MODELING PHARMACOLOGICAL EFFECTS THROUGH MULTI GRAPH AND MULTI-RELATION GRAPH EMBEDDING

AMIR JALILIFARD

MODELING PHARMACOLOGICAL EFFECTS THROUGH MULTI GRAPH AND MULTI-RELATION GRAPH EMBEDDING

Tese apresentado ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

Orientador: Adriano Alonso Veloso

Belo Horizonte

Fevereiro de 2023

AMIR JALILIFARD

MODELING PHARMACOLOGICAL EFFECTS THROUGH MULTI GRAPH AND MULTI-RELATION GRAPH EMBEDDING

Thesis presented to the Graduate Program in Computer Science of the Universidade Federal de Minas Gerais in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Advisor: Adriano Alonso Veloso

Belo Horizonte

February 2023

Acknowledgment

First and formost, an immense thank to my supervisor, Adriano Alonso Veloso, for him providing invaluable guidance and feedbacks, and challenging me to grow as a scientist. I would like to give my deepest appreciation to the Prof. Estevam Rafael Hruschka Júnior and Prof. Renato Assunção as they changed my mindset forever and instilled enthusiasm, curiosity and love of learning in my heart. Indeed great teachers inspire hope and ignite the imagination. I would also like to extend my thanks to my fellow lab-mates in the LAboratory for Treating INformation (LATIN), Alberto Hideki Ueda, Anderson Bessa da Costa, Celso Renato França Júnior, Dehua Chen, Gabriel de Azevedo Cardoso, Rafael Glater Machado, Alef Henrique de Castro Monteiro and João Mateus de Freitas Veneroso for being a bunch of great people in and out of the lab and for sharing their knowledge with me during these years. I'm extremely grateful to the Department of Computer Science (DCC) of the Universidade Federal de Minas Gerais for giving me the opportunity of being a part of this department and learning with a group of the best teachers and scientists in the field of computer science. I also would like to dedicate this work to my unborn daughter, Eliza, and I hope that this work makes her proud when she grows up. Finally, special thanks to my father, my mother and my dear wife Ananda, as they are my heroes and this endeavor would not have been possible without their unlimited support and their unconditional love.

Resumo

Mecanismo de ação refere-se a interação bioquímica específica através da qual uma droga produz um efeito farmacológico. A modelagem matemática e computacional facilita análises multiparamétricas simultâneas de processos biológicos dinâmicos e a identificação de novos usos de medicamentos já conhecidos. Motivados pelo estudo da reposição de medicamentos, apresentamos um modelo de incorporação de gráficos de múltiplas relações que aprende representações latentes de medicamentos e doenças, para que a posição das representações de medicamentos e doenças no espaço de incorporação possa revelar candidatos de reposicionamento. Exploramos grafos de interação de droga-proteína e doença-droga, a fim de construir contextos de relações específicas a partir dos quais as representações são obtidas. Além disso, a estrutura específica de medicamentos e doenças farmacológicas permite a incorporação rápida (fast embedding), analisando apenas as vizinhanças imediatas de vértices e considerando a topologia do grafo que possibilita criação dos embeddings de forma mais eficiente. Nosso estudo revela várias propriedades interessantes do espaço de embedding de medicamentos e doenças, o que torna a seleção de candidatos na fase de hipótese mais fácil e mais confiável. Além disso, mostramos que é possível usar uma combinação de representações latentes de interações de droga-ação, droga-mecanismo e droga-alvo para prever as interações entre drogas e doenças e, em seguida, utilizar os falso-positivos para encontrar possíveis candidatos de reposicionamento. Espera-se que os resultados deste trabalho pavimentem o caminho para uma nova área na descoberta de novos usos de drogas, mostrando que doenças e medicamentos que estimulam as mesmas variáveis latentes, ocupam regiões próximas no espaço de embedding e têm mecanismos de ação semelhantes.

Abstract

The mechanism of action is a specific chemical interaction by which a drug produces its effect. It explicitly characterizes the function of a drug in cellular machinery. Mathematical and computational modeling facilitates the concurrent multi-parametric analyses of dynamic biological processes and the identification of new uses of already known drugs. Motivated by the study of drug repurposing, we present a multi-relation graph embedding model that learns latent representations of drugs and diseases so that the position of each drug and disease in the embedding space may reveal repositioning candidates. We exploit drug-protein and drug-disease interaction graphs to build relation-specific contexts from which representations are obtained. Further, the specific structure of drug-disease and drug-protein graphs enables fast embedding by analyzing only the immediate node neighborhood. Our research reveals some interesting properties of drug-disease embedding space, making the candidate selection in the hypothesis phase easier and less costly. Moreover, we show that it is possible to use the combination of latent representations of drug-action, drug-mechanism, and drug-target bipartite graphs to predict drug-disease interactions and then utilize the false positives to find possible repurposing candidates. We also studied the proximities in a model's decision space through a link prediction task to identify hidden similarities among drugs and diseases and managed to find several drug repurposing candidates. The results of the current work is expected to pave the way to a new horizon in the discovery of new uses of drugs, showing with evidences that diseases and medications which stimulate the same embedding feature and occupy nearby regions have similar mechanisms of action.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

It is known that some drugs used in the treatment of a disease may show effectiveness in treating other diseases as well. This takes place if the primary target of a drug is involved in several pathologies or when a drug affects the activity of additional targets relevant to the primary one [Pushpakom et al., 2019]. Drug development is an expensive and time-consuming process (see Figure 1.1). Developing new drug costs between $500 million to $2 billion and takes 10-17 years [Berdigaliyev and Aljofan, 2020]. Therefore, avoiding the long process of drug discovery and reducing healthcare costs is a motivation for investigating the already known pharmaceutical agents that act on multiple targets [Kumar et al., 2019]. The enormous cost and lengthy drug development process underline the need for an alternative approach.

Figure 1.1: Traditional drug discovery steps.

Drug repositioning, also known as drug repurposing, reprofiling, retasking, or therapeutic switching [Tobinick, 2009a],[Berdigaliyev and Aljofan, 2020], is a way to reduce the cost and time of developing new drugs [Tobinick, 2009b]. It is the process of identifying new indications for already approved medications. Since repositioning relies on previously approved drugs, their toxicity profiles are already known [Kato et al., 2015]. Also, because they have already passed multiple toxicity tests, drug repositioning candidates tend to be ready for clinical trials quickly [Berdigaliyev and Aljofan, 2020]. Therefore, it can decrease the traditional timeline from 10-17 years to only 3-12 years [Berdigaliyev and Aljofan, 2020]. The significance of drug repositioning would be more tangible, knowing that a small percentage of pro-

posed drugs can pass the hypothesis generation and target screening phases successfully (on average less than 10% of proposed drugs pass these phases). This rate for oncology drugs is only 5% [Kato et al., 2015]. This low success rate emphasizes the importance of drug repositioning in reducing the cost and risk of drug development. In addition, a large number of potential drugs never reach clinical testing. Furthermore, fewer than 15% of compounds entering clinical development ultimately receive approval, despite most of them being deemed safe. Therefore, repurposing is attractive and pragmatic, given drug discovery's substantial cost and time requirements.

One of the earliest drug repurposing cases is the use of reticulose for post-radiation effects in the 1950s [Yonkman, 1959],[Atkinson Jr et al., 1975], [Hurle et al., 2013]. Later in the 1980s, anti-malarial drugs were suggested as potential repurposing candidates for Rheumatoid Arthritis and Connective Tissue disease [Harris, 1981], [Mathewson, 1982]. A famous example of drug repositioning is Pfizer's Sildenafil (Viagra) which was repositioned from angina treatment to erectile dysfunction treatment in men in 1998 [Novac, 2013] and recently has been studied as a possible treatment for Age-related Macular Degeneration (AMD) [Coleman et al., 2018]. Thalidomide is another example of drug repositioning. It first was introduced as a sedative-hypnotic and then was withdrawn because of severe side effects. Later, Thalidomide was reintroduced as an anti-cancer agent and used for erythema nodosum leprosum and HIV wasting syndrome. In some clinical trials, Duloxetine, a medication for depression, was found to be effective in the treatment of stress urinary incontinence in women [Hurley et al., 2006]. Overall, between 2007 and 2009, about 30-40% of newly approved drugs were repurposed medications [Jin and Wong, 2014].

For a successful drug repositioning, the drug must have a positive effect on a different disease but, its highest value resides in that its use for the novel indication surpasses the currently available therapeutic options for that condition. Experimental approaches for drug repositioning generally involve high-throughput assays where libraries of approved compounds are tested against biological targets of interest. The effects of a large number of the Food and Drug Administration (FDA) approved compounds on gene expression have been measured on several cultured human cell lines (the Connectivity Map known as CMap), and this information have been utilized to investigate similarities between drugs' mechanism of action [Hurle et al., 2013],[Kwon et al., 2020], [Mousavi et al., 2020] (the term *mechanism of action* refers to a biochemical interaction through which a drug substance produces its pharmacological effect).

The process of drug repurposing can be performed either experimentally or computationally, also called *in silico* drug repurposing, which belongs to computational pharmacology [Shim and Liu, 2014]. In this approach, repurposing is categorized into

discovering new indications for an already known drug (drug-centric) and identifying effective drugs for a disease (disease-centric) and has the common strategy of similarity assessment between drugs and diseases [Liu et al., 2013]. The development of in silico drug repurposing and its wide use today have been made possible by two technological trends [Shim and Liu, 2014]. The first trend is generating and gathering high-throughput data from various sources, including genomics, proteomics, chemoproteomics, and phenomics. As a result, data characterizing disease phenotypes and drug profiles, and entire pathway maps have become available. The second is due to advances in computational and data sciences which have made the development of repurposing algorithms possible, along with retrospective analysis and database maintenance for experimental data [Hodos et al., 2016].

1.1 Motivation

Our work is motivated by the problem of finding drug repurposing opportunities by modeling the mechanisms of action of drugs. For example, the mechanism of action of selective serotonin reuptake inhibitors, or SSRIs, inhibits serotonin's reuptake, increasing serotonin's serotonin level in the brain and improving the individual's mood. As another example, different biological solutions can be used to decrease blood pressure chemically. A first solution is to remove the excess salt from the body, thereby decreasing the tension in the vessels. Another solution is to inhibit the vasoconstrictive signaling of a hormone. Finally, it is also possible to act directly on the cells physically, narrowing vessels and preventing their unwanted action [Ong et al., 2007]. Each of the solutions mentioned earlier requires a different mechanism of action. A drug may have several mechanisms of action, and therefore it can potentially play different roles by perturbing proteins involved in various biological processes, which are accountable for the drug poly-pharmacology [Car, 2012]. Drug repurposing is a direct application of drug polypharmacology [Zhang et al., 2016, Anighoro et al., 2014]. Sometimes a new candidate for an available drug may be identified only by chance. However, new data analytic methods and a large number of available data enable the development of systematic approaches to identify and assess drug repositioning candidates with significantly lower costs.

When using complex drug-disease-protein graph structures for drug repositioning, our primary goal is to discover the hidden and unknown relations between the components' interactions and finally use these unknown relations to facilitate the complex and time-consuming drug discovery process. We are mainly interested in understanding the

characteristics of embedding and link predictor model's decision space of drug-disease-protein interactions and using them to predict new possible candidates for an individual or a group of diseases that share similar traits.

1.2 Main contributions

This proposal focuses on learning and exploring the latent space of the graphs constructed upon the official information of drug-disease-protein, drug-action, drug-mechanisms, drug-target, and drug-indication interactions. Furthermore, we identify and assess challenges and opportunities in using these data in several tasks of drug repurposing. The main contributions of the current work are as follow:

- Proposing a new unsupervised node representation learning approach based on adaptive sampling of nodes' nearest neighbors and as a result, predicting the new uses of drugs by modeling pharmacological effects through fast and effective learning of the latent representations of drugs and diseases.
- Identifying important characteristics of embedding space of the drug-disease-protein graph and utilizing them to accelerate the process of repurposing candidate selection, and showing through this process that the latent representations of drugs and diseases in the decision space can in fact be used as distance measures of similarities between different mechanisms of action.
- Proposing the use of false negatives of a link prediction task, carried on the fused embeddings of separately learned latent representations of independent graphs of drug-action, drug-mechanism, and drug-target interactions, as a means of identifying new repositioning opportunities.

1.3 Thesis statement

This thesis aims to show, based on the evidences, that:

- The node influence factors in a graph structure can be used to learn a good quality embedding for each node while decreasing the number of training data significantly.
- The latent representation of the drug-disease-protein graph has unique properties that can be explored and utilized to find possible repositioning candidates for various diseases.

- The Shapley space, which is mainly used to interpret the model's decision-making processes, can also be used in both supervised and unsupervised tasks. This result implies the possibility of joining the concepts of predictability and interpretability.
- Drug-action, drug-mechanism, and drug-target graphs' latent representations can be learned separately and used to predict possible drug-disease interactions, which itself may lead finding new uses of already known drugs.

1.4 Thesis structure

The remainder of this thesis is organized as follows. Chapter 2 gives a brief background about the process of drug repurposing and discusses the related works. Chapter 3 gives a brief explanation of graph embedding methods. Chapter 4 details the whole process and the results of our studies on modeling pharmacological effects through multi-relation and multi-graph unsupervised embedding learning. Chapter 5 presents a summary and the conclusions for the current research. The complementary results are shown in the appendix.

Chapter 2

Related Works

In this chapter, methods used to find new drug opportunities are outlined. We first talk about the experimental approaches and then explain the computational methods. These two approaches can be used, alone or combined, to achieve a more holistic point of view and increase the chance of success in drug repurposing.

2.1 Drug repurposing strategies

Typically drug repurposing consists of three general phases:

1. Identifying the candidate molecule for a given indication (hypothesis phase).
2. Assessment of drug's effect in preclinical models.
3. Evaluation of the efficiency of the drug in clinical trials.

Among these three steps, hypothesis generation and identifying the right candidate for a disease of interest with high confidence are critical. Here is where modern methods can be helpful. A variety of experimental and computational approaches have been leveraged to implement various modalities of drug repositioning. In the following, we summarize the experimental approaches and then detail three main branches of computational methods, discussing the recent studies and research tendencies in each one.

2.1.1 Experimental methods

Experimental approaches utilize several strategies in hypothesis generation step which are as follows:

- Target-based (also known as structured-based).
- Repurposing based on transcriptional signatures.
- Drug-based.
- Literature-based.

Target-based methods rely on the idea that similar proteins have similar functionalities. Hence, similarity comparison can be used to find secondary targets of an already existing drug [Ehrt et al., 2016]. In comparison with blind search, which does not use pharmacological or biological information, target-based repurposing directly links targets with disease mechanisms, and hence the chance of drug discovery improves significantly. In addition, the target-based approaches can screen nearly all drug compounds with known chemical structures. The disadvantage, however, is that target-based methods cannot identify unknown mechanisms beyond the targets already known.

Molecular transcriptional signatures can be compared to create relations between drugs and new diseases indications. This approach explores drug-disease relations through comparing gene expression profiles between disease and drugs and provides helpful information for finding new unknown targets for the known drugs [Thorman et al., 2020]. The advantage of this method is in its ability to identify new mechanisms of action for drugs.

Drug-based methods rely on the concept that similar compounds have similar biological properties. In drug repurposing, this group of methods has been used widely to analyze and predict the activity of ligands for new targets [GNS et al., 2019], [Liu et al., 2010].

Finally, literature-based strategies aim to prepare potential data that help researchers establish a new hypothesis by studying the numerous hidden indirect relations between drug, target, and disease extracted from the scientific literature. Literature mining is carried out through Medical Subject Heading (MeSH) terms to retrieve relevant information [Lekka et al., 2011], [Sun et al., 2017].

2.1.2 Computational methods

Computational approaches are primarily data-driven. These methods utilize a variety of data sources such as drug-disease and drug-protein datasets, electronic health records, gene expressions, and chemical structures, formulating repurposing hypotheses

and then carrying out clinical tests. Computational approaches consist of three main branches:

- Computational molecular docking.
- Network mapping.
- Data mining and machine learning.

These methods significantly reduce drug development costs and enable the joint analysis of different data sources, including biomedical, genomics, and pharmacological data, which improve drug repurposing efficiency. In the following, each of the above methods is explained.

2.1.2.1 Computational molecular docking

Molecular docking [Meng et al., 2011] is a computational structure-based method that analyzes and predicts the binding site between ligand (drug) and target (receptor). If there is prior knowledge about the target receptor of the disease, multiple medications, which also have the same target, may be investigated to find the possible drug candidates. Conversely, the library of drugs with the same target as the already known one (known in terms of treating a specific disease) can be interrogated to identify novel interactions that can be taken forward with new uses. Dakshanamurthy and colleagues [Dakshanamurthy et al., 2012] used numerical methods in order to calculate the similarity between ligand descriptors and pocket using Euclidean distance [Liberti et al., 2014]. Their algorithm called TMFS (train, match, fit, streamline) manages to predict drug-target associations with 91% accuracy for the majority of drugs. Kinnings et al. [Kinnings et al., 2009] developed a chemical system of biology methods to identify off-targets of major pharmaceuticals on a proteome-wide scale. They proved the efficiency of this approach by repositioning the available drugs for Parkinson's in treating Multi-drug-resistant (MDR) and Extensively drug-resistant (XDR) tuberculosis.

Although molecular docking has been used in many researches, it suffers from three major standpoints. First, this approach is restricted to those studies in which the 3-dimensional structures of protein targets are available. Second, there is a lack of a well-formed macromolecular target database that provides valuable structural information. Finally, the usefulness of the docking algorithm has been questioned regarding the inaccuracy of the scoring function and the limitations in the predictability of the proposed methods [Pagadala et al., 2017].

2.1.2.2 Pathway or network mapping

The network-based aims to organize the relation between biological molecules to discover new properties at the network level and investigate how cellular systems simulate different properties under different conditions. A network is a graph structure where each node can be considered a molecular entity (i.e., drugs and proteins), and edges represent a direct or indirect interaction. Network mapping is specifically helpful in finding the hidden relations between drug and target in complex diseases as they are known to be caused by the perturbations of biological networks. Hu et al. [Hu and Agarwal, 2009] used the GEO dataset [Barrett et al., 2006] to construct a network containing 645 disease-disease, 5,008 disease-drug, and 164,374 drug-drug interactions and integrated this network with molecular profiles and knowledge of drugs and drug targets to infer drug repositioning opportunities. Through developing a hybrid model composed of a network component called Cancer-signaling Bridges and Bayesian Factor Regression, Jin and colleagues [Jin et al., 2012] managed to identify the off-target effects and repositioning drugs for cancer therapeutics. In another recent study, Zhou and colleagues analyzed the network proximity of drug targets and HCoV–host interactions in the human interactome and managed to identify three potential drug combinations that can help treat 2019-nCoV and SARS-CoV-2 [Zhou et al., 2020]. Barrenas et al. [Barrenäs et al., 2012] analyzed disease-specific and protein-protein networks and found that genes containing disease-associated single nucleotide polymorphisms (SNPs) tend to form highly connected clusters in protein-protein interaction networks. This characteristic was used to identify highly interconnected gene clusters from highly diverse complex diseases and find new drug repurposing opportunities. Fiscon et al. exploited SAveRUNNER [Fiscon and Paci, 2021], a recently developed network-based algorithm for drug repurposing that quantifies disease-associated genes' proximity to drug targets to identify drug candidates for Amyotrophic Lateral Sclerosis (ALS) [Fiscon et al., 2021]. This approach allowed to identify 403 repurposable drugs that were strongly associated with the disease.

2.1.2.3 Data mining and machine learning

Data mining is the automatic discovery of unknown patterns from existing information. On the other hand, machine learning aims at predicting the behavior of future coming information by learning the properties of several groups of existing knowledge. Recently, both data mining and machine learning have been vastly employed in drug repurposing. Aliper et al. [Kumar et al., 2019] used a fully connected deep neural network for training the model using transcriptional data at the gene level to predict drug

therapeutic and to use them in drug repurposing. Donner et al. [Donner et al., 2018] proposed a ligand-based approach based on learning the embedding of gene expression profiles using deep neural networks and considered it as a measure of compound functional similarity for drug repurposing. Hu and Agarwal [Hu and Agarwal, 2009] created a disease-drug network using publicly available gene expressions. Wu and colleagues developed and evaluated a network-based approach to predict effective drugs and reveal potential drug mechanisms of action at the level of signaling pathways. To do so, they constructed a patient-specific signaling network by integrating known disease-associated genes with patient-derived gene expression profiles and showed that it significantly improved the success rate of discovering effective drugs for the PC-3 prostate cancer [Wu et al., 2017].

By emerging new deep learning and graph learning methods, which manage to learn complex network structures, the network-based methods are becoming an exciting research area with promising drug discovery and repurposing results. Recently, Zeng et al [Zeng et al., 2019]. proposed a network-based deep learning approach for drug repurposing by integrating ten networks: one drug-side-effect, one drug-disease, one drug–target, and seven drug-drug networks. Using a random walk embedding learning, they first generated network representation from a heterogeneous network of ten drug-related interactions. They then fused matrices of each network into a low-dimensional feature representation common to all networks via a multimodal deep autoencoder. Finally, a collective variational autoencoder was used to predict potential associations between drugs and diseases for Alzheimer's and Parkinson's. Although the results of this study are promising, this method is complex, and it needs at least ten considerably extensive networks to train the model and to have successful predictions. Following the same direction, Yan and colleagues developed a network-based method based on bi-random walks and multiple disease-disease and drug-drug similarity measures in order to predict potential drug-disease associations [Yan et al., 2019]. Marinka and colleagues [Zitnik et al., 2018] trained a Convolutional Neural Network (CNN) on a graph, with proteins and drugs as its nodes and drug-protein and drug-drug interaction as the edges, in order to model the polypharmacy side-effects. Deepika and Geetha [Deepika and Geetha, 2018] used node representations along with a bagging Support Vector Machine (SVM) to predict drug-drug interactions. Finally, Gao et al. [Gao et al., 2018] applied a Short-term Memory Recurrent Neural Network and a Graph Convolutional Neural Network (GNN) to learn the low-dimensional representations of proteins and drug structures and engaged them in the prediction of drug-target interaction.

Traditional drug development strategies are costly, failure-prone, and expensive.

Therefore, drug repositioning has recently drawn attention and brings drugs out faster for clinical use. However, drug repositioning is a complex process involving multiple factors such as technology, commercial models, patents, investment, and market demands. Although many medical databases have been established, selecting the appropriate approach to make full use of massive amounts of medical data is still a challenge. Therefore, it is urgent to develop new approaches for drug repositioning.

