[52] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[53] Kevin B Korb and Ann E Nicholson. *Bayesian artificial intelligence*. CRC press, 2010.

[54] Trent Kyono, Yao Zhang, and Mihaela van der Schaar. Castle: Regularization via auxiliary causal graph discovery. *Advances in Neural Information Processing Systems*, 33:1501–1512, 2020.

[55] Paras Lakhani and Baskaran Sundaram. Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, 284(2):574–582, 2017.

[56] Steffen L Lauritzen and David J Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988.

[57] Thuc Duy Le, Tao Hoang, Jiuyong Li, Lin Liu, Huawen Liu, and Shu Hu. A fast pc algorithm for high dimensional causal discovery with multi-core pcs. *IEEE/ACM transactions on computational biology and bioinformatics*, 16(5):1483–1495, 2016.

[58] Matthew D Li, Nishanth T Arun, Mishka Gidwani, Ken Chang, Francis Deng, Brent P Little, Dexter P Mendoza, Min Lang, Susanna I Lee, Aileen O'Shea, et al. Automated assessment and tracking of covid-19 pulmonary disease severity on chest radiographs using convolutional siamese neural networks. *Radiology: Artificial Intelligence*, 2(4), 2020.

[59] Matthew D Li, Ken Chang, Ben Bearce, Connie Y Chang, Ambrose J Huang, J Peter Campbell, James M Brown, Praveer Singh, Katharina V Hoebel, Deniz Erdoğmuş, et al. Siamese neural networks for continuous disease severity evaluation and change detection in medical imaging. *NPJ digital medicine*, 3(1):1–9, 2020.

[60] Yujia Li, Oriol Vinyals, Chris Dyer, Razvan Pascanu, and Peter Battaglia. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018.

[61] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[62] Jingyu Liu, Gangming Zhao, Yu Fei, Ming Zhang, Yizhou Wang, and Yizhou Yu. Align, attend and locate: Chest x-ray diagnosis via contrast induced attention network with limited supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10632–10641, 2019.

[63] Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Conference on Causal Learning and Reasoning*, pages 509–525. PMLR, 2022.

[64] Arunit Maity, Tusshaar R Nair, Shaanvi Mehta, and P Prakasam. Automatic lung parenchyma segmentation using a deep convolutional neural network from chest x-rays. *Biomedical Signal Processing and Control*, 73:103398, 2022.

[65] Dimitris Margaritis. Learning bayesian network model structure from data. Technical report, Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science, 2003.

[66] Fred A Mettler Jr, Mythreyi Bhargavan, Keith Faulkner, Debbie B Gilley, Joel E Gray, Geoffrey S Ibbott, Jill A Lipoti, Mahadevappa Mahesh, John L McCrohan, Michael G Stabin, et al. Radiologic and nuclear medicine studies in the united states and worldwide: frequency, radiation dose, and comparison with other radiation sources—1950–2007. *Radiology*, 253(2):520–531, 2009.

[67] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[68] Nasrullah Nasrullah, Jun Sang, Mohammad S Alam, Muhammad Mateen, Bin Cai, and Haibo Hu. Automated lung nodule detection and classification using deep learning combined with multiple strategies. *Sensors*, 19(17):3722, 2019.

[69] Andrew A Neath and Joseph E Cavanaugh. The bayesian information criterion: background, derivation, and applications. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(2):199–203, 2012.

[70] Dong Yul Oh, Jihang Kim, and Kyong Joon Lee. Longitudinal change detection on chest x-rays using geometric correlation maps. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 748–756. Springer, 2019.

[71] Agnieszka Onisko. Probabilistic causal models in medicine: Application to diagnosis of liver disorders. In *Ph. D. dissertation, Inst. Biocybern. Biomed. Eng., Polish Academy Sci., Warsaw, Poland*, 2003.

[72] Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407*, 2018.

[73] Sarah Parisot, Sofia Ira Ktena, Enzo Ferrante, Matthew Lee, Ricardo Guerrero, Ben Glocker, and Daniel Rueckert. Disease prediction using graph convolutional networks: application to autism spectrum disorder and alzheimer's disease. *Medical image analysis*, 48:117–130, 2018.

[74] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, pages 8026–8037, 2019.

[75] Judea Pearl. *Causality*. Cambridge university press, 2009.

[76] Judea Pearl. Causal inference. *Causality: objectives and assessment*, pages 39–58, 2010.

[77] Francesco Piccialli, Vittorio Di Somma, Fabio Giampaolo, Salvatore Cuomo, and Giancarlo Fortino. A survey on deep learning in medicine: Why, how and when? *Information Fusion*, 66:111–137, 2021.

[78] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.

[79] Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan.

Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

[80] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

[81] Marco Scutari and Jean-Baptiste Denis. *Bayesian Networks with Examples in R*. Chapman and Hall, Boca Raton, 2nd edition, 2021. ISBN 978-0367366513.

[82] Ankita Shelke, Madhura Inamdar, Vruddhi Shah, Amanshu Tiwari, Aafiya Hussain, Talha Chafekar, and Ninad Mehendale. Chest x-ray classification using deep learning for automated covid-19 screening. *SN computer science*, 2(4):1–9, 2021.

[83] Li Shen, Laurie R Margolies, Joseph H Rothstein, Eugene Fluder, Russell McBride, and Weiva Sieh. Deep learning to improve breast cancer detection on screening mammography. *Scientific reports*, 9(1):12495, 2019.

[84] Seung Yeon Shin, Soochahn Lee, Il Dong Yun, and Kyoung Mu Lee. Deep vessel segmentation by learning graphical connectivity. *Medical image analysis*, 58:101556, 2019.

[85] Johnatan Carvalho Souza, João Otávio Bandeira Diniz, Jonnison Lima Ferreira, Giovanni Lucca França da Silva, Aristofanes Correa Silva, and Anselmo Cardoso de Paiva. An automatic method for lung segmentation and reconstruction in chest x-ray using deep neural networks. *Computer methods and programs in biomedicine*, 177:285–296, 2019.

[86] David J Spiegelhalter. Learning in probabilistic expert systems. *Bayesian statistics*, 4:447–465, 1992.

[87] Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. *Causation, prediction, and search*. MIT press, 2000.

[88] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[89] Roger Stark. The looming doctor shortage. *Washington Policy Center*, 2011.

[90] Petre Stoica and Yngve Selen. Model-order selection: a review of information criterion rules. *IEEE Signal Processing Magazine*, 21(4):36–47, 2004.

[91] Kenji Suzuki. Overview of deep learning in medical imaging. *Radiological physics and technology*, 10(3):257–273, 2017.

[92] Jodie B Ullman and Peter M Bentler. Structural equation modeling. *Handbook of Psychology, Second Edition*, 2, 2012.

[93] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[94] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[95] Atul Kumar Verma, Rahul Saxena, Mahipal Jadeja, Vikrant Bhateja, and Jerry Chun-Wei Lin. Bet-gat: An efficient centrality-based graph attention model for semi-supervised node classification. *Applied Sciences*, 13(2):847, 2023.

[96] Matthew J Vowels, Necati Cihan Camgoz, and Richard Bowden. D'ya like dags? a survey on structure learning and causal discovery. *ACM Computing Surveys*, 55(4): 1–36, 2022.

[97] Xiaoqiang Wang, Yali Du, Shengyu Zhu, Liangjun Ke, Zhitang Chen, Jianye Hao, and Jun Wang. Ordering-based causal discovery with reinforcement learning. *arXiv preprint arXiv:2105.06631*, 2021.

[98] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2097–2106, 2017.

[99] Wikipedia contributors. Topological sorting — Wikipedia, the free encyclopedia, 2023. URL https://en.wikipedia.org/w/index.php?title=Topological_sorting&oldid=1146072031. [Online; accessed 1-April-2023].

[100] D Randall Wilson and Tony R Martinez. The need for small learning rates on large problems. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, volume 1, pages 115–119. IEEE, 2001.

[101] Jenna Wong, Mara Murray Horwitz, Li Zhou, and Sengwee Toh. Using machine learning to identify health outcomes from electronic health record data. *Current epidemiology reports*, 5:331–342, 2018.

[102] Joy Wu, Yaniv Gur, Alexandros Karargyris, Ali Bin Syed, Orest Boyko, Mehdi Moradi, and Tanveer Syeda-Mahmood. Automatic bounding box annotation of chest x-ray data for localization of abnormalities. In *Proceedings of the 17th International Symposium on Biomedical Imaging (ISBI)*, pages 799–803. IEEE, 2020.