Although the results of the recent studies are encouraging, to the best of our knowledge, none has considered investigating the characteristics of embedding and decision space of drug-disease interactions. Identifying similar behaviors enables predicting possible repositionable drugs and explaining why a specific drug is a good candidate for treating a specific or a group of diseases. In chapter 4, we propose a fast graph-based method for creating the latent representation of drug and disease interactions, exploring its essential properties and using them to find new repositioning opportunities. We show that the collective and individual behavior of drugs in a graph structure conveys specific characteristics that may reveal some hidden and unknown drug-disease relations that further help researchers choose candidates that are more likely useful in treating a new group of diseases. In the next chapter, the concept of graph learning and some of the well-known graph embedding approaches are detailed.

Chapter 3

Graph representation learning

Graphs are important structures used to model a set of objects, as a source of knowledge, that are related in some sense. Although vastly employed in various applications such as molecular structure, digital maps, and time-series data, researchers have recently considered graphs an interesting source of pattern recognition [Xia et al., 2021]. For example, one may be interested in learning the structure of friendships in social networks, suggesting new friends for users, or modeling and predicting drug side effects by learning drug-protein relations in a graph. No matter which application of graph learning we are interested in, the primary challenge is to find a way to incorporate the complex structure of graphs (nodes and edges) into the existing machine learning algorithms. In order to address this challenge, several algorithms have been proposed aiming at learning graph structure and creating a lower-dimensional representation of graphs (called embedding) [Angles and Gutierrez, 2008], [Grover and Leskovec, 2016a] and [Kipf and Welling, 2016a].

The idea behind representation learning is to find a mapping that embeds a set of n observations with p predictors in a low-dimensional vector space R^d called embedding space where the distance between observations can be considered as a measure of similarity. Although it was first used to find the representation of words in a set of sentences, recently, it was applied on graphs in order to find the embedding of nodes while preserving the graph's structure in a low-dimensional space [Hamilton et al., 2017b]. Graph representation learning approaches can generally be divided into shallow embedding and generalized encoder-decoder. We start by presenting a brief explanation of encoder-decoder methods, and subsequently, we will explain some representation learning approaches in detail.

3.1 Encoder-decoder

The main idea behind encoder-decoder methods is that if we can reconstruct higher-dimensional information from a lower one, the latter can act as a compact form of the higher-dimensional space and then be used in machine learning tasks. For example, the encoder may convert high dimensional data related to nodes (or even entire graph) to a vector or a single point Z in lower-dimensional space.

$$Encoder : V \rightarrow R^d \tag{3.1}$$

The decoder, on the other hand, takes the lower-dimensional representations and reconstructs the original data (see Figure 3.1).

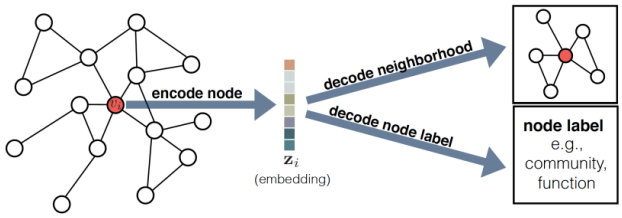


Figure 3.1: The structure of encoder-decoder. The encoder creates a lower-dimensional representation of graph and the decoder try reconstructing the original graph space. [Hamilton et al., 2017b]

To create low-dimensional representations, decoder maps pairs of embeddings to original data related to this pair based on some user-defined similarity measure (i.e. the probability of seeing the vector V_j through a random walk of length l originated from V_i). To do this, the encoder-decoder often tries to minimize a loss function:

$$\ell = \sum_{(V_i,V_j)} l(DEC(Z_i, Z_j), s_g(V_i, V_j)) \tag{3.2}$$

where s_g is user-defined similarity measure of the original pair of vertices in higher-dimensional space. Every method categorized as an encoder-decoder approach ultimately minimizes the above loss function to generate embeddings, although the way architecture of the decoder-encoder is designed and the similarity measures are different for each method. In the following, two main approaches in graph representation learning are explained.

3.2 Shallow embedding

Shallow methods are a mostly different interpretation of already existing matrix factorization whose principal goal is to decompose a matrix into the product of lower-dimensional rectangular matrices. We can generally categorize the shallow embedding algorithms into matrix factorization and random walk methods. We first explicate matrix factorization as it is the basis of the general shallow embedding, and afterward, a brief explanation of random walk methods is given.

3.2.1 Matrix factorization

Older methods like Principal Component Analysis (PCA) [Wold et al., 1987] are inefficient in reducing data dimensionality where there is a non-linear correlation between variables. Thereby, novel methods like Local Linear Embedding (LLE) [Roweis and Saul, 2000], and Isomap [Balasubramanian and Schwartz, 2002] were introduced in order to capture the non-Euclidean distances while reducing the dimensionality. Laplacian Eigenmaps [Belkin and Niyogi, 2002] is one of the most well-known matrix factorization methods. Using this approach, when the data are transformed to a lower-dimensional subspace, non-linear correlations between variables are retained under projection. Closely related to LLE and spectral clustering, given t data points in n-dimensional space, it starts by constructing a neighboring weighted graph with t nodes and a set of edges which are weighted based on their distance from that point. The projection of the nodes in the lower dimension can then be considered as the graph's representation.

The decoder is defined as L^2 norm between the pair of encodings:

$$DEC(Z_i, Z_j) = \| Z_i - Z_j \|_2^2 \tag{3.3}$$

Then the reconstruction is done by minimizing the loss function defined by equation 3.2.

In addition to Euclidean norm, recent methods [Cao et al., 2015], [Ou et al., 2016], [Ahmed et al., 2013] suppose that the strength of the relation between two nodes is proportional to the dot product of their embeddings and define the decoder as:

$$DEC(Z_i, Z_j) = Z_i^T . Z_j \tag{3.4}$$

Soon after, the mean squared error (MSE), as the loss function, is minimized in

order to realize the reconstruction.

$$\ell = \sum \parallel DEC(Z_i, Z_j) - s_g(V_i, V_j) \parallel_2^2 \tag{3.5}$$

3.2.2 Random walk approaches

The basic idea of this group of methods is the use of the probability of co-occurrence of a pair of vertices through a random walk of length T as a measure of similarity between their embeddings instead of a deterministic measure of node similarity as mentioned in factorization methods. In this approach, sequences of nodes are generated through random walks and then fed into the model to learn the node representations [Bojchevski et al., 2018], [Jalilifard et al., 2019], [Gamage et al., 2020]. Thus the decoder is defined as:

$$DEC(Z_i, Z_j) = \frac{e^{z_i^T z_j}}{\sum_{v_k \epsilon V} e^{z_i^T z_k}} \approx P_{g,T}(v_i|v_j) \tag{3.6}$$

where $p_{g,T}(v_j|v_i)$ is the probability of visiting v_j during a random walk of length T starting from v_i, and with T usually defined to be in the range $T\epsilon\{2, ..., 20\}$. The node embeddings are then calculated through minimizing the cross-entropy loss function:

$$\ell = \sum -log(DEC(Z_i, Z_j)) \tag{3.7}$$

Since computing the normalization factor is expensive, methods like Node2vec [Grover and Leskovec, 2016a] and DeepWalk [Perozzi et al., 2014a] attempt to approximate it using different strategies. For example, Node2vec employs a set of random negative samples to approximate the normalizing factor, while DeepWalk applies the hierarchical softmax technique using a binary-tree structure which accelerates the computation.

3.2.2.1 DeepWalk

The DeepWalk uses local information obtained from truncated random walks to learn latent representations by treating walks as the equivalent of sentences. The walks begin from a selected node, and then move to randomly selected neighbors for a defined number of steps. The DeepWalk seeks to optimize the node weight matrix by correctly predicting context nodes, given a center node. In other words, the model aims to maximize the probability of correctly predicting all context nodes simultaneously, given a center one. This maximization optimizes the weight matrix θ that best represents

nodes in a vector space. The parameter θ is then the result of a concatenation of input and output weight matrices $[W_{input}, W_{output}]$. The weights are then optimized during the minimization of the following loss function:

$$J(\theta; w^{(t)}) = -log \prod_{c=1}^{C} \frac{exp(W_{output_{(c)}}.h)}{\sum_{i=1}^{v} exp(W_{output_{(i)}}.h)} \tag{3.8}$$

where h is the representation of the center node, and c and i are the context nodes and their index, respectively. This loss functions can be simplified as:

$$J(\theta; w^{(t)}) = -\sum_{c=1}^{C}(W_{output_{(c)}}.h) + C \times log \sum_{i=1}^{V} exp(W_{output_{(i)}}.h) \tag{3.9}$$

The node embedding matrices $[W_{input}, W_{output}]$ are optimized through forward and backward propagations. For each iteration, the model learns to reduce prediction error by optimizing the weight matrix θ, thus acquiring higher quality embedding matrices that better capture relationships among words.

In general, the DeepWalk consists of three steps:

1. Sampling, which is done during random walks of length T. The authors of DeepWalk confirm that it is sufficient to perform 32 to 64 random walks from each node. They also show that choosing 40 steps is the ideal length for a random walk.

2. Generating random walks corresponding to each node and maximizing the probability of predicting the nodes met during random walks given the central node.

3. Concatenating the central and context representations of each node and forming the final embedding matrix.

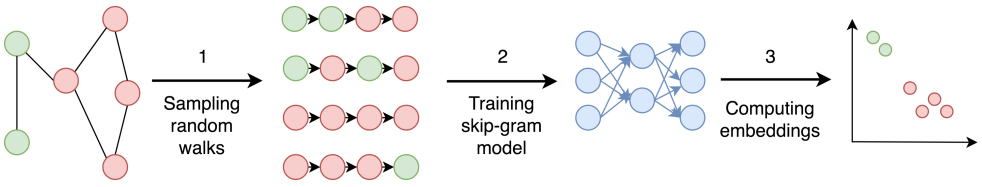


Figure 3.2: Three steps of the DeepWalk.

The main disadvantage of DeepWalk is its inability to emphasize the local and global structure of the network. However, this problem can be addressed using ideas from a more sophisticated graph embedding algorithm like Node2vec, fixing this shortcoming.

3.2.2.2 Node2vec

The Node2vec is a modification of DeepWalk with a difference in how random walks are generated. More specifically, it generates second-order biased walks where instead of looking at direct neighbors of the current state, the second-order transition applies a bias factor α (which is a function that takes as inputs the current state and the potential next state) to reweight the edge weights depending on the previous state. The α function is defined as:

$$\alpha_{P,Q}(t,u) = \begin{Bmatrix} \frac{1}{P} & if\ d_{tu} = 0 \\ 1 & if\ d_{tu} = 1 \\ \frac{1}{Q} & if\ d_{tu} = 2 \end{Bmatrix} \tag{3.10}$$

where t and u are the previous and next states, and d_{tu} is the distance between the two.

In order to generate biased random walks, the Node2vec introduces two parameters, P and Q. Parameter Q is called *exploration rate* and it defines how probable it is that a random walk discovers the unknown parts of graph, while parameter P, also known as *return rate*, is the probability of random walk returning to the previous node. Parameter P controls the discovery of microscopic view around nodes. Parameter Q on the other hand, controls the discovery of the neighborhood of a node (also called *macro-view*).

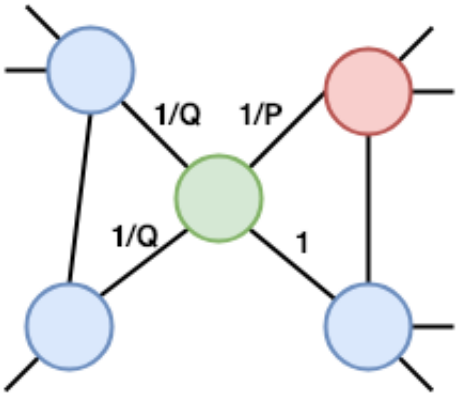


Figure 3.3: The probabilities of a random walk in Node2vec. A step is taken from the red to the green node. If the current state is the green node, the probability of going back to the red node is 1/P. Since the two blue nodes connected to the green state has no direct connection with the red node (the previous state), the probability of going to these nodes is 1/Q. On the other hand, for the blue node with a direct connection to the red one, the transition probability is set to 1.

The biased walk generation is based on the idea that if two states (nodes) are

not connected, then α is set to be $1/q$. Thus one can either increase the probability of going outward (meaning that the walk is not restricted in a localized neighborhood) by specifying a small q value or conversely restrict the walks to a local neighborhood by setting a large q value. If the two states are identical, meaning that the walk is returning back to the previous state, then α is set to be $1/p$, where p is the return parameter. Finally, if the two states are not identical and connected, alpha is set to 1.

The probability of moving from one state to another given the current one is calculated by:

$$p(u|t,v) = \frac{\alpha_{P,Q}(t,u)w(u,v)}{\sum_{u'\epsilon N_v} \alpha_{P,Q}(t,u')w(u',v)} \tag{3.11}$$

where t is the previous state, v is the current state, w is the transition weight and $u^{'}$ represents all the other possible states in the neighborhood of node v.

The Node2vec is able to create global and local strategies and learn better node representations. However, the calculation of transition probabilities adds an extra time cost. Also it is not an efficient algorithm for densely connected graphs, as its time complexity depends on the graph's branching factor b and is equal to $O(b^2)$ [Pimentel et al., 2019].

3.3 Generalized encoder-decoder

Shallow embedding approaches learn the representation of each node independently. Consequently, there is no parameter sharing (which can act as a form of regularization) [Hamilton et al., 2017b]. Another shortcoming is that this group of methods is inherently transductive. This means that the algorithm cannot generate embedding for unseen data in the training phase. Moreover, these methods consider only the pairwise relation of nodes in graph structure during the embedding creation process and fail to learn the attribute information. In order to solve these problems, recently, a new branch of representation learning methods has emerged, which follows the same logic of encoder-decoder methods but differs in the sense that it introduces more complex encoders.

3.3.1 Neighborhood autoencoders

As explained in the previous methods, encoding is done by minimizing the loss function related to the similarity of pair of embeddings with their real similarity in graph structure without considering the information of the surrounding nodes. Algorithms

like Structural Deep Network Embedding (SDNE) [Wang et al., 2016a] and Deep Neural Graph Representations (DNGR) [Cao et al., 2016] uses autoencoders to compress the neighboring information of each node. To do so, a similarity matrix S_{ij} between all pairs is constructed. In this matrix, each row s_i represents the similarity of ith node with all the other nodes in the graph. As a result, each row is viewed as a vector representation of v_i's neighborhood in high-dimension. This representation is then compressed using an autoencoder to generate a low-dimensional embedding.

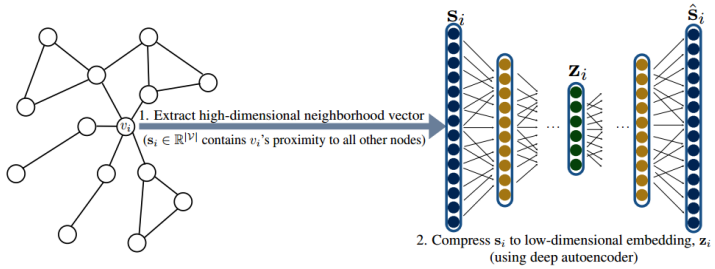


Figure 3.4: The neighborhood autoencoder first extracts a high-dimensional neighborhood vector for each node, summarizing its similarity to all other nodes in the graph. This vector is then fed through a deep autoencoder to reduce its dimensionality and produce a low-dimensional embedding.

Afterward, the loss function is minimized so that the sum of squared difference between decoding of encoded s_i and s_i itself over all v_i is minimum:

$$\ell = \sum \parallel (DEC(ENC_{autoencoder}(s_i), s_i)) \parallel_2^2 \tag{3.12}$$

3.3.2 Neighborhood aggregation and convolutional encoders

Although the neighborhood autoencoders tackle the problem of parameter sharing between embeddings, they need the entire graph for generating embeddings which can be problematic when the graph is enormous or constantly growing. It also means that if a graph is dynamic, the whole training phase should be repeated each time a new node is added to the graph.

Neighborhood aggregation methods like GraphSage [Hamilton et al., 2017a] and variations of Graph Convolutional Neural Networks (GCN) [Kipf and Welling, 2016b] [Kipf and Welling, 2016a] aims at solving these problems by aggregating the node's local neighborhood information into embeddings. As a result, each node is represented as a function of its surrounding neighborhood. First, embeddings are built upon the original input. Then, the encoding is done through iterations, where nodes aggregate the embeddings of their neighborhood using an aggregation function. Hence, even if a

node is not present in the training phase, its neighboring nodes can still be adequately represented.

In general, all the algorithms based on the idea of neighborhood aggregation follow the same procedure, although they differ in aggregation function and how they combine the previous embedding vectors with the current aggregation information.

Recently node representation learning has shown great promise in predicting interactions and discovering hidden characteristics of graphs. It can also benefit drug repurposing by representing the complex interactions between drugs and diseases. In the next chapter, we detail two case studies where a fast and effective node embedding method is applied to learn the low-dimensional representations of nodes in heterogeneous graphs, and then these representations are used to reveal unknown drug-disease interactions, which can possibly be considered as new uses of known drugs.

Chapter 4

Case study: drug repurposing through learning complex structures of graphs

We carried out our studies on drug repurposing in two cases. First, we used drug-protein and drug-disease interaction graphs to model mechanisms of action. More specifically, we built a large and heterogeneous graph comprising drug, disease, and protein entities linked according to information collected from the biomedical literature. In the graph, relations involving drugs and proteins represent mechanisms of action, while drug indications are given by relations involving drugs and diseases. Our goal is to find a low-dimensional latent representation for drugs and diseases so that the latent representation embeds the relationship between mechanisms of action and drug indications. We then, employ link prediction based on the drug-disease representations to capture repurposing candidates. Although through link prediction we are able to find possible unknown connections between drugs and diseases, we are also interested to explain why and how the trained model choose specific connections between drugs and diseases. Therefore, we analyze the properties of the generated embedding space, also the decision space of the model where we can explain the model's reasoning characteristics and further to use them to find repositioning candidates. In the second study, we applied the methods developed through the first study to extend our research and evaluate the possibility of using multiple embeddings generated from multiple independent graphs, related to four different types of relations among diseases and drugs, to train several graph models independently, join the embeddings, create a single vector representation for drugs, and use these representations to predict new uses of drugs.

4.1 Multi-relation graph unsupervised embedding for drug repurposing

Our main goal is to discover new relations between drugs and diseases by utilizing existing public drug-disease protein interactions. We built a large and heterogeneous graph comprising drug, disease, and protein entities that are linked according to information collected from the biomedical literature, as shown in Figure 4.1. More specifically, we formulate the drug repositioning problem as a three-layer multi-relation directed graph $\mathcal{G} = (V, R, E)$, where V is a set of entities (i.e., drugs, diseases and proteins), R is a set of relations (i.e., drug-protein, drug-disease and protein-protein), and E is a set of edges connecting different entities in V. In the graph, relations involving drugs and proteins represent mechanisms of action, and repositioning opportunities are represented by (hidden) relations involving drugs and diseases. The graph also contains protein-protein interactions in order to increase connectivity and information propagation while learning node representations.

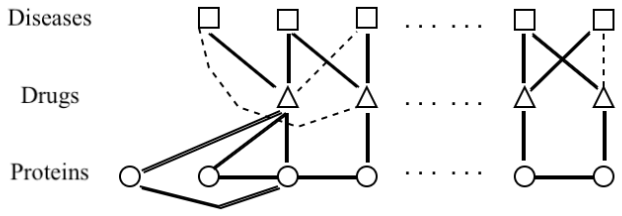


Figure 4.1: Multi-relation graph, composed of drug-protein, drug-disease and protein-protein interactions. A drug (△) perturbs some proteins (◯) and this drug is indicated to certain diseases(□). A single drug may perturb different proteins, and these proteins may also interact. Further, the same drug may be indicated to different diseases. Links may provide evidence for repositioning opportunities (i.e., dotted links).

4.1.1 Data

As in [Zitnik et al., 2018], we used the human protein-protein interaction (PPI) network compiled by [Menche et al., 2015], [Chatr-aryamontri et al., 2015], integrated with additional PPI information from [Szklarczyk et al., 2017]. The PPI graph contains physical interactions experimentally documented in humans, such as metabolic enzyme-coupled interactions and signaling interactions. The network is unweighted and undirected with 19,085 proteins and 719,402 interactions. For the graph of drug-protein, we obtained relationships between drugs and proteins from the STITCH database [Chatr-Aryamontri et al., 2015]. This database integrates various chemical and protein

networks with over 8,083,600 interactions between 8,934 proteins and 519,022 chemicals. We considered only the interactions between chemicals (i.e., drugs) and proteins that have been experimentally verified, comprising 16,546 proteins and 584 drugs, and there are 1,824,204 interactions among them. Drugbank [Wishart et al., 2008] was used to retrieve known drug-disease links. DrugBank is a bioinformatics and cheminformatics resource that provides a knowledge-base for drugs, drug actions and drug targets. We focused on 584 drugs that were indicated to 508 diseases, resulting in a total of 2,836 drug-disease links.

4.1.2 Unsupervised node representation learning and link prediction

Drug-protein and drug-disease interaction graphs usually exhibit a sparse structure, and often, a protein is linked to drugs occupying different parts of the graph. We utilize the structural roles of the proteins in order to learn representations for drugs and diseases. More specifically, we applied the general idea of SkipGram [Mikolov et al., 2013a] to learn representations in an unsupervised manner, but instead of performing expensive random walks to produce contexts [Grover and Leskovec, 2016b, Perozzi et al., 2014b], we employ a restricted number of permutations over the immediate neighborhood of a node as context to generate its representation. Further, we exploit the multi-relation nature of the graph by employing two types of contexts while learning representations. Specifically, some contexts are composed of drugs and proteins (i.e., mechanisms-of-action), while others composed of drugs and diseases (i.e., drug indications). This results in an embedding for each drug and for each disease, so that adjacent entities are placed close to each other in the vector space, while unconnected entities are pushed away. As a result, drugs and diseases that have a similar distribution of neighbors will end up being nearby.

4.1.2.1 Adaptive ego-centric graph representation learning

In natural language processing, the context of a word is often approximated by other words surrounding it. In graphs, however, a node's context is an even more complex concept. As mentioned earlier, algorithms like DeepWalk, and Node2Vec generate text-like sentences through random walks, and consequently, the context is all the nodes that appear along a walk. In contrast, in the current approach, the contexts are solely based on the neighborhoods of nodes, defined here as the nodes directly connected to them. Consequently, their neighborhoods will mainly express nodes' representations,

and those with similar locality (contexts) will be associated with similar embeddings. Therefore, embeddings are primarily focused on second-order proximity. Thus, the more common neighbors shared by a pair of vertices (v_i, v_j), the more their proximity in the embedding space.

The Nearest Neighbor Embedding (NBNE) first creates a set of groups S through a fixed number of permutations over each node's neighbors. It then trains the vector representations of nodes by maximizing the log-likelihood of predicting a node given another one in a group and given a set of representations r, making each node in a group predict all the others. The log-likelihood maximized by NBNE is given by:

$$\max_{r} \quad \frac{1}{|S|} \sum_{s \in S} \left(\log \left(p \left(s|r \right) \right) \right) \tag{4.1}$$

where $p\left(s|r\right)$ is the probability of each group, which is defined as:

$$\log \left(p \left(s|r \right) \right) = \frac{1}{|s|} \sum_{i \in s} \left(\sum_{j \in s, j \neq i} \left(\log \left(p \left(v_j | v_i, r \right) \right) \right) \right) \tag{4.2}$$

where v_i is a vertex and v_j are the other vertices in the same group. The probabilities in this model are learned using the feature vectors $r_{v_{i,j}}$, which are then used as the nodes' embeddings. The probability $p\left(v_j|v_i, r\right)$ is given by:

$$p\left(v_j|v_i, r\right) = \frac{\exp\left(r\prime_{v_j}^T \times r_{v_i}\right)}{\sum_{v \in V} \left(\exp\left(r\prime_{v}^T \times r_{v_i}\right)\right)} \tag{4.3}$$

where $r\prime_{v_j}^T$ is the transposed output feature vector of vertex j, used to make predictions. The representations $r\prime_v$ and r_v are learned simultaneously by optimizing Equation 4.1. This optimization step is performed using stochastic gradient ascent with negative sampling [Mikolov et al., 2013b].

One drawback of using a fixed number of permutations for neighborhood sampling, however, is sequence redundancy, which can result in slow training and low-quality of learned embeddings [Li et al., 2019]. We propose sampling nodes' neighborhood information based on their influence factor so as to overcome the fixed sampling drawbacks. Node influence implies the ability of a node to broadcast information. The faster and broader a node spreads, the more influential it becomes.

Burt's Constraint [Burt, 2004] is usually used to measure structural holes. Burt suggests it as a measurement for the concept of bridging social capital. Essentially, a constraint measures how much the ego's friends are also connected among themselves.

It is defined by both dyadic constraint and overall or ego-level constraint. Dyadic constraint c_{ij} is the extent to which one node i constraints another one j. It can be defined as:

$$c_{ij} = (p_{ij} + \sum_{q \in N(i)-j} p_{iq} p_{qj})^2 \tag{4.4}$$

where p_{ij} is the proportion of energy or edge weights an ego i invests in its neighbors j, and $p_{iq}p_{qj}$ is the indirect influence. The proportion of energy p_{ij} can be defined as:

$$p_{ij} = \frac{a_{ij} + a_{ji}}{\sum_{K \epsilon N(i)} (a_{ik} + a_{ki})} \tag{4.5}$$

where a_{ij} are the elements of the graph adjacency matrix. Overall constraint, c_i, is then:

$$c_i = \sum_{j \in N(i)} c_{ij} \tag{4.6}$$

Nodes with lower constraint values (less information blockage and high access) are not limited by their neighbors and can access the information flow outside of a highly cohesive nodes pool. Conversely, those with high constraints do not get much exposure to information outside of what is already accessible by their neighbors. Therefore, nodes that act as information brokers have lower constraint values.

To sample node neighborhoods efficiently, we allocate an influence score to each node based on its connectivity and constraint values so that nodes with more connections and more access to the information flow receive a higher influence score. Thus, the more influential a node in a graph's structure, the more samples are gathered from its locality. We define the adaptive permutation p' for each vertex as following:

$$p'_i = p - \frac{p}{\lceil log_2(H(c_i^{-1}, deg(i)) + 2) \rceil} \tag{4.7}$$

where p is a pre-defined permutation parameter, $deg(i)$ is the degree of ith node of the graph, and $0 \leq p'_i \leq p$. The choice of logarithmic operation is due to its flattening property in conditions when nodes with large number of edges and different constraint values may result in similar number of permutations (considering that the initial number of permutations p is often small). The harmonic average H of the

positive real numbers $x_1, x_2, \ldots, x_n$ is defined to be:

$$H = \left(\frac{\sum_{i=1}^{n} x_i^{-1}}{n} \right)^{-1} \tag{4.8}$$

and if defined in terms of the arithmetic, is the quantity which its reciprocal is the arithmetic mean of the reciprocals of the given quantities.

One major disadvantage of arithmetic mean is its sensitivity to extreme values, especially when the sample size is small. Therefore, it is not an appropriate measure of central tendency for skewed distributions. The choice of harmonic average is due to its punishing property for extreme values where there is low or no balance between the quantities. In the context of adaptive sampling, nodes with low values of degree and highest constraint have higher information blockage, which suggests the existence of many interconnections among the node's direct neighbors. This implies that those sequences which are pruned in the current node, will likely be sampled in the next iterations when the current neighbors will be the central node. This is due to the existence of several possible paths between neighbors in nodes with high constraint. Therefore, these nodes are candidates for more pruning. On the other hand, for those nodes with high direct neighbors and low constraint, more samples should be collected. In case of the nodes with high degree and maximum constraint $c_i = 1$ (the maximum possible information blockage) or low degree and minimum constraint (maximum inverse of the constraint), the harmonic mean produces a bias toward the lower values, resulting in more pruning (see Figure 4.2), although the pruning is less than the case where there are low number of direct neighbors and the information blockage is maximum.

During the sequence sampling, we adapt the permutation number regarding the influence score calculated for each node. For nodes without any neighbors or with very high constraint, there is no need to generate many permutations. The more neighbors with less information access blockage a node has, the closer the selected permutation p' to p. The Figure 4.3 illustrates an example of how the number of adapted permutations changes for different values of initially set permutations p while sampling from the drug-disease-protein graph. Note that for weighted graphs, the degree of a node can be replaced with node strength [Barrat et al., 2004] to generate biased permutations based on the sum of weights attached to ties belonging to a node (see Algorithm 1).

Burt's theory suggests that nodes hold certain positional characteristics from how they are embedded in neighborhoods. These specific characteristics change based on how the connection among the edges are distributed. Because Burt's constraint measures the direct and indirect connectivity between the neighbors of an ego, in the absence of communities or neighbors' inter-connectivity, the nodes' constraint is

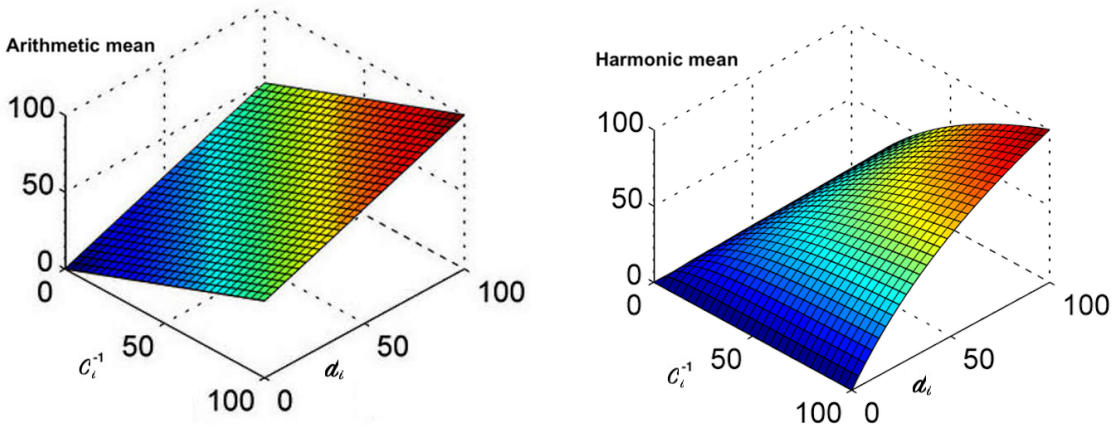


Figure 4.2: A comparison between arithmetic and harmonic means for different values of the inverse of node constraint c^{-1} and degree d for randomly selected nodes i. Note that the lowest theoretic value of the inverse constraint is 1.

uniform. One example of such graphs is bipartite networks [Asratian et al., 1998] where there is a limitation on the nodes' connection such that ties are only cross classes. This implies that no direct path between the nodes of the same type is allowed and the indirect connections are through the ego. Therefore, for such graphs where our domain knowledge shows the nonexistence of inter-connections between the ego's neighbors or lack of communities, we may simplify the equation 4.7, taking into consideration only the node's degree or strength (in case of weighted graphs):

$$p' = p - \frac{p}{\lceil log_2(deg(i) + 2) \rceil} \tag{4.9}$$

Hence, simpler sampling strategy is applicable for sparse graphs. For multipartite networks, where the graph can be partitioned into multiple sets, the number of indirect connections among the neighbors depends on how each node is positioned between the sets [Dawande et al., 2001]. Thus, for such graph structures the effectiveness of both fully adaptive and degree-based adaptive sampling methods can be measured through cross-validation [Berrar, 2019] and the best strategy may be selected for the future cycles of model training.

Many applications of proximity or similarity in graphs use the notion of first and second order proximity. The first-order proximity in a graph is the local pairwise proximity between two vertices. For each pair of nodes linked by an edge (u, v), the weight on that edge, w_{uv}, indicates the first-order proximity between u and v (the weight w_{uv} for unweighted graphs is 1). In the absence of an edge between u and

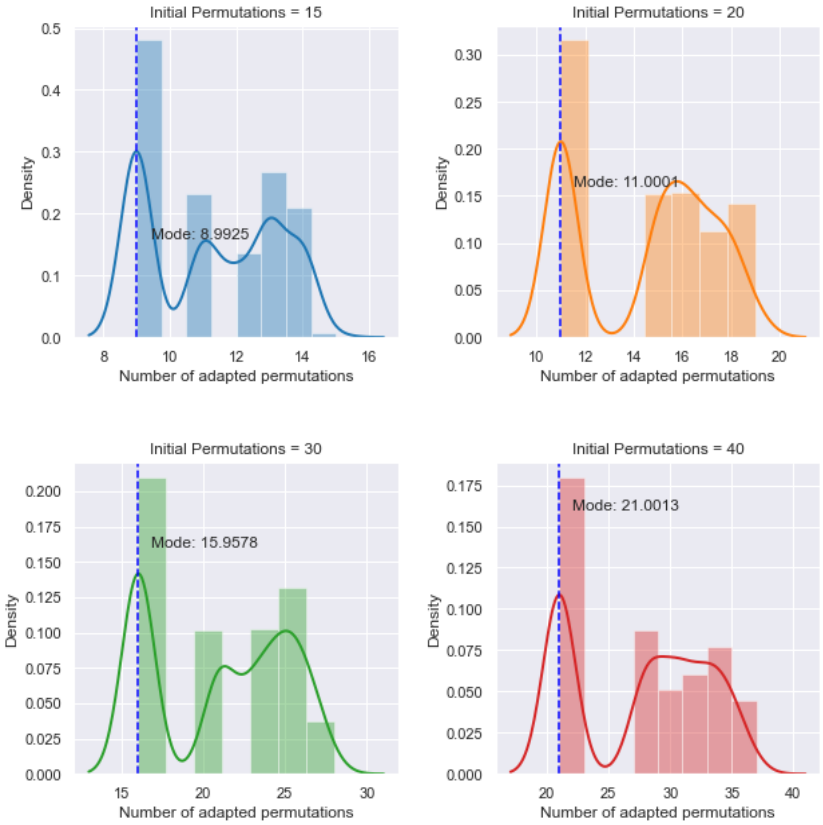


Figure 4.3: Distributions of the adapted permutations p' based on node influence factor for different initially set permutations p.

v, their first-order proximity is defined to be 0. The first-order similarity is then the concept that connected nodes in a network should have similar properties. On the other hand, the second-order proximity between a pair of nodes (u, v) is the similarity between their neighborhood network structures. From mathematics perspective, let $p_u = (w_{u,1}, ..., w_{u,|V|})$ denote the first-order proximity of u with all the other nodes, then the second-order proximity between u and v is defined by the similarity between p_u and p_v. If no vertex is linked from/to both u and v, the second-order proximity between u and v is 0. The notion of second-order proximity indicates that nodes with similar neighborhoods should have common characteristics.

The proposed approach takes advantage of both first and second-order proximities as the representation of set of nodes v_j that are in the context of the ego v_i are learnt so that the sum of the conditional probabilities of the context given the target node is maximized (see the equation 4.2). By optimizing the log probability, the algorithm maximizes the likelihood of predicting a neighbor given a node, creating node embeddings so that nodes with similar neighbors have similar representations. Since there is more than one neighbor in each group, this model also makes connected nodes

Algorithm 1: Adaptive sequence generation

GenerateSequence (G, p, l)
- **inputs :** the graph G, upper bound permutation number p and sequence length l
- **output:** An array of generated sequences
- $sequences \leftarrow [\emptyset]$;
- **foreach** *node* $v \in G$ **do**
  - $ds \leftarrow v.degree()$ or $v.strength()$;
  - $constraint \leftarrow v.constraint()$;
  - $influence\ score = Harmonic\ mean(inverse\ of\ constraint, ds)$
  - $p' = p - p/log_2(influence\ score + 2)$
  - $group \leftarrow [\emptyset]$;
  - **for** $i = 0;\ i <= p';\ i = i + 1$ **do**
    - $group = [v] + permute(v.neighbors, l)$
    - $sequences.append(group)$
- **return** *sequences*;

have similar representations because they will both predict each other's neighbors, resulting in representations also with first-order similarities. A trade-off between first and second-order proximity can be achieved by changing the initial number of permutations p which controls the number of distinct nodes within each generated sequence and the length of the sequence l. The higher the number of permutations in the current node, the more likely the generated sequences through neighbors permutations have shared nodes (the second-order similarity), and the higher the length of the sequence l, the more the model learns about the ego's direct neighbors (the first-order similarity).

Here, we considered Node2Vec and DeepWalk as strong baselines, because they are widely used and have been proved to be highly effective [Grover and Leskovec, 2016a]. The main difference between Adaptive NBNE and the two baselines is the sampling strategy. DeepWalk uses uniform random walks to explore the around neighborhood. The major limitation of this approach is the lack of any control over the explored regions. Node2vec, on the other hand, takes into consideration the information of both previous and the potential next nodes to generate biased walks and in each step calculates a group of weights for all the possible paths of the next step. The choice of sampling method has major effect in both the embedding quality and the time complexity. DeepWalk has a time complexity bounded by $O(|V| \times n)$ where $|V|$ is the number of nodes and n represents the number of walks. Node2vec takes two additional parameters, p and q, which are then used to increase the probability of returning to the parent node or going further away. Although this

results in having a more efficient sampling and as a results a better embedding quality [Grover and Leskovec, 2016a], several models should be trained in order to choose the best values for p and q. Another disadvantage is its time and memory complexity dependency on the graph's branching factor b and having a time complexity bounded by $O(b^2)$, making it hardly scalable to large and dense networks [Liu and Krishnan, 2021]. The Adaptive NBNE based on calculating the structural holes has a worst case time complexity bounded by $O(|V| \times d^2)$ where n is the number of vertices for which the Burt's constraint is calculated and d is the average degree in the graph. Although for the worst case scenario our approach has a higher time complexity than the DeepWalk due to the calculation of node constraints, the reduction in the total number of sequences and using the first and second-order proximities results in embeddings with higher quality and a faster training as the sampling occurs efficiently and only in the direct neighborhood. The adaptive sampling based on node degree has a low worst case time complexity bounded by $O(|V| \times average(p\prime))$. Because of the near-bipartite property of sparse graphs [Cranston and Yancey, 2020] and considering that the real-world graphs are often sparse [Danisch et al., 2018] [Chung, 2010], this approach is potentially best suited for real-world problems as it can significantly reduce the excessive sequences generated in the previous approaches while increasing the embeddings' quality.

4.1.2.2 Model selection

We first evaluate the performance of the fixed-size permutation (interchangeably we will use NBNE as a short form of fixed-size permutation NBNE) by comparing the quality of its embeddings against those generated by Node2vec and DeepWalk through a link prediction task. Hence, we conducted a random search with 25 iterations. Cross-validation is used to find the best embedding space of node representations. Finally, the following configurations were selected for each of the algorithms as the best hyperparameters: a) NBNE with a window size of 24, number of permutations equal to 30; b) Deepwalk with a window size of 12, number and length of walks equal to 7 and 25; c) Node2vec with window size, number and length of walks equal to 5, 57 and 73, respectively.

4.1.2.3 Link prediction

Our official drug indications contain 2836 links. We divided the dataset into five folds, each time one of them is used for the validation and the rest for the training. However, as the indications contain only positive examples, negative examples are essential to

training a classifier. This is due to the fact that the drug-disease graph has many nodes without any connection to the others. Therefore, we generated 301964 negative examples using the complementary graph of the known indications. For embedding quality evaluation, we used the mean area under the curve (AUC score) of Random Forest [Breiman, 2001] trained on the generated embeddings as a primary measure for assessing the performance of the node representation learning algorithms.

Our results show that the NBNE achieves numbers as high as 85% in terms of mean area under the curve, while Node2vec and Deepwalk achieve 68% and 67% of mean area under the curve, respectively (see Figure 4.4). With 55% and 72%, the fixed-size nearest node embedding also shows better results in both kolmogorov-Smirnov (KS) score [Massey Jr, 1951] and true positive rate (recall), when compared with 46% and 45% KS and 63% and 62% recall for Deepwalk and Node2vec respectively (see the complete results in the Table 4.5). As explained, the main difference of the NBNE from the other two algorithms is the context generation approach, as NBNE is based on gathering information of the immediate neighborhood.

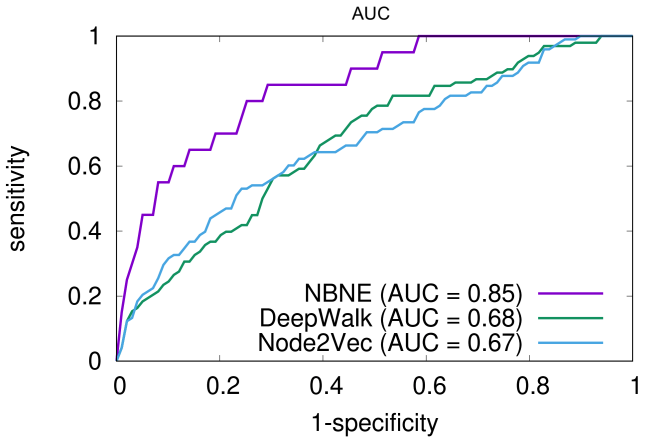


Figure 4.4: The area under the curve related to the link prediction task using the embeddings generated by Deepwalk, Node2vec and NBNE.

4.1.2.4 Adaptive strategy compared to the fixed-size permutations

As mentioned before, sequence redundancy may lead to slow training and lower embedding quality. The adaptive permutation can enable efficient sampling and fast training as it considers information accessibility of nodes in order to pay more attention to those of strong influence. A total of 3 types of embeddings, with permutation parameters set to 15, 20, 30, and 40 for each of the adaptive variants, and fixed permutation NBNE were trained, and the results were analyzed. More specifically, we compared fixed-size

and adaptive permutation neighborhood sampling in three criteria: 1) a total number of sentences used in training, 2) training time, and 3) the area under the curve when the generated embeddings were used in the link prediction task.

As seen in Figure 4.5, as the permutation number increases, the number of sequences generated by NBNE increases rapidly, and fixed permutation strategy results in generating 17280, 23040, 34560, and 46080 samples when the permutation parameter was set to 15, 20, 30 and 40. On the other hand, the Full Adaptive NBNE, where both node constraint and its degree are used to sample from the neighborhood, yields 6429, 8179, 11481, and 14651 sequences for the same initial number of permutations. Hence, by applying a fully adaptive permutation sampling strategy, we manage to reduce the number of sequences by approximately 63%, 65%, 67%, and 75% for 15, 20, 30, and 40 number of initial permutations, respectively (see Table 4.6). Furthermore, because the number of samples is reduced significantly, the training becomes much faster than fixed-size sampling as the number of initially set permutations increases. For degree-based adaptive sampling, the number of generated sequences are 12847, 16287, 23635, 31142 for 15, 20, 30, and 40 initial permutations, respectively. This higher number of sequences compared to the fully adaptive strategy is due to the high pruning degree of the additional constraint factor in equation 4.7 which leads to penalizing the nodes with higher information blockage.

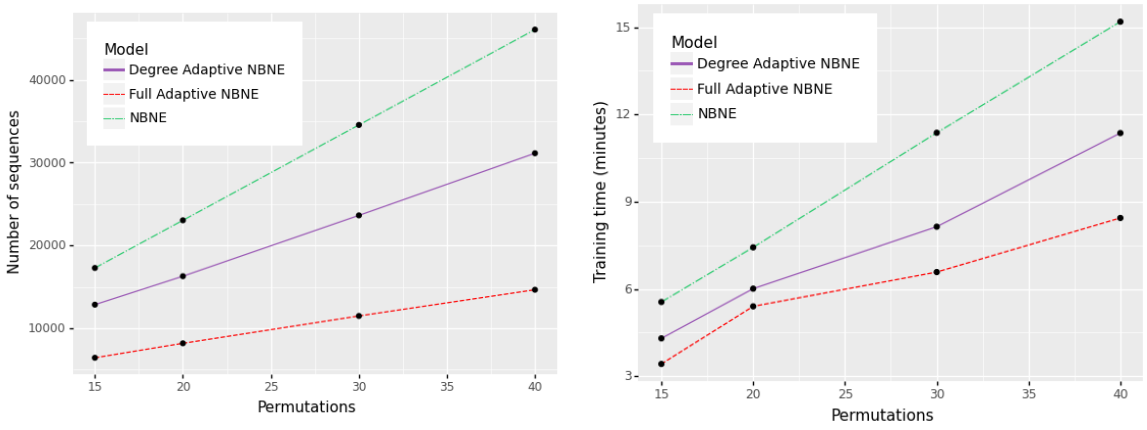


Figure 4.5: the training time and the total number of sentences generated during the training phase in fully adaptive and degree based adaptive NBNE compared to the fixed-size permutation strategy. As shown, both the Full the Degree Adaptive approaches results in less training sequences and faster training. Between the three methods, the full adaptive sampling strategy resulted in the least number of sequences and consequently the least training time.

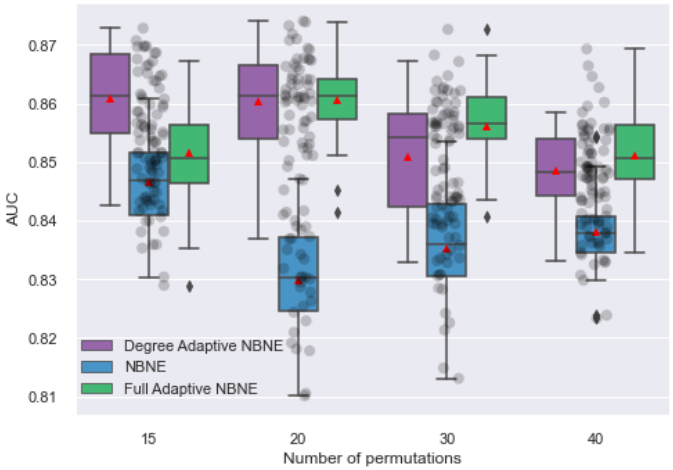


Figure 4.6: The box plot of the area under the curve (AUC) for the Full Adaptive, Degree Adaptive, and fixed permutation sampling. For each model and each initially set permutation number, 30 experiments in from of 30-folds cross-validation were carried out and the AUC metric was calculated during each experiment.