[103] Joy T Wu, Ali Syed, Hassan Ahmad, et al. Ai accelerated human-in-the-loop structuring of radiology reports. In *Proceedings of the Americal Medical Informatics Association (AMIA) Annual Symposium*, 2020.

[104] Joy T Wu, Nkechinyere Nneka Agu, Ismini Lourentzou, Arjun Sharma, Joseph Alexander Paguio, Jasper Seth Yao, Edward Christopher Dee, William G Mitchell, Satyananda Kashyap, Andrea Giovannini, et al. Chest imagenome dataset for clinical reasoning. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021.

[105] Kevin Xia, Kai-Zhan Lee, Yoshua Bengio, and Elias Bareinboim. The causal-neural connection: Expressiveness, learnability, and inference. *Advances in Neural Information Processing Systems*, 34:10823–10836, 2021.

[106] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4582–4591, 2017.

[107] Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. Causalvae: Disentangled representation learning via neural structural causal models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9593–9602, 2021.

[108] Yue Yu, Jie Chen, Tian Gao, and Mo Yu. Dag-gnn: Dag structure learning with graph neural networks. In *International Conference on Machine Learning*, pages 7154–7163. PMLR, 2019.

[109] Matej Zecevic, Devendra Singh Dhami, Petar Velickovic, and Kristian Kersting. Relating graph neural networks to structural causal models. *arXiv preprint arXiv:2109.04173*, 2021.

[110] Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. *Advances in neural information processing systems*, 31, 2018.

[111] Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski. Graph convolutional networks: a comprehensive review. *Computational Social Networks*, 6(1):1–23, 2019.

[112] Xiang Zhang, Marko Zeman, Theodoros Tsiligkaridis, and Marinka Zitnik. Graph-guided network for irregularly sampled multivariate time series. *arXiv preprint arXiv:2110.05357*, 2021.

[113] Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

[114] Shengyu Zhu, Ignavier Ng, and Zhitang Chen. Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477*, 2019.

Appendices

Appendix A

A.1 Topological Sorting

Topological sorting is a common operation in graph theory that involves arranging the vertices of a DAG into a linear ordering such that for every directed edge (u, v), vertex u comes before vertex v in the ordering. This linear ordering is known as a topological sort or topological order of the DAG.

The topological ordering of the DAG results in an upper or lower triangular adjacency matrix. Having the original DAG topologically sorted is essential to performing interventions in the DAG and yet maintaining the property of acyclicity. Once the DAG is topologically sorted we select the last node in the sorted order as the target node which defines the GNN prediction task. This ensures that both the graphs defined by $A_{G_\mathcal{V}}$ and $A_{G_{\mathcal{V}'}}$ are DAGs.

A.2 Pearson Correlation Test on Accuracy, BIC score, and noise level

Table A.1 shows the Pearson correlation coefficient for accuracy, BIC score, and noise. The statistics at $p < 0.05$ significance are highlighted in bold.

Table A.1: Pearson Correlation test on accuracy, BIC score, and noise level. The p-values for each correlation statistic are shown in parentheses.

Network Name	**Noise - Accuracy**	**Noise - BIC**	**BIC - Accuracy**
Cancer	-0.02 (0.87)	-0.47 (**7.0e-04**)	0.42 (**3e-03**)
Asia	-.0.72 (**7.06e-09**)	-0.93 (**1.0e-21**)	0.73 (**3.7e-09**)
Sachs	0.26 0.07	-0.87 **6.79e-16**	0.32 **0.02**
Child	0.21 0.14	-0.81 **3.4e-12**	0.18 0.21
Alarm	-0.05 (0.71)	-0.79 (**2.3e-11**)	0.05 (0.69)
Water	-0.6 (**5.91e-06**)	-0.22 (0.13)	0.07 (0.64)
Hailfinder	0.31 (**0.03**)	0.03 (0.82)	-0.5 (**1.4e-05**)
Hepar2	0.0 (0.99)	0.009 (0.95)	0.15 (0.31)
Diabetes	-0.83 (**1.50e-11**)	-0.53 (**2.3e-4**)	0.44 (**0.004**)