Fold	AUC - NBNE (p = 15)	AUC - NBNE with full adaptive sampling (p = 15)	AUC - NBNE with degree adaptive sampling (p = 15)
1	0.83 ± 0.0141	0.843 ± 0.01368	0.858 ± 0.01311
2	0.858 ± 0.01313	0.856 ± 0.01318	0.864 ± 0.01289
3	0.848 ± 0.01348	0.856 ± 0.01319	0.867 ± 0.0127
4	0.84 ± 0.01378	0.851 ± 0.01339	0.855 ± 0.01324
5	0.852 ± 0.01335	0.85 ± 0.01343	0.865 ± 0.01285
6	0.854 ± 0.01325	0.852 ± 0.01336	0.873 ± 0.01252
7	0.859 ± 0.01309	0.849 ± 0.01346	0.86 ± 0.01305
8	0.855 ± 0.01323	0.846 ± 0.01355	0.87 ± 0.01265
9	0.846 ± 0.01355	0.844 ± 0.01363	0.854 ± 0.01328
10	0.842 ± 0.01370	0.86 ± 0.01304	0.852 ± 0.01336
11	0.836 ± 0.01192	0.829 ± 0.01415	0.843 ± 0.01366
12	0.849 ± 0.01347	0.845 ± 0.0136	0.861 ± 0.01299
13	0.857 ± 0.01317	0.855 ± 0.01323	0.865 ± 0.01285
14	0.851 ± 0.01239	0.846 ± 0.01355	0.868 ± 0.0127
15	0.85 ± 0.01340	0.866 ± 0.01278	0.87 ± 0.01264
16	0.847 ± 0.01152	0.85 ± 0.01341	0.87 ± 0.01265
17	0.843 ± 0.01466	0.854 ± 0.0132	0.853 ± 0.0133

18	0.838 ± 0.01384	0.835 ± 0.01394	0.859 ± 0.01309
19	0.845 ± 0.0136	0.844 ± 0.01365	0.87 ± 0.01265
20	0.843 ± 0.01165	0.856 ± 0.01319	0.862 ± 0.0129
21	0.84 ± 0.01278	0.864 ± 0.01289	0.861 ± 0.01301
22	0.841 ± 0.01375	0.849 ± 0.01345	0.869 ± 0.0126
23	0.836 ± 0.01092	0.85 ± 0.01340	0.85 ± 0.01343
24	0.85 ± 0.01341	0.844 ± 0.01364	0.859 ± 0.01306
25	0.842 ± 0.01372	0.848 ± 0.01348	0.846 ± 0.01355
26	0.852 ± 0.01336	0.856 ± 0.01318	0.865 ± 0.01283
27	0.852 ± 0.01336	0.865 ± 0.01282	0.856 ± 0.01321
28	0.861 ± 0.013	0.863 ± 0.0129	0.869 ± 0.01269
29	0.842 ± 0.01172	0.867 ± 0.01275	0.843 ± 0.01368
30	0.838 ± 0.01384	0.854 ± 0.0132	0.871 ± 0.01261

Table 4.1: The area under the curve with 95% confidence interval for 30-fold test when the initial permutation parameter was set to 15.

Fold	AUC - NBNE (p = 20)	AUC - NBNE with full adaptive sampling (p = 20)	AUC - NBNE with degree adaptive sampling (p = 20)
1	0.824 ± 0.0143	0.866 ± 0.0128	0.866 ± 0.0128
2	0.821 ± 0.0144	0.852 ± 0.0133	0.86 ± 0.0131
3	0.823 ± 0.0134	0.841 ± 0.0137	0.853 ± 0.0133
4	0.83 ± 0.0141	0.854 ± 0.0133	0.864 ± 0.0129
5	0.81 ± 0.0127	0.874 ± 0.0125	0.862 ± 0.013
6	0.847 ± 0.0135	0.872 ± 0.0126	0.867 ± 0.0128
7	0.825 ± 0.0143	0.864 ± 0.0129	0.856 ± 0.0132
8	0.826 ± 0.0102	0.862 ± 0.013	0.851 ± 0.0134
9	0.842 ± 0.0137	0.861 ± 0.013	0.861 ± 0.013
10	0.83 ± 0.0141	0.857 ± 0.0132	0.867 ± 0.0127
11	0.834 ± 0.014	0.868 ± 0.0127	0.871 ± 0.0126
12	0.81 ± 0.0147	0.861 ± 0.013	0.849 ± 0.0134
13	0.837 ± 0.0139	0.86 ± 0.013	0.868 ± 0.0127
14	0.835 ± 0.0139	0.868 ± 0.0127	0.865 ± 0.0128

15	0.833 ± 0.014	0.863 ± 0.0129	0.85 ± 0.0134
16	0.83 ± 0.0141	0.861 ± 0.013	0.837 ± 0.0139
17	0.829 ± 0.0142	0.851 ± 0.0134	0.86 ± 0.013
18	0.832 ± 0.0141	0.853 ± 0.0133	0.863 ± 0.0129
19	0.841 ± 0.0137	0.869 ± 0.0127	0.873 ± 0.0125
20	0.83 ± 0.0141	0.856 ± 0.0132	0.857 ± 0.0136
21	0.828 ± 0.0142	0.862 ± 0.0129	0.856 ± 0.0132
22	0.818 ± 0.0145	0.869 ± 0.0127	0.874 ± 0.0125
23	0.818 ± 0.0145	0.858 ± 0.0131	0.868 ± 0.0127
24	0.838 ± 0.0138	0.861 ± 0.013	0.858 ± 0.0131
25	0.819 ± 0.0145	0.862 ± 0.013	0.847 ± 0.0135
26	0.839 ± 0.0138	0.863 ± 0.0129	0.866 ± 0.0128
27	0.839 ± 0.0138	0.845 ± 0.0136	0.871 ± 0.0126
28	0.84 ± 0.0138	0.864 ± 0.0129	0.854 ± 0.0133
29	0.832 ± 0.012	0.861 ± 0.013	0.852 ± 0.0133
30	0.837 ± 0.011	0.861 ± 0.013	0.864 ± 0.0129

Table 4.2: The area under the curve with 95% confidence interval for 30-fold test when the initial permutation parameter was set to 20.

Fold	AUC - NBNE (p = 30)	AUC - NBNE with full adaptive sampling (p = 30)	AUC - NBNE with degree adaptive sampling (p = 30)
1	0.831 ± 0.0141	0.862 ± 0.013	0.841 ± 0.0138
2	0.837 ± 0.0139	0.856 ± 0.0132	0.857 ± 0.0132
3	0.831 ± 0.0141	0.856 ± 0.0132	0.839 ± 0.0138
4	0.835 ± 0.014	0.86 ± 0.013	0.843 ± 0.0137
5	0.844 ± 0.0136	0.841±0.0138	0.849±0.0135
6	0.844 ± 0.0136	0.855±0.0132	0.853±0.0133
7	0.839 ± 0.0138	0.857±0.0132	0.86±0.013
8	0.844 ± 0.0136	0.849±0.0135	0.841±0.0138
9	0.828 ± 0.0142	0.86 ± 0.013	0.845 ± 0.0136
10	0.842 ± 0.0137	0.855 ± 0.0133	0.858 ± 0.0131
11	0.824 ± 0.0143	0.861 ± 0.012	0.862 ± 0.013
12	0.834 ± 0.014	0.854 ± 0.0133	0.858 ± 0.0131

13	0.843 ± 0.0137	0.844 ± 0.0136	0.846 ± 0.0136
14	0.813 ± 0.0147	0.861 ± 0.013	0.865 ± 0.0129
15	0.834 ± 0.014	0.857 ± 0.0132	0.839 ± 0.0138
16	0.823 ± 0.0144	0.857 ± 0.0132	0.857 ± 0.0132
17	0.839 ± 0.0138	0.844 ± 0.0136	0.867 ± 0.0128
18	0.821 ± 0.0144	0.859 ± 0.0131	0.858 ± 0.0131
19	0.839 ± 0.0138	0.856 ± 0.0132	0.858 ± 0.0131
20	0.815 ± 0.0146	0.85 ± 0.0134	0.833 ± 0.014
21	0.833 ± 0.014	0.846 ± 0.0136	0.862 ± 0.013
22	0.834 ± 0.014	0.868 ± 0.0127	0.856 ± 0.0132
23	0.854 ± 0.0133	0.862 ± 0.0129	0.843 ± 0.0137
24	0.844 ± 0.0136	0.844 ± 0.0137	0.846 ± 0.0135
25	0.849 ± 0.0135	0.864 ± 0.0129	0.856 ± 0.0132
26	0.828 ± 0.0142	0.86 ± 0.013	0.859 ± 0.0131
27	0.831 ± 0.0141	0.873 ± 0.0125	0.856 ± 0.0132
28	0.84 ± 0.0138	0.854 ± 0.0133	0.84 ± 0.0138
29	0.837 ± 0.0139	0.857 ± 0.0132	0.836 ± 0.0139
30	0.845 ± 0.0136	0.863 ± 0.0129	0.842 ± 0.0137

Table 4.3: The area under the curve with 95% confidence interval for 30-fold test when the initial permutation parameter was set to 30.

Fold	AUC - NBNE (p = 40)	AUC - NBNE with full adaptive sampling (p = 40)	AUC - NBNE with degree adaptive sampling (p = 40)
1	0.834 ± 0.014	0.849 ± 0.0135	0.843 ± 0.0137
2	0.83 ± 0.0141	0.845 ± 0.0136	0.856 ± 0.0132
3	0.833 ± 0.014	0.835 ± 0.014	0.855 ± 0.0132
4	0.836 ± 0.0139	0.856 ± 0.0132	0.849 ± 0.0135
5	0.823 ± 0.0143	0.849 ± 0.0135	0.856 ± 0.0132
6	0.838 ± 0.0138	0.837 ± 0.0139	0.845 ± 0.0136
7	0.824 ± 0.0143	0.848 ± 0.0135	0.859 ± 0.0131
8	0.854 ± 0.0133	0.853 ± 0.0133	0.843 ± 0.0137
9	0.836 ± 0.0139	0.84 ± 0.0138	0.849 ± 0.0135
10	0.841 ± 0.0138	0.863 ± 0.0129	0.85 ± 0.0134

11	0.84 ± 0.0138	0.853 ± 0.0133	0.844 ± 0.0136
12	0.84 ± 0.0138	0.866 ± 0.0128	0.846 ± 0.0136
13	0.844 ± 0.0136	0.837 ± 0.0139	0.843 ± 0.0137
14	0.84 ± 0.0138	0.86 ± 0.013	0.847 ± 0.0135
15	0.836 ± 0.0139	0.852 ± 0.0133	0.855 ± 0.0132
16	0.839 ± 0.0138	0.852 ± 0.0134	0.839 ± 0.0138
17	0.836 ± 0.0139	0.857 ± 0.0132	0.848 ± 0.0135
18	0.837 ± 0.0139	0.869 ± 0.0127	0.844 ± 0.0136
19	0.836 ± 0.0139	0.852 ± 0.0134	0.848 ± 0.0135
20	0.854 ± 0.0133	0.865 ± 0.0129	0.852 ± 0.0133
21	0.841 ± 0.0137	0.85 ± 0.0134	0.833 ± 0.014
22	0.849 ± 0.0134	0.845 ± 0.0136	0.84 ± 0.0138
23	0.839 ± 0.0138	0.847 ± 0.0135	0.855 ± 0.0132
24	0.833 ± 0.014	0.847 ± 0.0135	0.849 ± 0.0135
25	0.831 ± 0.0141	0.856 ± 0.0132	0.848 ± 0.0135
26	0.833 ± 0.014	0.845 ± 0.0136	0.851 ± 0.0134
27	0.842 ± 0.0137	0.847 ± 0.0135	0.856 ± 0.0132
28	0.838 ± 0.0139	0.86 ± 0.0131	0.847 ± 0.0135
29	0.839 ± 0.0138	0.849 ± 0.0135	0.855 ± 0.0132
30	0.849 ± 0.0134	0.854 ± 0.0133	0.848 ± 0.0135

Table 4.4: The area under the curve with 95% confidence interval for 30-fold test when the initial permutation parameter was set to 40.

Because the adaptive sampling strategy generates sequences based on the information access rate of nodes, it also manages to improve the model performance as shown in Figure 4.6. The results suggest that adaptive sampling delivers better classification results for both degree-based and full adaptation while reducing the number of sequences and the training time, in average being faster with approximately 38% for full and 31% for degree based adaptation strategies. This reduction is observable in all the other configurations as illustrated in Table 4.6. The difference between the number of sequences in degree based and fully adaptive sampling due to the existence of some protein-protein connections which results in a higher information blockage for the drugs connected to these proteins. Note that when model's hyperparameters are fixed, the training time is only affected by the number of samples gathered through the calculation of constraint and node degrees, as discussed earlier. Since in all the

Model	Mean AUC	Mean KS	Mean Recall
NBNE (p = 15)	0.839	0.559	0.723
NBNE with full adaptive sampling (p = 15)	0.844	0.579	0.758
NBNE with degree adaptive sampling (p = 15)	0.869	0.591	0.7672
NBNE (p = 20)	0.825	0.543	0.741
NBNE with full adaptive sampling (p = 20)	0.855	0.562	0.732
NBNE with degree adaptive sampling (p = 20)	0.86	0.585	0.758
NBNE (p = 30)	0.846	0.544	0.715
NBNE with full adaptive sampling (p = 30)	0.857	0.594	0.767
NBNE with degree adaptive sampling (p = 30)	0.855	0.572	0.741
NBNE (p = 40)	0.83	0.521	0.706
NBNE with full adaptive sampling (p = 40)	0.857	0.592	0.784
NBNE with degree adaptive sampling (p = 40)	0.854	0.562	0.75
Deepwalk	0.681	0.462	0.632
Node2vec	0.672	0.457	0.619

Table 4.5: Alternative metrics used to measure the performance of Deepwalk, Node2vec, fixed-size, full adaptive and degree adaptive nearest neighbor embedding methods for 5-fold trials.

Model	Number of generated sequences	Training time (minutes)
NBNE (p = 15)	17280	5.55
NBNE with full adaptive sampling (p = 15)	6429	3.42
NBNE with degree adaptive sampling (p = 15)	12847	4.30
NBNE (p = 20)	23040	7.43
NBNE with full adaptive sampling (p = 20)	8179	5.40
NBNE with degree adaptive sampling (p = 20)	16287	6.01
NBNE (p = 30)	34560	11.37
NBNE with full adaptive sampling (p = 30)	11481	6.58
NBNE with degree adaptive sampling (p = 30)	23635	8.14
NBNE (p = 40)	46080	15.19
NBNE with full adaptive sampling (p = 40)	14651	8.44
NBNE with degree adaptive sampling (p = 40)	31142	11.36

Table 4.6: Sequence numbers and training time obtained in different configurations of fixed-size, full adaptive and degree adaptive sampling.

previous methods the sequence generation has been considered as a sub-process of the embedding training, we also consider the node influence factor calculation phase as a part of training. Finally, the best results for k-fold cross validation evaluation are gained when the number of initial permutations is set to 20, where the full adaptive strategy reaches a mean value of 86.6% and a maximum value of 87.4% of the area under the curve in 30 folds among a total of 360 trained models during the test phase (see Tables 4.1, 4.2, 4.3 and 4.4).

After generating high quality embeddings of the nodes in drug-disease-protein graph through an unsupervised, fast and efficient algorithm, we are interested in understanding the characteristics of this latent space and possibly use them to extract

insights that may potentially accelerate the drug repurposing process. In the next section, we present the process of generating low-dimensional visualizations of this latent space, also its traits which lead to a better understanding of the proximities of diseases and their associated drugs.

4.1.3 Drug-disease embedding space properties

We employ the t-distributed stochastic neighbor embedding (t-SNE) [Van der Maaten and Hinton, 2008] in order to visualize the embedding space of drugs and diseases. The t-SNE is a technique used to visualize high dimensional data by giving each data point a location in a two-dimensional map. The visualization suggests some interesting insights about the reasons that lead to the good performance of our method. In order to have a clear visualization, the drugs are represented by triangles (blue) and the diseases by rectangles (green). Further, some points are also highlighted in the figures to demonstrate interesting properties of drugs, diseases and their interactions in the embedding space. Figure 4.7 shows that similar diseases have close vector representations. Figure 4.7 (Left) shows a cluster of diseases (red points) representing pain-related diseases, while Figure 4.7 (Right) shows a cluster of muscle-related diseases. These visualizations suggest that our method generates meaningful representations as related diseases are located close to each other in the embedding space.

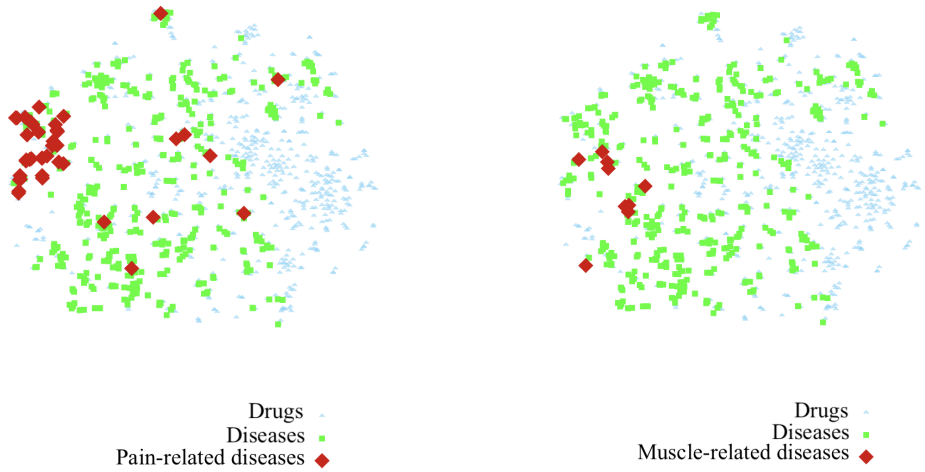


Figure 4.7: Proximity of related diseases. Left — pain-related diseases. Right — muscle-related diseases.

We have also analyzed the spatial relation of diseases and their corresponding drugs. Figure 4.8 (Left) highlights drugs which are used to treat breathing difficulty. In this case, most of the indicated drugs are concentrated next to the disease. The

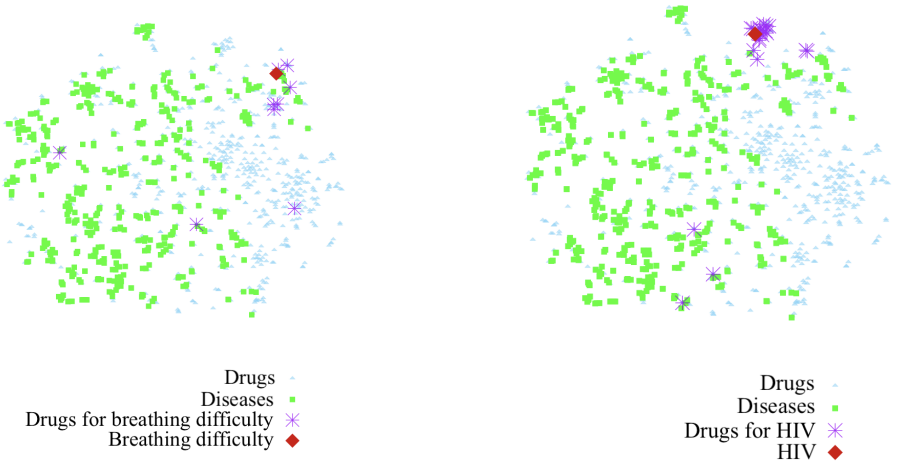


Figure 4.8: Proximity of diseases and their corresponding medications. Left — breathing difficulty. Right — HIV. In general, drug indications and the corresponding health problems are located closely

same trend is observed in Figure 4.8 (Right), where we highlighted the HIV disease. Again, most of the indicated drugs are placed next to the disease. These visualization give us a great insight of the embedding space generated by our algorithm and how its properties can help in searching new uses for known drugs.

Since proximity in the embeddings space implies the existence or lack of the existence of connections among drugs and a specific group of diseases, these results suggest that nearby regions can be appropriate choices for finding drug repurposing opportunities. As mentioned earlier, the process of drug discovery is a long and expensive process. Although the traditional methods have successfully reduced this prolonged process, the hypothesis generation phase still requires a lot of time, and conducting these efforts in the right direction can help be more efficient in this phase. Therefore, after generating meaningful representations for drugs and diseases, they can be used to limit the search space of repurposing opportunities, reducing the time and cost of finding new uses of known drugs.

For drug repurposing, both predictability (in form of a variety of supervised and unsupervised tasks like classification and clustering) and explainability are important in identifying the possible candidates. Although the generated embedding space has interesting properties which help limit the search space in hypothesis generation phase, link prediction task with the use of local approximator algorithms like random forest and boosted decision trees [Woodruff, 2017] assists predicting the possible links in a non-linear manner and through a local approximation. Furthermore, it lets automating the process of identifying new uses of drugs. Since the embedding space does not represent these non-linear and local decisions boundaries which exist in the decision

space, it may not be ideal for clustering tasks. On the other hand, generating such a decision space for complex models (i.e. Random Forest) can be expensive due to high time complexity of capturing such non-linear patterns. Therefore, it is desirable to learn the mapping from the original space to the decision space using a small proportion of data, transferring the entire observations to the decision space using the learnt mapping function and then carrying out all the prediction and explanation efforts in a single space. To do so, it is necessary to identify the decision space properties and verify the possibility of carrying out prediction tasks in a such space. In the next sections, we analyze its properties and evaluate the possibility of drug repositioning in this space.

4.1.4 Predictability in decision space

Predictability is the degree to which a correct prediction or forecast of a machine learning system is generated, either qualitatively or quantitatively. On the other hand, explainability can facilitate the understanding of how a model makes its decisions, having insights about different aspects of data and variables that make the prediction possible. Although intrinsic interpretation methods are by nature more interpretable (i.e., linear models), they may not be able to learn highly complex data efficiently [Krakovska et al., 2019]. Nonetheless, often non-parametric models can learn intricate patterns well, but they also can be difficult to explain [Adadi and Berrada, 2018].

Lots of studies have proposed post-hoc explanations that quantifies variable impact on a model [Moradi and Samwald, 2021], [Rudin, 2019]. For example, the Random Forest was developed with built-in permutation importance that evaluates reductions in model performance after removing each variable. Therefore, one commonly used post-hoc approach is to ask tree-based models' opinion about variable importance in a dataset and then train another one using the selected most important variables by the previous model [Kazemitabar et al., 2017]. Although these approaches have been applied widely, they still leave the model and its decisions a black-box. Recently, the game theory-based approaches have been trying to unveil the underlying behavior of the black-box algorithms and get insights about data by understanding the model's decisions [Ribeiro et al., 2016], [Lundberg and Lee, 2017a]. Although these approaches have been used successfully in explaining non-parametric black-box models, prediction and explanation still take place separately and in two distinct spaces (Shapley and feature spaces). Moreover, since black-box and non-parametric models are often slower to train and may suffer over-fitting [Ganguly and Cambier, 2021], simpler models are preferable when observations are linearly separable. Thereby, unifying predictability and explainability in a more linearly separable space is desirable.

In coalational game theory, a game consists of a set of players, a value function which maps the set of players $P \subseteq \{1, 2, ..., N\}$ to a real value $V(P)$, game is what reproduces the outcomes, and the mapping should satisfies $V(\emptyset) = 0$. The value function shows how much collective payoff a group of players can bring to the game through their collaboration. Shapley value answers the question of how much contribution a player i brings to the game through calculating the contribution of a set of players with and without the presence of player i. Therefore the Marginal Contribution δ of player i with respect to a coalition P can be defined as:

$$\delta_v(i, P) = V(P \cup i) - V(P) \tag{4.10}$$

Hence, the Shapley value of a player i, Φ_i, can be seen as a weighted average of its marginal contributions:

$$\Phi(i) = \sum_{S \subseteq P - \{i\}} \frac{V(S \cup i)) - V(S))}{\binom{n-1}{|S|}} \tag{4.11}$$

where S is a subset of players and n is the number of players in a collation, and satisfies the following axioms:

Symmetry: for players i, and j, if $\delta_V(i, P) = \delta_V(j, P)$ for any subset of players, then $\Phi(i) = \Phi(j)$. In other words, if the marginal contribution of two players for any permutation is equal, then their Shapley values are equal as well.

Additivity: for games a and b, and a single player i, $\Phi_{a+b}(i) = \Phi_a(i) + \Phi_b(i)$. This guarantees that value function of a combined game is equal to the sum of individual value functions of each game.

Dummy: for a single player i, if $\delta_V(i, P) = 0$, then the Shapley value $\Phi(i) = 0$. That is, if a player does not bring any contribution to the game, its Shapley value is zero.

Given a model $f(X = p_1, p_2, p_3, ..., p_d)$, the set of features 1 to d can be considered as players. Hence, model and its predictions are the game and the payoff V, respectively.

Lemma. If a model $f_1(X)$ has a better learning capability than a second model $f_2(X)$ and both models have better learning capability than a random classifier (i.e. a coin tosser), then the Shapley space of $f_1(X)$ is more linearly separable than the Shapley space of $f_2(X)$.

Proof. Shapley space is more linearly separable than feature space if a model $f(X)$ has a better learning capability than a random classifier (i.e. a coin tosser). Suppose a binary classification problem with balanced classes, and multi-variable ob-

servations X_i with feature set $\{p_1, p_2, p_3, ..., p_d\}$, belonging either to class 1 or 0. As before, for this model the Shapley value of feature p for an observation X_j, $\Phi_{p_i}(X_j)$, is defined as the weighted sum of marginal contributions. Considering the Dummy axiom, for a random classifier or a not learnable observation:

$$\forall p_1, p_2, ..., p_d \in P \quad , \quad \Phi(p_i) = 0 \tag{4.12}$$

That is, for a completely random classifier, the Shapley values for all the features are zero. Therefore, in a classifier with better predictability than a random predictor, for at least one of the features and one of the observation $\Phi_{p_i}(X_j) \neq 0$. That is:

$$E_{f(X)}[|\phi(p_i)|] \geq E_{f_{random}(X)}[|\phi(p_i)|] \tag{4.13}$$

Based on the definition of marginal contribution, for a non random classifier and a correctly classified instance X_j, if it belongs to class 1, then $\Phi_{p_i}(X_j) > 0$, and $\Phi_{p_i}(X_j) < 0$ otherwise. Therefore from (4.13):

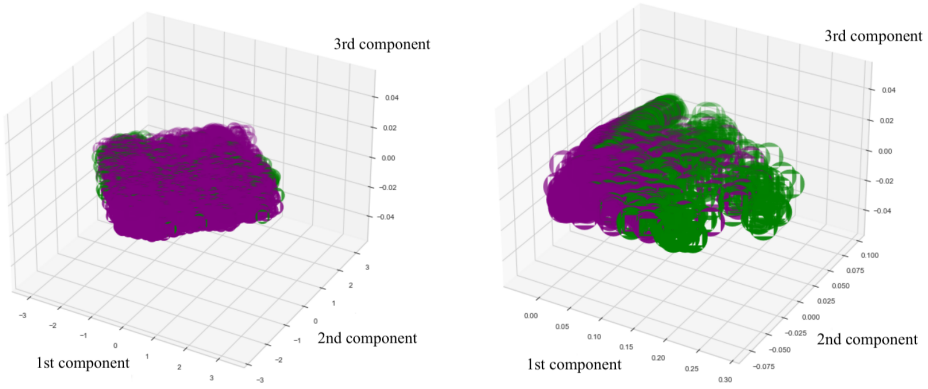
$$\begin{gathered} E_{f(X)}[\phi(p_i)|class = 1] \geq E_{f_{random}(X)}[\phi(p_i)|class = 1] \\ and \\ E_{f(X)}[\phi(p_i)|class = 0] \leq E_{f_{random}(X)}[\phi(p_i)|class = 0] \end{gathered} \tag{4.14}$$

Given two models $f_1(X)$ and $f_2(X)$, suppose $f_1(X)$ has better learability than $f_2(X)$. That is, $f_1(X)$ is able to classify more observations correctly and with a higher Kolmogorov-Smirnov distance(KS) between the cumulative distributions of predicted probabilities for each class. Because for a model with higher KS, less classification errors (more correctly classified observations) means larger difference between the marginal contributions of instances related to each class [Utkin et al., 2021], then the equivalent of 4.13 and 4.14 are satisfied for general case.

Namely, from 4.14, the mean values of the distributions of the observations associated with each class move away from each other. The more a model capable of learning, the more distance between the mean values of the instances related to each class in Shapley space and the better separability.

Transforming feature space into another with more separability is common in a group of statistical learning approaches. Support Vector Machines (SVM) is capable of handling nonlinear classification problems by mapping input vectors into a high-dimensional space with kernel functions where data are more linearly separable [Cortes and Vapnik, 1995], [Noble, 2006]. Training on Multilayer Perceptron (MLP) can be thought of as a process of learning a mapping from original feature space to

a more linearly separable one through matrix multiplication [Yegnanarayana, 2009]. Because of the membership separability characteristic of Shapley space, the weights of a Shapley regression represent a mapping function that transforms the original observations into a new favorable space with a higher separability than the original feature space. Therefore, we first trained a Shapley regression, based on Random Forests, only with a proportion of data used in the training phase of the previous link prediction task. Then the trained weights were used to transform the test data into the Shapley space. The Principal Component Analysis was applied to make it possible to visualize both spaces in lower dimensions. As shown in Figure 4.9a and 4.9b, since the trained Random Forest model has better learnability than a random classifier, its Shapley space is more separable than the original feature space.



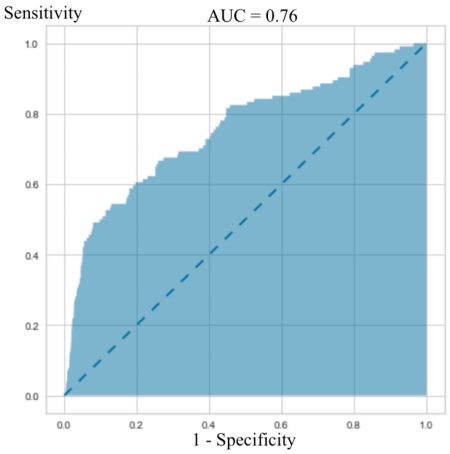
(a) Original feature space (b) Shapley space

Figure 4.9: A comparison between the Shapley and the embedding space of the drug-disease-protein graph in terms of separability of observations related to each class in three dimensions. Each dimension represents one of the principal components found by PCA.

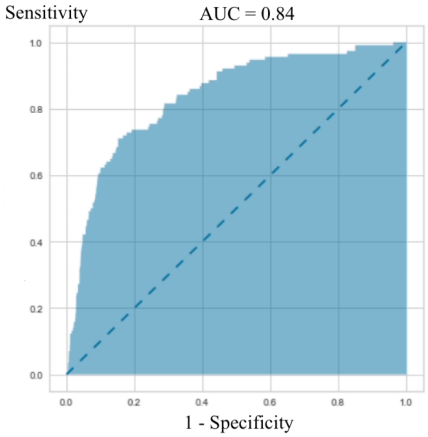
Asking the opinion of a primary model about the nature of data and using it in a second model is a common practice in machine learning [Mursalin et al., 2017], [Pan and Shen, 2009]. A grey-Box is a combination of white-Box and black-Box models, and its goal is to is develop an ensemble of black and white-Box models to combine and acquire the benefits of both [Pintelas et al., 2020]. Because of the non-linear nature of black-box models, they typically outperform the white-box models in complex learning tasks [James et al., 2013]. Therefore, due to the feature separability of Shapley space and because the interpretation of data is already embedded in Shapley values, one can make use of black-box models to learn the Shapley transformation based on a small proportion of data, transform the entire data to the Shapley space, and then train a white-box model (which has better interpretability and is generally faster to

train) in this new space so that it achieves better classification or regression results compared to the original space. Furthermore, migrating to Shapley space is favorable because both supervised and unsupervised tasks can be done in a more interpretable space where predictability and interpretability are met.

As illustrated in Figures 4.10a and 4.10b, Logistic Regression achieves 76% of AUC when trained on the original embedding space of the drug-disease-protein graph. At the same time, it manages to reach 84% of AUC after being trained on Shapley space of a Random Forest as the primary model.



(a) Logistic Regression trained on the original space.



(b) Logistic Regression trained on the Shapley space.

Figure 4.10: The performance of Logistic Regression as a white-box model in original and Shapley space.

The proposed grey-box modeling in combination with the Shapley separability property helps obtain the same flexibility of a complex and hard to explain black-box algorithm, with better explainability. The Random Forest, initially used as a black-box model, has a time complexity of $O(k \times n \times log(n) \times m)$ where k is the number of trees, and m and n are the number of independent variables and number of observations, respectively. The space complexity for this algorithm is $O(k \times 2^d)$ where d is the maximum depth. Since the time complexity of the splitting phase of tree-based methods for continuous variables depends on the number of observations (with a $O(n \times log(n))$ factor), using p proportion of data results in a faster training. The time and space complexity of a Logistic Regression is $O(n \times m)$ and $O(m)$, respectively, having a faster training as well [Singh et al., 2009]. The proportion of data to be used in Shapley values generation phase can be considered as a hyperparameter and can be decided through cross-validation evaluation. As shown in Figure 4.11, applying only 20% of data is sufficient to capture the complex decision making process of the

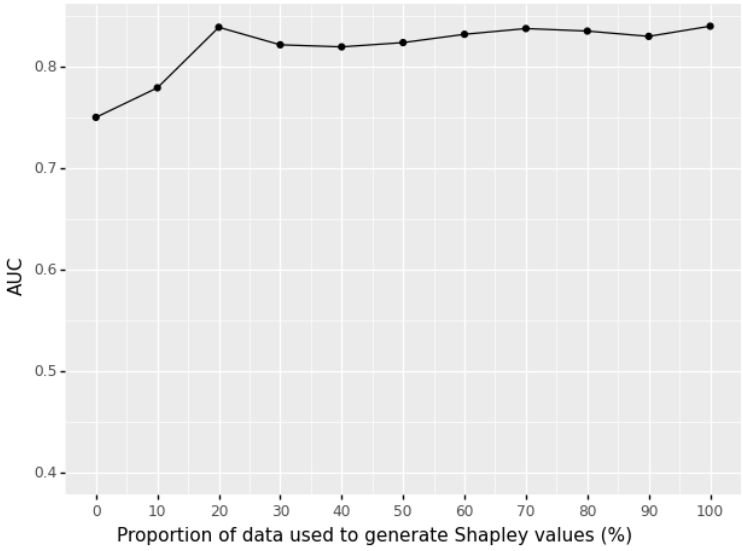


Figure 4.11: The proportion of data used to generate Shapley values and train the Logistic Regression versus the model performance in terms of the area under the curve.

Random Forest by the Logistic Regression as a less complex white-box algorithm, resulting in a better explanation of the decision making process and faster training. It is worth mentioning that the time complexity of generating Shapley values using SHAP approximation is of the order $O(k \times n \times m \times d^2)$ and thus, using a p proportion of data leads to faster generation of Shapley values as well.

4.1.5 Clustering in Shapley versus embedding space

Shapley values enable understanding the model's global decision making process through combining many local explanations. From the structural perspective, the embedding space generated by the proposed algorithm maintains the first and second-order similarities, making the nearest neighbors located in the same region. However, proximity in the embedding space does not necessarily convey a link between the the neighbors. The proximity based on the connectivity of the nearest neighbors is decided by model and within its decision space, making the Shapley space more convenient for further interpretations.

Previously, it was shown that the Shapley space is more linearly separable than the original space since for a model with better separability than a random model, the expected value of the weighted average of marginal contributions of each player for observations of different classes move away from each other. In addition to the concentration of these expected values of the marginal contributions of different classes in different regions, those observations with similar feature contributions should occupy

nearby regions.

Lemma. For a given class and a set of observations $x_1, x_2, x_3, ..., x_n$ and a set of variables $p_1, p_2, p_3, ..., p_d$, observations with similar weighted average of marginal contributions $\Phi(1), \Phi(2), \Phi(3), ..., \Phi(d)$, are more closely located than other data points with less similar weighted marginal contributions.

Proof. Suppose two observations x_i and x_j with similar $\Phi(d)$s, and another observation x_k with different weighted average of marginal contributions compared to x_i and x_j. That is, at least for one of the players p_d in the game, x_i and x_j have more similar marginal contributions compared to x_k. The Euclidean distance between two data points a and b is defined as:

$$d(a,b) = ||b-a||_2 = \sqrt{\sum_{i=1}^{d}(b_i - a_i)^2} \tag{4.15}$$

Therefore the Euclidean distance between x_i and x_j is,

$$d(x_i, x_j) = \sqrt{(\Phi(1)_j - \Phi(1)_i)^2 + (\Phi(2)_j - \Phi(2)_i)^2 + ... + (\Phi(d)_j - \Phi(d)_i)^2} \tag{4.16}$$

The Euclidean distance between each of the x_i or x_j with x_k is then:

$$d(x_{i|j}, x_k) = \sqrt{(\Phi(1)_k - \Phi(1)_{i|j})^2 + (\Phi(2)_k - \Phi(2)_{i|j})^2 + ... + (\Phi(d)_k - \Phi(d)_{i|j})^2} \tag{4.17}$$

where $i|j$ represents an attribute related to ith or jth observations. For example, $d(x_{i|j}, x_k)$ shows the distance between x_k and either x_i or x_j, and $\Phi(d)_{i|j}$ represents the weighted marginal contribution of either ith or jth observations for dth player.

Because two observations with more proximity were defined to have at least one more similar weighted marginal contribution than a third data point, thereby:

$$\begin{aligned} & |\Phi(1)_j - \Phi(1)_i| < |\Phi(1)_k - \Phi(1)_{i|j}| \\ or \quad & |\Phi(2)_j - \Phi(2)_i| < |\Phi(2)_k - \Phi(2)_{i|j}| \\ or \quad & , ..., |\Phi(d)_j - \Phi(d)_i| < |\Phi(d)_k - \Phi(d)_{i|j}| \end{aligned} \tag{4.18}$$

Therefore from 4.16, 4.17 and 4.18:

$$d(x_i, x_j) < d(x_{i|j}, x_k) \tag{4.19}$$

As a result, x_i and x_j are closer compared to x_k. Note that although a measure of Euclidean distance was used to prove the proximity between x_i and x_j, the same

result is valid for any $p - norm$ of Minkowski distance [Gueorguieva et al., 2017]:

$$d(x_i, x_j) = ||x_j - x_i||_p = \left(\sqrt{\sum_{d=1}^{D} (\Phi(d)_j - \Phi(d)_i)^p} \right)^{\frac{1}{p}} \tag{4.20}$$

where p represents the pth norm.

As shown, the sum of the marginal contributions of non-random model is supposed to be similar for observations associated with each class, making not only the expected value of coalitions of each feature more linearly separable than the original space, also makes the observations with similar feature marginal contribution be located in the nearby regions. These attributes do not hold in the embedding space. Hence, when the goal is to cluster the closely related observations based on their proximity in having similar links, Shapley space is a more appropriate choice for clustering when the similarity metric is based on Minkowski distance.

The practical benefit that the generated Shapley space brings to our clustering task can be shown using clustering quality metrics. Suppose that the data points i of the embedding space have been grouped by K clusters C_k for $K = 1$ to K. The average distance between the data point i and all the other observations in cluster C_k is:

$$a_i = \frac{1}{|C_k| - 1} \sum_{i,j \epsilon C_k, i \neq j} d(i, j)) \tag{4.21}$$

where $d(i, j)$ is some measure of Minkowski distance between observations i and j. Therefore, the smallest mean distance of a point i to all the other data points in the other clusters $C_{m,m \neq k}$, called as smallest mean dissimilarity, is defined as:

$$b_i = min \frac{1}{|C_m|} \sum_{j \epsilon C_{m,i \neq j\prime}} d(i, j\prime)) \tag{4.22}$$

where the cluster with the smallest mean dissimilarity is called *neighboring cluster*. The Silhouette [Rousseeuw, 1987] of a single data point is then defined as:

$$s_i = \frac{b_i - a_i}{max(a_i, b_i)} \quad , \quad if \quad |C_k| > 1 \tag{4.23}$$

and

$$s_i = 0 \quad , \quad if \quad |C_k| = 1 \tag{4.24}$$

The Silhouette coefficient is defined to be the the maximum value of the mean

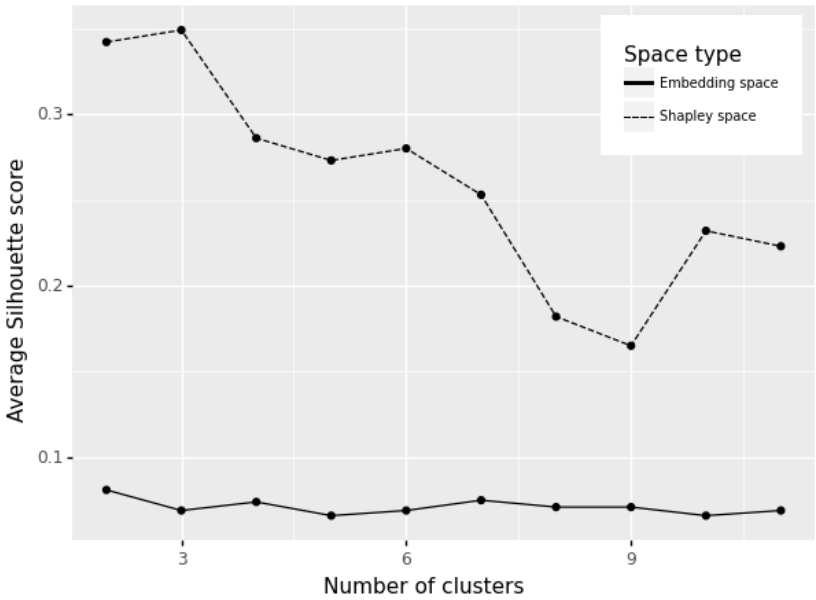
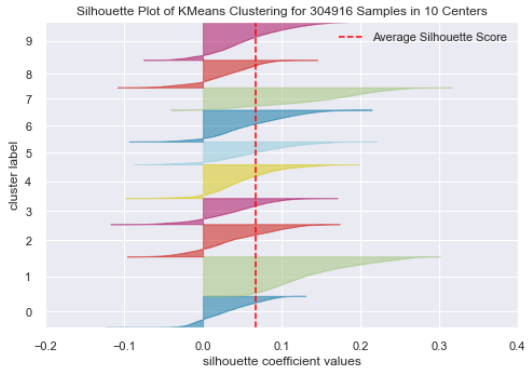


Figure 4.12: A comparison of Silhouette coefficients for different number of clusters ranging from $K = 2$ to 11. The clustering in Shapley space constantly shows a better separation of clusters for all the the values of K.

$s(i)$ over all the observations of the entire dataset and ranges from -1 to 1. A score of 1 indicates a perfect clustering and the closer the Silhouette coefficient value to 1, the more well apart from each other the clusters are. On the other hand, a Silhouette coefficient of 0 and -1 means indifferent and wrong clustering, respectively.

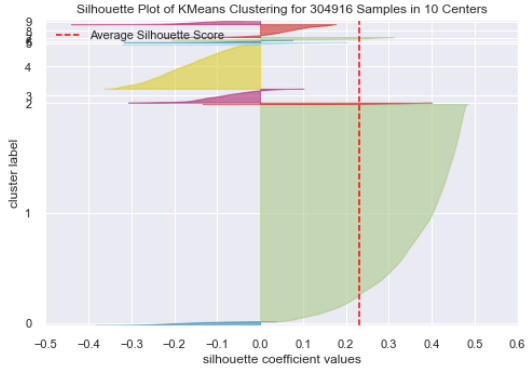
In order to compare the cluster qualities, the clustering was carried out in both Shapley and embedding spaces and then, the Silhouette coefficients were calculated for different values of K in K-means clustering ranging from 2 to 11 (as the best number of clusters which was identified through elbow method). As illustrated in Figure 4.12 and 4.13, for all the values of K in K-means, the Shapley space constantly enables the clustering algorithm to form groups with better separation. More specifically, clustering in the embedding space resulted in an average Silhouette coefficient of 0.0689 across all the K configurations, while the Shapley space leads to a better average Silhouette score of 0.257.

It should be noted that when Gradient Boosting-based algorithms such as XGboost [Chen et al., 2015] are used to train and generate Shapley values, if the loss is set to be a binary logistic objective function, the Shapley values are log-odds. In case the calculated values are probabilities generated using an inverse of log-odds, the generated

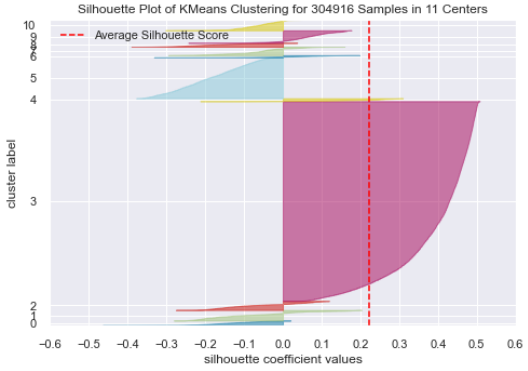


(a) Silhouette score for K-means, $K = 10$, trained on the original space

(b) Silhouette score for K-means, $K = 11$, trained on the original space



(c) Silhouette score for K-means, $K = 10$, trained on the Shapley space



(d) Silhouette score for K-means, $K = 11$, trained on the Shapley space

Figure 4.13: Examples of groups formed through K-means clustering for $K = 10$ and $K = 11$ in the embedding and Shapley spaces.

space is a Simplicial space [Baues, 1995] and can be defined as:

$$S^D = \left\{ X = [x_1, x_2, x_3, ..., x_D] \, \epsilon R^D | X_i > 0, \sum_{i=1}^{D} x_i = k \right\} \tag{4.25}$$

Because the probabilities generated by log-odds alongside all the dimensions sum to 1, the probability space can be considered as a Simplicial space where $k = 1$. Such a space generated by compositional data [Aitchison, 1982] can be modeled with Dirichlet distribution [Makgai et al., 2021]. Because the observations in Dirichlet distribution are located in the probability space, the Euclidean distance usually used in clustering algorithms like K-means does not hold anymore [Bui et al., 2017]. Thereby, another distance measures like Bhattacharyya [Bhattacharyya, 1943] distance should be adopted [Bui et al., 2017]. Also for any types of analysis based on distance (i.e. mean distance between data points), Isometric Logratio Transformations (ILR) can be applied to transform the compositional coordinates to real ones while preserving all

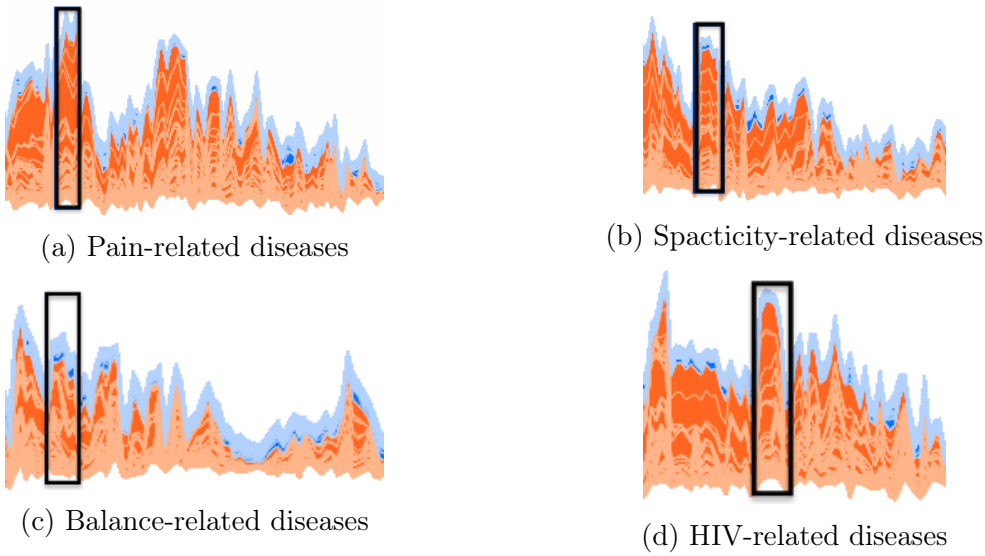
metric properties [Egozcue et al., 2003].

4.1.6 Drug repurposing opportunities in decision space

We used SHAP [Lundberg and Lee, 2017b] based on a Random Forest model initially trained on a proportion of data (only the training set) and then transformed the entire embedding data (associated with the train and test sets) related to the drug-disease-protein graph into the Shapley space. Afterward, we analyzed the response factor of variables associated with each disease and drug. The response factor of a variable is its tendency to push the model's prediction toward a specific result. Therefore in our link prediction problem, a positive response factor is a variable's attempt to convince the model to predict the existence of a link between a drug and disease, and negative response factors do the opposite. We found out that those diseases and medications that share the same group of positive and negative response factors have similar characteristics. As illustrated in Figure 4.14, the response factors of each cluster show a relation with a specific group of diseases and drugs. Based on the explanation of the model, illnesses related to the same cluster are supposed to be related as well, and the medications of each cluster are possibly good candidates for drug repurposing [Vogt et al., 2014]. This property has been shown in several clusters. For example, one of the clusters shows that diseases related to muscle and joint pain, bipolar II disorder, and cluster headaches respond equally to specific embedding features. Interestingly, a recent research confirms that Migraine with active headache is associated with other painful physical symptoms among patients with major depressive disorder [Hung et al., 2019]. In another example, our algorithm put Multiple Sclerosis (MS) and Toxoplasmosis in the same cluster. According to a study carried out by [Enriquez-Marulanda et al., 2017], Toxoplasmosis should be considered as a differential diagnosis of tumefactive MS. Our algorithm also decided to group Amyotrophic Lateral Sclerosis (ALS) and HIV, which confirms the result of the studies where it was reported that that ALS-like disorder should be considered an HIV-related neurologic complication [Moulignier et al., 2001] [Douville and Nath, 2017]. Recognition of these HIV-related motor neuron syndromes is crucial because they may positively respond to the same group of treatments. These syndromes also raise theoretical issues whether sporadic ALS could ever be caused by a virus or autoimmunity. It is still not known how HIV might cause a motor neuron disorder [Jubelt and Berger, 2001]. Another interesting result is the association between panic disorder and Amyotrophic lateral sclerosis (ALS) found by our algorithm and recently confirmed by [Siciliano et al., 2019]. According to this study, 33% of patients who suffer from ALS also suffer from some kind

of panic and anxiety disorder.

As mentioned before, our model suggests using the medications of each cluster as possible candidates for treating the diseases which are members of the same group. Accordingly, our algorithm suggests Gabapentin as a candidate for bipolar II disorder. This result has been confirmed by [Fullerton et al., 2010] and [Pande et al., 2000], and in several other studies. While Naproxen is used in treating balance problems, it can also be used for treating Myofascial Pain ([Khalighi et al., 2016]). Our algorithm confirms this result as it puts these diseases and medications in the same group. Recent studies show that Fibromyalgia is associated with muscle tension and depression [Bosco et al., 2019]. As another recent research carried out by [Guymer and Littlejohn, 2019a] shows, Amitriptyline, which has been used in the treatment of muscle tension, is a possible candidate for Fibromyalgia. By putting Amitriptyline and Fibromyalgia in the same group, our algorithm chooses this drug as a possible repurposing medication to treat Fibromyalgia. Finally, [Bubnova et al., 2019] and [Donato and Brown, 2019] confirmed that both Amlodipine and Atorvastatin caused significant improvement in patients with high blood pressure, which is in accordance with our results.



(a) Pain-related diseases

(b) Spacticity-related diseases

(c) Balance-related diseases

(d) HIV-related diseases

Figure 4.14: Four examples of groups of diseases with the same response factors. Those diseases with the same set of Shapley features which move the model toward a specific class show having similar symptoms.

Having the same group of negative and positive response factors is an important Shapley space property that reveals how drugs with a similar mechanism of action and diseases with similar symptoms share similar latent representations, However, the similarity of the Shapley values related to response factors reveals which regions of Shapley

space have more chance of finding repurposing candidates. In order to understand the effect of having similar Shapley values, we used K-means [MacQueen et al., 1967] with $K = 11$ as the maximum point of decrease in intra variance. As mentioned before, the number of clusters, K, was decided by the elbow method. Since the sum of the intra-group variances slowly continues decreasing up to $K = 25$, higher number of clusters based on the maximum number of meaningful groups can also be examined.

Diseases with no similar connecting edge with the same group of diseases in the original graph were separated and their possible relation was analyzed. In this Shapley space, disease proximity is translated to the existence or lack of existence of similar connections with a group of drugs and proteins (either directly or indirectly). Because diseases with the same underlying mechanism may have similar symptoms [Pietzner et al., 2021], we expect that diseases located in the same groups to have similar phenotypes and thus having repositionable drugs [Vogt et al., 2014]. By analyzing the disease pairs in each group, we found several repurposing opportunities. Namely, our algorithm indicates a relation between Optic Neuritis and Renal Calculi (which itself can cause Renal failure). Wolfgang and colleagues mentioned that Renal calculi can be an underlying cause of Optic Neuropathy [Winkelmayer et al., 2001]. Lee et al. reported that Bilateral Optic Neuropathy can be a rare uraemic manifestation of end-stage renal disease [Lee and Vaithilingam, 2011]. Also, a recent study mentions that eye shares genetic and structural pathways with the kidney and suggests that ocular and kidney diseases may be closely linked [Escoli et al., 2018]. Interestingly, a recent study [Hayes et al., 2020] reported the effectiveness of Corticotropin, a drug used in the treatment of optic neuritis, to improve renal damage. Our algorithm put Glioblastoma and Eczema into the same clusters, indicating a strong relationship between these two diseases. This is in accordance with the research carried out by Wang and colleagues, where it is reported that Eczema is associated with a reduced risk of Glioma [Wang et al., 2016b]. Azathioprine is mainly used in the treatment of people with severe atopic Eczema. Recently, Nam et al. reported that Azathioprine is a promising therapeutic to treat Glioblastoma [Nam et al., 2021]. Brown and colleagues confirmed that prevalence of incontinence is significantly elevated among women with pre-diabetes, when compared with women with normal glucose levels [Brown et al., 2006]. The near relation between Essential Tremor(ET) and Parkinson's has been studied in [Tarakad and Jankovic, 2018], and authors cited that some patients diagnosed with ET show an increased risk of developing Parkinson's years after initial tremor attacks. Gabapentin is a drug used in the treatment of ET. Abe et al. confirmed that Gabapentin had improved both hallucination and pain in patients diagnosed with Parkinson's [Abe et al., 2016] (see Tables 4.7 and 4.8).

These results suggest that both embedding and decision spaces (related to the prediction of drug-disease interactions) have several useful characteristics that permit selecting a specific group of candidates with a high possibility of repositioning among thousands of drugs, accelerating the drug repurposing process and providing a geometric explanation.

Disease	Associated with	Also confirmed by
ALS-like disorders	HIV-related neurologic	[Douville and Nath, 2017]
Panic disorder	ALS	[Siciliano et al., 2019]
Toxoplasmosis	MS	[Guymer and Littlejohn, 2019b]
Optic Neuritis	Renal Calculi	[Winkelmayer et al., 2001]
Glioblastoma	Eczema	[Wang et al., 2016b]
Essential Tremor	Parkinson's	[Tarakad and Jankovic, 2018]

Table 4.7: Diseases with similar shapley characteristics are associated with each other.

Drug	Disease	Also confirmed by
Naproxen	Myofascial Pain	[Khalighi et al., 2016]
Amitriptyline	Fibromyalgia	[Guymer and Littlejohn, 2019b]
Amlodipine	High blood pressure	[Donato and Brown, 2019]
Atorvastatin	High blood pressure	[Bubnova et al., 2019]
Corticotropin	Optic Neuritis	[Hayes et al., 2020]
Azathioprine	Glioblastoma	[Nam et al., 2021]
Pergabalin	Generalized unprovoked vulvodynia	[van Beekhuizen et al., 2018]
Tizanidine	Chronic Pain	[Peck et al., 2020]
Celecoxib	Musculoskeletal Arthritis	[Krasselt and Baerwald, 2019]
Amitriptyline	Generalized unprovoked vulvodynia	[van Beekhuizen et al., 2018]
Cyclobenzaprine	Chronic Pain	[Peck et al., 2020]
Brexpiprazole	Schizophrenia	[Ekinci and Ekinci, 2018]
Cetirizine	Asthma	[Corsico et al., 2019]
Gabapentin	Bipolar II disorder	[Fullerton et al., 2010]

Table 4.8: Some of the candidates for drug repurposing found by the proposed method which also have been confirmed in the recent studies.

During this study, we proposed an unsupervised node representation learning approach based on adaptive sampling of nodes' nearest neighbors. Using this method we managed to decrease the number of generated sequences up to 70% and we reached an approximately 40% faster training for $p = 20$ as the configuration which resulted in the best link prediction result. We showed that in the generated embedding space, diseases with similar symptoms and their corresponding drugs are located in nearby regions. This important discovery of the relation of drugs and diseases in the embedding

space directly benefits the time consuming processes of finding repositionable drugs by limiting the search space. Here in addition to indicating possible candidates, we aim at explaining why a specific or a group of drugs are good repurposing candidates. Therefore, we analyzed the explanation space and found out that diseases with similar positive and negative response factors are associated with each other and usually share a same group of symptoms. Because in the explanation space of drug-disease-protein graph having similar response factors is an indication of having connections with the same group of diseases and proteins, those drugs with similar response factors should possibly share connections with the same group of diseases. By utilizing the specific characteristics of the explanation space, we integrated all the prediction, explanation and clustering tasks in the same space where the explanation comes with each single observation, the predictions become more explainable and the groups generated through clustering task are more separable. Finally, by applying the proposed approach we managed to find several repositioning opportunities which some have also been confirmed in the recent studies as already shown in Table 4.8 (see the complete list of the associated drugs and diseases in the appendix).

4.2 Extended case study for modeling pharmacological effects with multi-graph embedding

Repositioning opportunities can be uncovered systematically by following either a disease-centric approach, as a result of a close relation between an old and new indication, a target-centric one, which links a known target and its established drug to a new indication, or a drug-centric approach, which connects a known drug to a new target and its associated indication. Disease-centric repositioning may appear faster and more direct than target-centric and drug-centric repositioning. In fact, a disease-centric repositioning hypothesis is based on a close connection between drug-indications. This appear to be the ideal approach as it enables the possibility of drug repositioning without a need to have deep knowledge about physico-chemial interactions between drugs, targets and proteins. However, if this were the case, one cancer drug would cure all forms of cancer. Disease-centric approaches require a detailed understanding of the disease phenotype and underlying molecular processes to pursue the novel indication. In target-centric repositioning, we consider only drugs for which old and new indication are of different type. However, a novel link from target to new in-

dication is a rare finding. Since each of these approaches has its own pros and cons, we propose a novel hybrid approach which models the latent relation among drug-target, drug-mechanism, drug-indication and disease-indication connections in order to predict novel drug connections with diseases.

4.2.1 Data

We collected the data from Drugbank, which is a web-enabled database containing molecular information about drugs, their interactions, their mechanisms, and their targets [Wishart et al., 2018]. Our cohort is composed of 315 drugs, 1218 diseases and four types of interactions: a) drug-action; b) drug-mechanism; c) drug-target and d) drug-indication each with 102033, 3593, 36454 and 10714 interactions respectively. As illustrated in Figure 4.15, based on these interactions, we constructed four bipartite graphs.

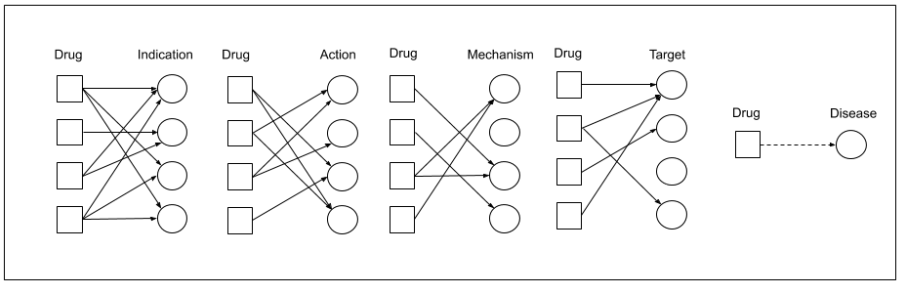


Figure 4.15: Four types of interactions including drug-indication, drug-target, drug-action and drug-mechanism were used to create four independent graphs. These interactions are learnt and used to predict the possible drug and disease links.

Here one of our goal is to investigate the possibility of modeling drug-disease interactions through modeling drug-indication, drug-action, drug-target and drug-mechanism interactions without having prior knowledge about drug-disease links. By doing so, we would construct a deductive reasoning through the following hypothesis: if two or more drugs have similar interactions (in drug-indication, drug-target, drug-action and drug-mechanism graphs), they are likely repositionable.

4.2.2 Modeling strategy and results

We created four bipartite graphs and then trained them separately using degree adaptive NBNE (with window size = 24 and number of permutations = 20) as explained in the previous section. We tested the performance of each graph representation individually and chose drug-action, drug-target and drug-mechanism representations as the

best candidates for creating the final drug representation. It is important to mention that it is in fact possible to form a single graph of drug, action, mechanism and target attributes and find the representations in a single space, but we rather train graphs separately for three reasons: a) it is much easier to interpret the meaning of embeddings for separate graphs than a single graph with complex interactions; b) since graph learning algorithms approximate complete graph exploration through random walks or neighborhood permutations, creating multiple representations for huge graphs can reduce the learning error and generate better embeddings [Epasto and Perozzi, 2019]; c) different embeddings have different meanings and can be utilized in different problems.

After extracting node representations (with 128 dimensions), a single representation with 384 features for each drug is created. In the next step, for each disease we create a dataset containing the representations (with 384 features) of 315 drugs and a binary class which is set to 1 if the drug is used in the treatment of the disease and 0 otherwise.

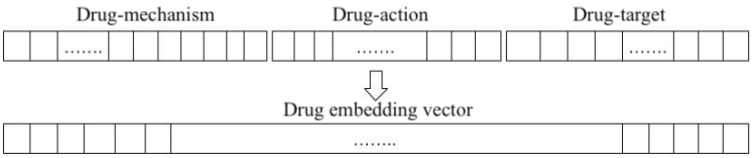


Figure 4.16: After concatenating the embeddings of four graphs, the final drug representation is a vector with 384 features.

Twenty two diseases were selected to examine the effectiveness of our method in finding the possible interactions between drug and diseases. The criteria of choosing diseases is the number of drugs are currently used to treatment of the disease. This is due to the fact that diseases with extremely low number of drugs are hard to model.

In order to avoid the curse of dimensionality problem and choosing the most informative set of variables, the best set of features were chosen using feature ranking with recursive feature elimination, Support Vector Machine (SVM) with linear kernel as the estimator and a 5-fold cross validation. After choosing the best set of features, a 10-fold cross-validation is used to assess the embedding's quality. We divide the dataset related to each disease into 10 folds, each time one of them is used for the validation and the rest for the training. Support Vector Machine, K-nearest Neighbors and XGBoost were used to model drug-disease interactions. Using XGBoost, as the algorithm with the best classification results, our method manages to identify the drug-disease interaction and correctly guess when a drug is or is not used in treatment of disease. As before, we used AUC-ROC score as the performance metric. As shown in

Table 4.9, the best results are obtained for Cardiovascular diseases with 86%, Dementia with 83% and Anxiety with 85% AUC-ROC score.

Disease	AUC
Infection	76%
Neoplasm	69%
Depressive disorder	73%
Melanoma	65%
Cardiovascular diseases	86%
Schizophrenia	79%
Lymphoma	72%
Anxiety	85%
Dementia	83%
Clorectal Neoplasms	70%
HIV	58%
Anxiety	85%
Chronic pain	68%
Carcinoma Non-small Cell Long	66%
Luekemia Myeloid Acute	68%
Ovarian Neoplasms	57%
Anxiety	85%
Multiple Myeloma	63%
Pancreatic Neoplasms	57%
Carcinoma renal cell	58%
Heart failure	60%
Breast Neoplasms disorder	58%

Table 4.9: The result of link prediction of drug-disease interactions for 22 diseases using the combined drug-action, drug-mechanism and drug-target embeddings.

We analyzed the confusion matrix and observed some interesting results. Because in our method each instance is a combination of drug-action, drug-target and drug-mechanism interactions, drugs with similar representations have similar interactions. As a result, false positives (those drugs with similar indication, action and mechanism) may reveal repositioning opportunities or drug side effects (as they are additional, usually undesirable, effects of medications which can arise through the same mechanism of action as the therapeutic effect).

Carvedilol is used for treating high blood pressure and heart failure and was confused by our model as a treatment of dementia. As mentioned in a study by Wang et al. [Wang et al., 2011], evidence from their studies reveals that carvedilol interferes with β aggregation mechanisms and demonstrates the efficacy of carvedilol treatment to mitigate AD-type amyloid neuropathology and although this drug has not yet been

Drug	Disease	Also confirmed by
Brexpiprazole	Dementia	[Khalighi et al., 2016]
Amitriptyline	Fibromyalgia	[Grossberg et al., 2019]
Chlorpromazine	Anxeity	[Meng et al., 2018]
Brexpiprazole	Anxeity	[Thase et al., 2019]
Testosterone Enanthate	Depression	[Bhasin and Seidman, 2019]
Zileuton	Depressive disorder	[Nam et al., 2021]
Asenapine	Depression	[Vieta and Montes, 2018]
Clofazimine	Lymphoma	[Lentz et al., 2019]
Ribavirin	Lymphoma	[Dominguez-Gomez et al., 2019]
Carvedilol	Amyloid Neuropathology	[Wang et al., 2011]

Table 4.10: A group of the candidates for drug repurposing found by analysing the false positives related to a link prediction task where the concatenated embeddings of drug-mechanism, drug-action and drug-target are used to model drug-disease interactions. False positives are those instances which show high similarity with the in-use medications. These results have also been confirmed in the recent studies.

approved by FDA, their study supports the continued development of carvedilol for the treatment and prevention of Alzaimer Disease (AD) dementia. Our algorithm also finds a strong relation between Brexpiprazole and AD dementia. Using this drug was supported by a recent study which emphasises that it has the potential to be an efficacious, safe, and well-tolerated treatment for dementia [Grossberg et al., 2019].

Chlorpromazine which is primarily used to treat psychotic disorders such as schizophrenia has similar vector to the drugs used in treatment of anxiety and our model suggests a relation between this drug and anxiety which was also confirmed in other study carried out by Meng et al. [Meng et al., 2018]. Brexpiprazole, which is another drug used in treatment of Schizophrenia, shows similar vector representation to the in-use anxiety drugs. A recent study indicates that adjunctive brexpiprazole 2–3 mg/day is efficacious in reducing depressive symptoms in patients with clinically relevant anxiety symptoms who have not responded to antidepressant monotherapy and this drug was well tolerated in patients with clinically relevant anxiety symptoms [Thase et al., 2019].

Zileuton is an orally active inhibitor of 5-lipoxygenase used for the maintenance treatment of Asthma. Our algorithm considers the existence of a link between this drug and depressive disorder, a result that has been confirmed in [Li et al., 2018] which mentions that Zileuton abrogates LPS-induced depressive-like behaviors and neuroinflammation, and enhances response element-binding protein (CREB)/brain-derived neurotrophic factor (BDNF) signaling in the hippocampus, suggesting that Zileuton

could have potential therapeutic value for depression. Testosterone Enanthate is an androgen and anabolic steroid (AAS) medication which mainly is used in the treatment of low testosterone levels in men. In the literature, several recent studies have established positive role of testosterone in the treatment of depressive symptoms or depression in hypogonadal subjects [Royer et al., 2018] [Bhasin and Seidman, 2019]. These results agree with our data where there is a high similarity between the embeddings of Testosterone Enanthate and those drugs used in treatment of depressive disorder. The same conclusion is valid for Asenapine as it has similar interactions to those drugs are prescribed for treating depression. This has been confirmed by Vieta and his colleague [Vieta and Montes, 2018] where they conclude that Asenapine is effective in alleviating the symptoms of depression in patients with bipolar I disorder during mania.

The primary use of Clofazimine is for the treatment of Leprosy, however, several new studies suggest that it can be beneficial in treatment of a group of cancers including variations of Lymphoma [Durusu et al., 2017], [Lentz et al., 2019]. Ribavirin is used primarily to treat Viral Hemorrhagic Fevers, Hepatitis C and a variety of fevers such as Lassa fever, Crimean-Congo Hemorrhagic Fever and Venezuelan Hemorrhagic Fever. Lately, a study conducted by Guadalupe et al. confirmed that Ribavirin can be proposed to exert growth inhibitory effects on Lymphoma cell lines, particularly Hut78 cells and suggests that this medication has anti-Lymphoma potential, however, they emphasize that more clinical studies are required to determine the effectiveness of Ribavirin as a therapeutic agent for treating lymphoma. [Dominguez-Gomez et al., 2019]. All these results are also confirmed by our method through indicating strong connections between the generated embedding of each drug and the corresponding diseases (see Table 4.10).

In this chapter, we presented two novel drug repurposing methods based on graph learning. First, we showed the benefits of using adaptive sampling and how it helps generating faster embeddings with at least as high quality embedding as the fixed permutation strategy. It was also shown that it is possible to analyse the embedding space of drug-disease-protein graph and explore its special properties which help accelerate the discovery of new uses. Further, it was proven and shown with evidences that the Shapley space is more linearly separable and also more appropriate for clustering and it can constantly reach better groups quality than the embedding space. The clusters in Shapley space was then used to discover drugs with similar positive and negative response factors and consequently resulted in finding more repurposing opportunities. In the second study, we demonstrated the possibility of learning the drug representations of drug-action, drug-mechanism and drug-target interactions in form of multiple independent graphs, creating a new drug embeddings by unifying the

separately learned drug representations and finally utilizing them in order to predict drug-disease interactions. Finally, we showed that false positives may be used to find new uses of medications and to identify repurposing opportunities. In the next chapter, the conclusion and our plans for future studies are presented.

Chapter 5

Conclusion

Drug discovery is a complex process and often takes several years until a new drug is mass produced. Before a drug can even go to test process, it most be researched. The whole process begins with the identification of the target (protein or a pathway which is involved in a particular disease) for a drug to act upon. In this step, before going any further, researchers try to make sure that a certain target is actually involved in the disease. This stage usually takes between 3 to 6 years. Once the target is identified and validated, the search for finding the substances which have effect on it begins. This involves the laboratory testing of a tremendous number of compounds, often 10,000 or more, to determine which shows some activity against the target. At this phase, which is called preclinical studies, it is unlikely that the perfect candidate suddenly presents itself, but compounds that show promise will be identified. This phase often takes up to two years. The clinical trials, as the last phase, involve human tests and and provide information on efficiency and safety. This phase is the longest phase and can takes between 4 to 7 years.

Since repurposing is a shortcut for this cumbersome process, it has drown a lot of attention in drug discovery community. Some of new uses of drugs are the result of some random accident which was reported by patients. The mechanism-of-action is the function of drug in the cellular machinery and learning mechanisms-of-action is a crucial for identifying new opportunities for the use of already approved drugs.

Graphs are essential structures that model a variety of real-world problems. For a drug to be metabolized or act upon a receptor or by an enzyme, it must first be bound to that enzyme or the receptor protein. On the other hand, each drug treats a disease or a group of health conditions. Therefore, connections between drugs, proteins, and diseases suggest that these relations can be represented in a graph structure. Therefore, discovering the hidden relations between drugs and diseases in a graph structure and

using this information to identify new uses of drugs has recently drawn researchers' attention.

One recent promising approach to drug repositioning is to take advantage of machine learning algorithms to learn patterns in biological data related to drugs and then link them to the potential of treating specific diseases. However, in order to model graph structure, we first need to find a way to represent it in common structures like tabular form. One traditional way is to use the adjacency matrix and use it as the input of machine learning algorithms. Nevertheless, such a matrix is usually sparse, making it a poor choice due to high memory complexity. Therefore, it is indispensable to learn dense representations of graphs while maintaining their characteristics in their original form.

Node representation learning is a recently emerged area that has demonstrated its efficacy for generating low-dimensional representations of nodes, which leads to increased attention towards this research field. Several approaches have been proposed during the past few years to learn the node embeddings efficiently, although the majority suffers from either low-quality node representations or slow training.

The influence of a node is defined as its ability to spread information through network and affect its neighborhood. In short, ranking nodes based on their influence is identical to ordering them based on their importance. The current node representation learning approaches give same importance to all nodes and dedicate same effort to learn nodes surrounding information which result in a slow training and low-quality embeddings for influential nodes.

In this work, a new approach for selecting drug repurposing candidates was introduced. We first propose a new graph representation learning algorithm with adaptive sampling based on nodes' influence factors. The adaptive sampling strategy help dedicate more learning effort for important nodes and lesser for others, mainly isolated from the rest of the nodes. This results in fewer training sequences for sparse graphs, leading to faster learning. We showed that it is possible to reduce training sequences by up to 75% while maintaining learnability. We also demonstrated that it is possible to join model predictability and explainability by migrating from the original feature space to Shapley space. After generating the drug-disease-protein embeddings and applying the Shapley transformation based on the opinion of a Random Forest model trained on a small proportion of data, clustering and link prediction tasks were performed in this new space. It was pointed out that diseases with similar response factors are likely related to similar symptoms, and their drugs are possibly interchangeable. Furthermore, we showed that only having similar response factors is enough to detect groups of medications with a high chance of interchangeability, speeding up the selection of

drug repositioning candidates. Furthermore, we examined the possibility of predicting drug-disease links through learning the interactions existing in the drug-mechanism, drug-action and drug-target graphs. After training the node vector representations of each graph separately, the final embeddings were constructed through the concatenation of vectors for each node and subsequently they were used to predict the associated diseases. Finally we showed that the false positives can indeed be used to find drug repurposing opportunities.

As mentioned, drug repurposing aims to reduce the time and cost of finding new drugs. Nonetheless, the whole process of identifying new uses of already known drugs can still be complex and time consuming. The methods developed during the current study aim to simplify this complex process and make it possible to explain why a specific drug can be a good repositioning candidate for a specific health condition.

Understanding the meaning of each latent space variable for an individual or a group of diseases may reveal even more hidden characteristics of drugs and open up new ways of explaining why a drug is repositionable for a specific condition. Moreover, it was shown that efficient sampling could help the model learn more with less data. The proposed sampling strategy can be applied in other node representation learning algorithms, although depending on how a specific algorithm works, adapting the node influence metrics may be necessary to achieve the best results. Therefore, future works concern deeper analysis of latent space variables, experimenting the adaptive sampling in other node representation learning algorithms mentioned in the previous chapters, and testing the effect of other node influence metrics.

5.1 Publications during candidature

1. Jalilifard, A., Veloso D. (2022, July) Drug Repurposing Opportunities in Shapley Space. In 2022 IEEE World Congress on Computational Intelligence (WCCI). (Qualis A1)

2. Chen, D., Jalilifard, A., Veloso, A., Ziviani, N. (2020, July). Modeling pharmacological effects with multi-relation unsupervised graph embedding. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-7). (Qualis A1)

3. Jalilifard, A., Veloso, A. Drug Repurposing Through Learning Complex Structures of Graphs. Under review in the jornal of Expert Systems with Applications (Impact factor = 8.665).

Bibliography

[Abe et al., 2016] Abe, K., Chiba, Y., Katsuse, O., and Hirayasu, Y. (2016). A case of parkinson disease with both visual hallucination and pain improved by gabapentin. *Clinical neuropharmacology*, 39(1):55--56.

[Adadi and Berrada, 2018] Adadi, A. and Berrada, M. (2018). Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138--52160.

[Ahmed et al., 2013] Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V., and Smola, A. J. (2013). Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*, pages 37--48. ACM.

[Aitchison, 1982] Aitchison, J. (1982). The statistical analysis of compositional data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 44(2):139--160.

[Angles and Gutierrez, 2008] Angles, R. and Gutierrez, C. (2008). Survey of graph database models. *ACM Computing Surveys (CSUR)*, 40(1):1.

[Anighoro et al., 2014] Anighoro, A., Bajorath, J., and Rastelli, G. (2014). Polypharmacology: Challenges and opportunities in drug discovery. *J. Med. Chem.*, 57(19):7874--7887.

[Asratian et al., 1998] Asratian, A. S., Denley, T. M., and Häggkvist, R. (1998). *Bipartite graphs and their applications*, volume 131. Cambridge university press.

[Atkinson Jr et al., 1975] Atkinson Jr, A. J., Finkel, M. J., Burns, J. J., Hitchings, G. H., Kemp, B. A., and de Dennis, S. R. (1975). Panel on public service drugs and new uses for old drugs. *Clinical Pharmacology & Therapeutics*, 18(5part2):659--662.

[Balasubramanian and Schwartz, 2002] Balasubramanian, M. and Schwartz, E. L. (2002). The isomap algorithm and topological stability. *Science*, 295(5552):7--7.

[Barrat et al., 2004] Barrat, A., Barthelemy, M., Pastor-Satorras, R., and Vespignani, A. (2004). The architecture of complex weighted networks. *Proceedings of the national academy of sciences*, 101(11):3747--3752.

[Barrenäs et al., 2012] Barrenäs, F., Chavali, S., Alves, A. C., Coin, L., Jarvelin, M.-R., Jörnsten, R., Langston, M. A., Ramasamy, A., Rogers, G., Wang, H., et al. (2012). Highly interconnected genes in disease-specific networks are enriched for disease-associated polymorphisms. *Genome biology*, 13(6):1--9.

[Barrett et al., 2006] Barrett, T., Troup, D. B., Wilhite, S. E., Ledoux, P., Rudnev, D., Evangelista, C., Kim, I. F., Soboleva, A., Tomashevsky, M., and Edgar, R. (2006). Ncbi geo: mining tens of millions of expression profiles—database and tools update. *Nucleic acids research*, 35(suppl_1):D760--D765.

[Baues, 1995] Baues, H. J. (1995). Homotopy types. *Handbook of algebraic topology*, pages 1--72.

[Belkin and Niyogi, 2002] Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in neural information processing systems*, pages 585--591.

[Berdigaliyev and Aljofan, 2020] Berdigaliyev, N. and Aljofan, M. (2020). An overview of drug discovery and development. *Future Medicinal Chemistry*, 12(10):939--947.

[Berrar, 2019] Berrar, D. (2019). Cross-validation.

[Bhasin and Seidman, 2019] Bhasin, S. and Seidman, S. (2019). Testosterone treatment of depressive disorders in men: Too much smoke, not enough high-quality evidence. *JAMA psychiatry*, 76(1):9--10.

[Bhattacharyya, 1943] Bhattacharyya, A. (1943). On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Calcutta Math. Soc.*, 35:99--109.

[Bojchevski et al., 2018] Bojchevski, A., Shchur, O., Zügner, D., and Günnemann, S. (2018). Netgan: Generating graphs via random walks. *arXiv preprint arXiv:1803.00816*.

[Bosco et al., 2019] Bosco, G., Ostardo, E., Rizzato, A., Garetto, G., Paganini, M., Melloni, G., Giron, G., Pietrosanti, L., Martinelli, I., and Camporesi, E. (2019). Clinical and morphological effects of hyperbaric oxygen therapy in patients with interstitial cystitis associated with fibromyalgia. *BMC urology*, 19(1):108.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5--32.

[Brown et al., 2006] Brown, J. S., Vittinghoff, E., Lin, F., Nyberg, L. M., Kusek, J. W., and Kanaya, A. M. (2006). Prevalence and risk factors for urinary incontinence in women with type 2 diabetes and impaired fasting glucose: findings from the national health and nutrition examination survey (nhanes) 2001–2002. *Diabetes care*, 29(6):1307--1312.

[Bubnova et al., 2019] Bubnova, M., Aronov, D., and Persiyanova-Dubrova, A. (2019). Effects of rosuvastatin and atorvastatin on blood pressure, cerebral blood flow, endothelial function, angiotensin ii in patients with ischemic stroke-complicated hypertension. *Journal of Hypertension*, 37.

[Bui et al., 2017] Bui, Q. V., Sayadi, K., Amor, S. B., and Bui, M. (2017). Combining latent dirichlet allocation and k-means for documents clustering: effect of probabilistic based distance measures. In *Asian conference on intelligent information and database systems*, pages 248--257. Springer.

[Burt, 2004] Burt, R. S. (2004). Structural holes and good ideas. *American journal of sociology*, 110(2):349--399.

[Cao et al., 2015] Cao, S., Lu, W., and Xu, Q. (2015). Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM international on conference on information and knowledge management*, pages 891--900. ACM.

[Cao et al., 2016] Cao, S., Lu, W., and Xu, Q. (2016). Deep neural networks for learning graph representations. In *Thirtieth AAAI Conference on Artificial Intelligence*.

[Car, 2012] Car, D. (2012). *Polypharmacology in Drug Discovery*. Wiley.

[Chatr-Aryamontri et al., 2015] Chatr-Aryamontri, A., Breitkreutz, B.-J., Oughtred, R., Boucher, L., Heinicke, S., Chen, D., Stark, C., Breitkreutz, A., Kolas, N., O'Donnell, L., et al. (2015). The biogrid interaction database: 2015 update. *Nucleic acids research*, 43(D1):D470--D478.

[Chatr-aryamontri et al., 2015] Chatr-aryamontri, A., Breitkreutz, B.-J., Oughtred, R., Boucher, L., Heinicke, S., Chen, D., Stark, C., Breitkreutz, A., Kolas, N., O'Donnell, L., Reguly, T., Nixon, J., Ramage, L., Winter, A., Sellam, A., Chang, C., Hirschman, J., Theesfeld, C., Rust, J., Livstone, M., Dolinski, K., and Tyers, M. (2015). The biogrid interaction database: 2015 update. *Nucleic Acids Research*, 43:D470--D478.

[Chen et al., 2015] Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., et al. (2015). Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1--4.

[Chung, 2010] Chung, F. (2010). Graph theory in the information age. *Notices of the AMS*, 57(6):726--732.

[Coleman et al., 2018] Coleman, D. J., Lee, W., Chang, S., Silverman, R. H., Lloyd, H. O., Daly, S., and Tsang, S. H. (2018). Treatment of macular degeneration with sildenafil: results of a two-year trial. *Ophthalmologica*, 240(1):45--54.

[Corsico et al., 2019] Corsico, A. G., Leonardi, S., Licari, A., Marseglia, G., Miraglia del Giudice, M., Peroni, D. G., Salpietro, C., and Ciprandi, G. (2019). Focus on the cetirizine use in clinical practice: a reappraisal 30 years later. *Multidisciplinary Respiratory Medicine*, 14(1):1--7.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273--297.

[Cranston and Yancey, 2020] Cranston, D. W. and Yancey, M. P. (2020). Sparse graphs are near-bipartite. *SIAM Journal on Discrete Mathematics*, 34(3):1725--1768.

[Dakshanamurthy et al., 2012] Dakshanamurthy, S., Issa, N. T., Assefnia, S., Seshasayee, A., Peters, O. J., Madhavan, S., Uren, A., Brown, M. L., and Byers, S. W. (2012). Predicting new indications for approved drugs using a proteochemometric method. *Journal of medicinal chemistry*, 55(15):6832--6848.

[Danisch et al., 2018] Danisch, M., Balalau, O., and Sozio, M. (2018). Listing k-cliques in sparse real-world graphs. In *Proceedings of the 2018 World Wide Web Conference*, pages 589--598.

[Dawande et al., 2001] Dawande, M., Keskinocak, P., Swaminathan, J. M., and Tayur, S. (2001). On bipartite and multipartite clique problems. *Journal of Algorithms*, 41(2):388--403.

[Deepika and Geetha, 2018] Deepika, S. and Geetha, T. (2018). A meta-learning framework using representation learning to predict drug-drug interaction. *Journal of biomedical informatics*, 84:136--147.

[Dominguez-Gomez et al., 2019] Dominguez-Gomez, G., Cortez-Pedroza, D., Chavez-Blanco, A., Taja-Chayeb, L., Hidalgo-Miranda, A., Cedro-Tanda, A., Beltran-Anaya,

F., Diaz-Chavez, J., Schcolnik-Cabrera, A., Gonzalez-Fierro, A., et al. (2019). Growth inhibition and transcriptional effects of ribavirin in lymphoma. *Oncology reports*, 42(3):1248--1256.

[Donato and Brown, 2019] Donato, A. and Brown, K. (2019). In black africans with hypertension, amlodipine-based therapy vs perindopril–hydrochlorothiazide improved bp control. *Annals of internal medicine*, 171(2):JC5--JC5.

[Donner et al., 2018] Donner, Y., Kazmierczak, S., and Fortney, K. (2018). Drug repurposing using deep embeddings of gene expression profiles. *Molecular pharmaceutics*, 15(10):4314--4325.

[Douville and Nath, 2017] Douville, R. N. and Nath, A. (2017). Human endogenous retrovirus-k and tdp-43 expression bridges als and hiv neuropathology. *Frontiers in microbiology*, 8:1986.

[Durusu et al., 2017] Durusu, İ. Z., Hüsnügil, H. H., Ataş, H., Biber, A., Gerekçi, S., Güleç, E. A., and Özen, C. (2017). Anti-cancer effect of clofazimine as a single agent and in combination with cisplatin on u266 multiple myeloma cell line. *Leukemia research*, 55:33--40.

[Egozcue et al., 2003] Egozcue, J. J., Pawlowsky-Glahn, V., Mateu-Figueras, G., and Barcelo-Vidal, C. (2003). Isometric logratio transformations for compositional data analysis. *Mathematical geology*, 35(3):279--300.

[Ehrt et al., 2016] Ehrt, C., Brinkjost, T., and Koch, O. (2016). Impact of binding site comparisons on medicinal chemistry and rational molecular design. *Journal of medicinal chemistry*, 59(9):4121--4151.

[Ekinci and Ekinci, 2018] Ekinci, A. and Ekinci, O. (2018). Brexpiprazole: A partial dopamine agonist for the treatment of schizophrenia. *Reviews on Recent Clinical Trials*, 13(1):37--44.

[Enriquez-Marulanda et al., 2017] Enriquez-Marulanda, A., Valderrama-Chaparro, J., Parrado, L., Vélez, J. D., Granados, A. M., Orozco, J. L., and Quiñones, J. (2017). Cerebral toxoplasmosis in an ms patient receiving fingolimod. *Multiple sclerosis and related disorders*, 18:106--108.

[Epasto and Perozzi, 2019] Epasto, A. and Perozzi, B. (2019). Is a single embedding enough? learning node representations that capture multiple social contexts. In *The World Wide Web Conference*, pages 394--404.

[Escoli et al., 2018] Escoli, R., Oliveira, R., Santos, P., and Lobos, A. V. (2018). Kidney diseases with ocular involvement: a systematic review. *Portuguese Journal of Nephrology & Hypertension*, 32(3):268--278.

[Fiscon et al., 2021] Fiscon, G., Conte, F., Amadio, S., Volonté, C., and Paci, P. (2021). Drug repurposing: a network-based approach to amyotrophic lateral sclerosis. *Neurotherapeutics*, 18(3):1678--1691.

[Fiscon and Paci, 2021] Fiscon, G. and Paci, P. (2021). Saverunner: an r-based tool for drug repurposing. *BMC bioinformatics*, 22(1):1--10.

[Fullerton et al., 2010] Fullerton, C. A., Busch, A. B., and Frank, R. G. (2010). The rise and fall of gabapentin for bipolar disorder: a case study on off-label pharmaceutical diffusion. *Medical care*, 48(4):372.

[Gamage et al., 2020] Gamage, A., Chien, E., Peng, J., and Milenkovic, O. (2020). Multi-motifgan (mmgan): Motif-targeted graph generation and prediction. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4182--4186. IEEE.

[Ganguly and Cambier, 2021] Ganguly, S. and Cambier, T. (2021). *Quantum Computing with Silq Programming: Get up and running with quantum computing with the simplicity of this new high-level programming language*. Packt Publishing Ltd.

[Gao et al., 2018] Gao, K. Y., Fokoue, A., Luo, H., Iyengar, A., Dey, S., and Zhang, P. (2018). Interpretable drug target prediction using deep neural representation. In *IJCAI*, pages 3371--3377.

[GNS et al., 2019] GNS, H. S., Saraswathy, G., Murahari, M., and Krishnamurthy, M. (2019). An update on drug repurposing: re-written saga of the drug's fate. *Biomedicine & Pharmacotherapy*, 110:700--716.

[Grossberg et al., 2019] Grossberg, G. T., Kohegyi, E., Mergel, V., Josiassen, M. K., Meulien, D., Hobart, M., Slomkowski, M., Baker, R. A., McQuade, R. D., and Cummings, J. L. (2019). Efficacy and safety of brexpiprazole for the treatment of agitation in alzheimer's dementia: two 12-week, randomized, double-blind, placebo-controlled trials. *The American Journal of Geriatric Psychiatry*.

[Grover and Leskovec, 2016a] Grover, A. and Leskovec, J. (2016a). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855--864. ACM.

[Grover and Leskovec, 2016b] Grover, A. and Leskovec, J. (2016b). node2vec: Scalable feature learning for networks. In *Proceedings of KDD*, pages 855--864.

[Gueorguieva et al., 2017] Gueorguieva, N., Valova, I., and Georgiev, G. (2017). M&mfcm: fuzzy c-means clustering with mahalanobis and minkowski distance metrics. *Procedia computer science*, 114:224--233.

[Guymer and Littlejohn, 2019a] Guymer, E. K. and Littlejohn, G. O. (2019a). Pharmacological treatment options for fibromyalgia. *Prevention*, 10:00.

[Guymer and Littlejohn, 2019b] Guymer, E. K. and Littlejohn, G. O. (2019b). Pharmacological treatment options for fibromyalgia. *Prevention*, 10:00.

[Hamilton et al., 2017a] Hamilton, W., Ying, Z., and Leskovec, J. (2017a). Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, pages 1024--1034.

[Hamilton et al., 2017b] Hamilton, W. L., Ying, R., and Leskovec, J. (2017b). Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*.

[Harris, 1981] Harris, E. (1981). Antidepressants: old drugs, new uses. *AJN The American Journal of Nursing*, 81(7):1308--1315.

[Hayes et al., 2020] Hayes, K., Warner, E., Bollinger, C., Wright, D., and Fitch, R. M. (2020). Repository corticotropin injection versus corticosteroids for protection against renal damage in a focal segmental glomerulosclerosis rodent model. *BMC nephrology*, 21(1):1--14.

[Hodos et al., 2016] Hodos, R. A., Kidd, B. A., Shameer, K., Readhead, B. P., and Dudley, J. T. (2016). In silico methods for drug repurposing and pharmacology. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 8(3):186--210.

[Hu and Agarwal, 2009] Hu, G. and Agarwal, P. (2009). Human disease-drug network based on genomic expression profiles. *PloS one*, 4(8):e6536.

[Hung et al., 2019] Hung, C.-I., Liu, C.-Y., Yang, C.-H., and Wang, S.-J. (2019). Migraine with active headache was associated with other painful physical symptoms at two-year follow-up among patients with major depressive disorder. *PloS one*, 14(4):e0216108.

[Hurle et al., 2013] Hurle, M., Yang, L., Xie, Q., Rajpal, D., Sanseau, P., and Agarwal, P. (2013). Computational drug repositioning: from data to therapeutics. *Clinical Pharmacology & Therapeutics*, 93(4):335--341.

[Hurley et al., 2006] Hurley, D. J., Turner, C. L., Yalcin, I., Viktrup, L., and Baygani, S. K. (2006). Duloxetine for the treatment of stress urinary incontinence in women: an integrated analysis of safety. *European Journal of Obstetrics & Gynecology and Reproductive Biology*, 125(1):120--128.

[Jalilifard et al., 2019] Jalilifard, A., Caridá, V., Mansano, A., and Cristo, R. (2019). Can netgan be improved on short random walks? *arXiv preprint arXiv:1905.05298*.

[James et al., 2013] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

[Jin et al., 2012] Jin, G., Fu, C., Zhao, H., Cui, K., Chang, J., and Wong, S. T. (2012). A novel method of transcriptional response analysis to facilitate drug repositioning for cancer therapy. *Cancer research*, 72(1):33--44.

[Jin and Wong, 2014] Jin, G. and Wong, S. T. (2014). Toward better drug repositioning: prioritizing and integrating existing methods into efficient pipelines. *Drug discovery today*, 19(5):637--644.

[Jubelt and Berger, 2001] Jubelt, B. and Berger, J. R. (2001). Does viral disease underlie als?: Lessons from the aids pandemic.

[Kato et al., 2015] Kato, S., Moulder, S. L., Ueno, N. T., Wheler, J. J., Meric-Bernstam, F., Kurzrock, R., and Janku, F. (2015). Challenges and perspective of drug repurposing strategies in early phase clinical trials. *Oncoscience*, 2(6):576.

[Kazemitabar et al., 2017] Kazemitabar, S. J., Amini, A. A., Bloniarz, A., and Talwalkar, A. (2017). Variable importance using decision trees. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 425--434.

[Khalighi et al., 2016] Khalighi, H. R., Mortazavi, H., Mojahedi, S. M., Azari-Marhabi, S., and Abbasabadi, F. M. (2016). Low level laser therapy versus pharmacotherapy in improving myofascial pain disorder syndrome. *Journal of lasers in medical sciences*, 7(1):45.

[Kinnings et al., 2009] Kinnings, S. L., Liu, N., Buchmeier, N., Tonge, P. J., Xie, L., and Bourne, P. E. (2009). Drug discovery using chemical systems biology: repositioning the safe medicine comtan to treat multi-drug and extensively drug resistant tuberculosis. *PLoS computational biology*, 5(7):e1000423.

[Kipf and Welling, 2016a] Kipf, T. N. and Welling, M. (2016a). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

[Kipf and Welling, 2016b] Kipf, T. N. and Welling, M. (2016b). Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.

[Krakovska et al., 2019] Krakovska, O., Christie, G., Sixsmith, A., Ester, M., and Moreno, S. (2019). Performance comparison of linear and non-linear feature selection methods for the analysis of large survey datasets. *Plos one*, 14(3):e0213584.

[Krasselt and Baerwald, 2019] Krasselt, M. and Baerwald, C. (2019). Celecoxib for the treatment of musculoskeletal arthritis. *Expert opinion on pharmacotherapy*, 20(14):1689--1702.

[Kumar et al., 2019] Kumar, A. P., Lukman, S., and Nguyen, M. N. (2019). Drug repurposing and multi-target therapies.

[Kwon et al., 2020] Kwon, O.-S., Lee, H., Kong, H.-J., Kwon, E.-J., Park, J. E., Lee, W., Kang, S., Kim, M., Kim, W., and Cha, H.-J. (2020). Connectivity map-based drug repositioning of bortezomib to reverse the metastatic effect of galnt14 in lung cancer. *Oncogene*, 39(23):4567--4580.

[Lee and Vaithilingam, 2011] Lee, K.-G. and Vaithilingam, I. (2011). Bilateral optic neuropathy—a rare uraemic manifestation of end-stage renal disease. *NDT plus*, 4(6):455.

[Lekka et al., 2011] Lekka, E., Deftereos, S. N., Persidis, A., Persidis, A., and Andronis, C. (2011). Literature analysis for systematic drug repurposing: a case study from biovista. *Drug Discovery Today: Therapeutic Strategies*, 8(3-4):103--108.

[Lentz et al., 2019] Lentz, F., Reiling, N., Spengler, G., Kincses, A., Csonka, A., Molnár, J., and Hilgeroth, A. (2019). Dually acting nonclassical 1, 4-dihydropyridines promote the anti-tuberculosis (tb) activities of clofazimine. *Molecules*, 24(16):2873.

[Li et al., 2019] Li, B., Drozd, A., Guo, Y., Liu, T., Matsuoka, S., and Du, X. (2019). Scaling word2vec on big corpus. *Data Science and Engineering*, 4(2):157--175.

[Li et al., 2018] Li, D.-D., Xie, H., Du, Y.-F., Long, Y., Reed, M. N., Hu, M., Suppiramaniam, V., Hong, H., and Tang, S.-S. (2018). Antidepressant-like effect of zileuton is accompanied by hippocampal neuroinflammation reduction and creb/bdnf upregulation in lipopolysaccharide-challenged mice. *Journal of affective disorders*, 227:672--680.

[Liberti et al., 2014] Liberti, L., Lavor, C., Maculan, N., and Mucherino, A. (2014). Euclidean distance geometry and applications. *SIAM review*, 56(1):3--69.

[Liu and Krishnan, 2021] Liu, R. and Krishnan, A. (2021). Pecanpy: a fast, efficient and parallelized python implementation of node2vec. *Bioinformatics*, 37(19):3377--3379.

[Liu et al., 2010] Liu, X., Ouyang, S., Yu, B., Liu, Y., Huang, K., Gong, J., Zheng, S., Li, Z., Li, H., and Jiang, H. (2010). Pharmmapper server: a web server for potential drug target identification using pharmacophore mapping approach. *Nucleic acids research*, 38(suppl_2):W609--W614.

[Liu et al., 2013] Liu, Z., Fang, H., Reagan, K., Xu, X., Mendrick, D. L., Slikker Jr, W., and Tong, W. (2013). In silico drug repositioning–what we need to know. *Drug discovery today*, 18(3-4):110--115.

[Lundberg and Lee, 2017a] Lundberg, S. M. and Lee, S.-I. (2017a). A unified approach to interpreting model predictions. In *Proceedings of NIPS*, pages 4765--4774.

[Lundberg and Lee, 2017b] Lundberg, S. M. and Lee, S.-I. (2017b). A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 4765--4774. Curran Associates, Inc.

[MacQueen et al., 1967] MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281--297. Oakland, CA, USA.

[Makgai et al., 2021] Makgai, S., Bekker, A., and Arashi, M. (2021). Compositional data modeling through dirichlet innovations. *Mathematics*, 9(19):2477.

[Massey Jr, 1951] Massey Jr, F. J. (1951). The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68--78.

[Mathewson, 1982] Mathewson, M. (1982). New uses for old drugs: vasodilators (ceu home study). *Critical care update*, 9(11):7.

[Menche et al., 2015] Menche, J., Sharma, A., Kitsak, M., Ghiassian, S., Vidal, M., Loscalzo, J., and Barabási, A.-L. (2015). Uncovering disease-disease relationships through the incomplete interactome. *Science*, 347:1257601.

[Meng et al., 2018] Meng, Q., Li, R., Hou, F., and Zhang, Q. (2018). Effects of chlorpromazine on sleep quality, clinical and emotional measures among patients with schizophrenia. *Clinical neurology and neurosurgery*, 165:134--138.

[Meng et al., 2011] Meng, X.-Y., Zhang, H.-X., Mezei, M., and Cui, M. (2011). Molecular docking: a powerful approach for structure-based drug discovery. *Current computer-aided drug design*, 7(2):146--157.

[Mikolov et al., 2013a] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013a). Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111--3119.

[Mikolov et al., 2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. pages 3111--3119.

[Moradi and Samwald, 2021] Moradi, M. and Samwald, M. (2021). Post-hoc explanation of black-box classifiers using confident itemsets. *Expert Systems with Applications*, 165:113941.

[Moulignier et al., 2001] Moulignier, A., Moulonguet, A., Pialoux, G., and Rozenbaum, W. (2001). Reversible als-like disorder in hiv infection. *Neurology*, 57(6):995--1001.

[Mousavi et al., 2020] Mousavi, S. Z., Rahmanian, M., and Sami, A. (2020). A connectivity map-based drug repurposing study and integrative analysis of transcriptomic profiling of sars-cov-2 infection. *Infection, Genetics and Evolution*, 86:104610.

[Mursalin et al., 2017] Mursalin, M., Zhang, Y., Chen, Y., and Chawla, N. V. (2017). Automated epileptic seizure detection using improved correlation-based feature selection with random forest classifier. *Neurocomputing*, 241:204--214.

[Nam et al., 2021] Nam, H. J., Kim, Y. E., Moon, B.-S., Kim, H. Y., Jung, D., Choi, S., Jang, J. W., Nam, D.-H., and Cho, H. (2021). Azathioprine antagonizes aber-

rantly elevated lipid metabolism and induces apoptosis in glioblastoma. *Iscience*, 24(3):102238.

[Noble, 2006] Noble, W. S. (2006). What is a support vector machine? *Nature biotechnology*, 24(12):1565--1567.

[Novac, 2013] Novac, N. (2013). Challenges and opportunities of drug repositioning. *Trends in pharmacological sciences*, 34(5):267--272.

[Ong et al., 2007] Ong, L., Cheung, B., Man, Y., Lau, P., and Lam, S. (2007). Prevalence, awareness, treatment, and control of hypertension among united states adults 1999–2004. *Hypertension*, 1(49):69--75.

[Ou et al., 2016] Ou, M., Cui, P., Pei, J., Zhang, Z., and Zhu, W. (2016). Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1105--1114. ACM.

[Pagadala et al., 2017] Pagadala, N. S., Syed, K., and Tuszynski, J. (2017). Software for molecular docking: a review. *Biophysical reviews*, 9(2):91--102.

[Pan and Shen, 2009] Pan, X.-Y. and Shen, H.-B. (2009). Robust prediction of b-factor profile from sequence using two-stage svr based on random forest feature selection. *Protein and peptide letters*, 16(12):1447--1454.

[Pande et al., 2000] Pande, A. C., Crockatt, J. G., Janney, C. A., Werth, J. L., Tsaroucha, G., and Group, G. B. D. S. (2000). Gabapentin in bipolar disorder: a placebo-controlled trial of adjunctive therapy 1. *Bipolar disorders*, 2(3p2):249--255.

[Peck et al., 2020] Peck, J., Urits, I., Crane, J., McNally, A., Noor, N., Patel, M., Berger, A. A., Cornett, E. M., Kassem, H., Kaye, A. D., et al. (2020). Oral muscle relaxants for the treatment of chronic pain associated with cerebral palsy. *Psychopharmacology bulletin*, 50(4 Suppl 1):142.

[Perozzi et al., 2014a] Perozzi, B., Al-Rfou, R., and Skiena, S. (2014a). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701--710. ACM.

[Perozzi et al., 2014b] Perozzi, B., Al-Rfou, R., and Skiena, S. (2014b). Deepwalk: Online learning of social representations. In *Proceedings of KDD*, pages 701--710.

[Pietzner et al., 2021] Pietzner, M., Wheeler, E., Carrasco-Zanini, J., Cortes, A., Koprulu, M., Wörheide, M. A., Oerton, E., Cook, J., Stewart, I. D., Kerrison, N. D., et al. (2021). Mapping the proteo-genomic convergence of human diseases. *Science*, page eabj1541.

[Pimentel et al., 2019] Pimentel, T., Castro, R., Veloso, A., and Ziviani, N. (2019). Efficient estimation of node representations in large graphs using linear contexts. In *2019 International joint conference on neural networks (IJCNN)*, pages 1--8. IEEE.

[Pintelas et al., 2020] Pintelas, E., Livieris, I. E., and Pintelas, P. (2020). A grey-box ensemble model exploiting black-box accuracy and white-box intrinsic interpretability. *Algorithms*, 13(1):17.

[Pushpakom et al., 2019] Pushpakom, S., Iorio, F., Eyers, P. A., Escott, K. J., Hopper, S., Wells, A., Doig, A., Guilliams, T., Latimer, J., McNamee, C., et al. (2019). Drug repurposing: progress, challenges and recommendations. *Nature Reviews Drug Discovery*, 18(1):41.

[Ribeiro et al., 2016] Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135--1144.

[Rousseeuw, 1987] Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53--65.

[Roweis and Saul, 2000] Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323--2326.

[Royer et al., 2018] Royer, T., Michel, B., and Javelot, H. (2018). A case of resistant depression stabilized by testosterone enanthate in a context of hypoandrogenism. *Med Case Rep*, 4(2):71.

[Rudin, 2019] Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206--215.

[Shim and Liu, 2014] Shim, J. S. and Liu, J. O. (2014). Recent advances in drug repositioning for the discovery of new anticancer drugs. *International journal of biological sciences*, 10(7):654.

[Siciliano et al., 2019] Siciliano, M., Trojano, L., Trojsi, F., Monsurrò, M. R., Tedeschi, G., and Santangelo, G. (2019). Assessing anxiety and its correlates in amyotrophic lateral sclerosis: The state-trait anxiety inventory. *Muscle & nerve*.

[Singh et al., 2009] Singh, S., Kubica, J., Larsen, S., and Sorokina, D. (2009). Parallel large scale feature selection for logistic regression. In *Proceedings of the 2009 SIAM international conference on data mining*, pages 1172--1183. SIAM.

[Sun et al., 2017] Sun, P., Guo, J., Winnenburg, R., and Baumbach, J. (2017). Drug repurposing by integrated literature mining and drug–gene–disease triangulation. *Drug discovery today*, 22(4):615--619.

[Szklarczyk et al., 2017] Szklarczyk, D., Morris, J., Cook, H., Kuhn, M., Wyder, S., Simonovic, M., Santos, A., Doncheva, N., Roth, A., Bork, P., Jensen, L., and von Mering, C. (2017). The string database in 2017: quality-controlled protein–protein association networks, made broadly accessible. *Nucleic Acids Research*, 45:D362--D368.

[Tarakad and Jankovic, 2018] Tarakad, A. and Jankovic, J. (2018). Essential tremor and parkinson's disease: exploring the relationship. *Tremor and Other Hyperkinetic Movements*, 8.

[Thase et al., 2019] Thase, M. E., Weiller, E., Zhang, P., Weiss, C., and McIntyre, R. S. (2019). Adjunctive brexpiprazole in patients with major depressive disorder and anxiety symptoms: post hoc analyses of three placebo-controlled studies. *Neuropsychiatric disease and treatment*, 15:37.

[Thorman et al., 2020] Thorman, A. W., Reigle, J., Chutipongtanate, S., Shamsaei, B., Pilarczyk, M., Fazel-Najafabadi, M., Adamczak, R., Kouril, M., Morrow, A. L., Czyzyk-Krzeska, M. F., et al. (2020). Accelerating drug discovery and repurposing by combining transcriptional signature connectivity with docking. *bioRxiv*.

[Tobinick, 2009a] Tobinick, E. L. (2009a). The value of drug repositioning in the current pharmaceutical market. *Drug News Perspect*, 22(2):119--125.

[Tobinick, 2009b] Tobinick, E. L. (2009b). The value of drug repositioning in the current pharmaceutical market. *Drug News Perspect*, 22(2):119--125.

[Utkin et al., 2021] Utkin, L. V., Konstantinov, A. V., and Vishniakov, K. A. (2021). An imprecise shap as a tool for explaining the class probability distributions under limited training data. *arXiv preprint arXiv:2106.09111*.

[van Beekhuizen et al., 2018] van Beekhuizen, H. J., Oost, J., and van der Meijden, W. I. (2018). Generalized unprovoked vulvodynia; a retrospective study on the efficacy of treatment with amitriptyline, gabapentin or pregabalin. *European Journal of Obstetrics & Gynecology and Reproductive Biology*, 220:118--121.

[Van der Maaten and Hinton, 2008] Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

[Vieta and Montes, 2018] Vieta, E. and Montes, J. M. (2018). A review of asenapine in the treatment of bipolar disorder. *Clinical drug investigation*, 38(2):87--99.

[Vogt et al., 2014] Vogt, I., Prinz, J., and Campillos, M. (2014). Molecularly and clinically related drugs and diseases are enriched in phenotypically similar drug-disease pairs. *Genome medicine*, 6(7):1--17.

[Wang et al., 2016a] Wang, D., Cui, P., and Zhu, W. (2016a). Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1225--1234. ACM.

[Wang et al., 2016b] Wang, G., Xu, S., Cao, C., Dong, J., Chu, Y., He, G., and Xu, Z. (2016b). Evidence from a large-scale meta-analysis indicates eczema reduces the incidence of glioma. *Oncotarget*, 7(38):62598.

[Wang et al., 2011] Wang, J., Ono, K., Dickstein, D. L., Arrieta-Cruz, I., Zhao, W., Qian, X., Lamparello, A., Subnani, R., Ferruzzi, M., Pavlides, C., et al. (2011). Carvedilol as a potential novel agent for the treatment of alzheimer's disease. *Neurobiology of aging*, 32(12):2321--e1.

[Winkelmayer et al., 2001] Winkelmayer, W. C., Eigner, M., Berger, O., Grisold, W., and Leithner, C. (2001). Optic neuropathy in uremia: an interdisciplinary emergency. *American Journal of Kidney Diseases*, 37(3):e23--1.

[Wishart et al., 2008] Wishart, D., Knox, C., Guo, A., Cheng, D., Shrivastava, S., Tzur, D., Gautam, B., and Hassanali, M. (2008). Drugbank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Research*, 36:D901--D906.

[Wishart et al., 2018] Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., Sajed, T., Johnson, D., Li, C., Sayeeda, Z., et al. (2018). Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 46(D1):D1074--D1082.

[Wold et al., 1987] Wold, S., Esbensen, K., and Geladi, P. (1987). Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37--52.

[Woodruff, 2017] Woodruff, K. (2017). Introduction to boosted decision trees. In *Machine Learning Group Meeting*.

[Wu et al., 2017] Wu, H., Miller, E., Wijegunawardana, D., Regan, K., Payne, P. R., and Li, F. (2017). Md-miner: a network-based approach for personalized drug repositioning. *BMC systems biology*, 11(5):86.

[Xia et al., 2021] Xia, F., Sun, K., Yu, S., Aziz, A., Wan, L., Pan, S., and Liu, H. (2021). Graph learning: A survey. *IEEE Transactions on Artificial Intelligence*, 2(2):109--127.

[Yan et al., 2019] Yan, C.-K., Wang, W.-X., Zhang, G., Wang, J.-L., and Patel, A. (2019). Birwdda: A novel drug repositioning method based on multisimilarity fusion. *Journal of Computational Biology*, 26(11):1230--1242.

[Yegnanarayana, 2009] Yegnanarayana, B. (2009). *Artificial neural networks*. PHI Learning Pvt. Ltd.

[Yonkman, 1959] Yonkman, F. (1959). New drugs for old uses and new uses for old drugs. *Journal-Michigan State Medical Society*, 58(6):913.

[Zeng et al., 2019] Zeng, X., Zhu, S., Liu, X., Zhou, Y., Nussinov, R., and Cheng, F. (2019). deepdr: a network-based deep learning approach to in silico drug repositioning. *Bioinformatics*, 35(24):5191--5198.

[Zhang et al., 2016] Zhang, W., Bai, Y., Wang, Y., and Xiao, W. (2016). Polypharmacology in drug discovery: A review from systems pharmacology perspective. *Curr Pharm Des.*, 22(21):3171--3181.

[Zhou et al., 2020] Zhou, Y., Hou, Y., Shen, J., Huang, Y., Martin, W., and Cheng, F. (2020). Network-based drug repurposing for novel coronavirus 2019-ncov/sars-cov-2. *Cell Discovery*, 6(1):1--18.

[Zitnik et al., 2018] Zitnik, M., Agrawal, M., and Leskovec, J. (2018). Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, 34(13):i457--i466.

Appendix A

The complete list of the identified groups of similar drugs and diseases

As mentioned earlier, the identified clusters help find groups which likely have repositionable members. Although during our initial testes we used the elbow method and set the number of clusters to 11, depending on the level of granularity, it is possible to use more clusters and hence, generate clusters with smaller size and higher similarities. The choice of higher or lower number of clusters can be decided based on how flexible our analysis can be. Lower number of groups results in less similarity and at the same time more candidates to examine. High number of clusters, on the other hand, means a more limited search space but a higher chance of finding good repurposing candidates. Based on the results of our experiments in different levels of flexibility, some of the related diseases and sets with repositioning possibilities are shown in Tables A.1 and A.2.

Group	Diseases
1	Antiphospholipid antibody syndrome; Chronic obstructive pulmonary disease; Cluster headaches; Decreased appetite; Diabetes type II; Discoid eczema; Dyshidrotic eczema; Emphysema; Migraine; Perennial allergy; Sjogrens syndrome; Support hormone balance; Ulcerative colitis; Undifferentiated connective tissue disease
2	Asthma; Eczema; Human immunodeficiency virus (HIV); Insomnia; Skin irritation

3	Bladder problems; Breakthrough intermenstrual bleeding; Breast cancer; Chronic lymphocytic leukemia; Factor V leiden; Heart palpitations; Hyperlipidemia; Inflammation; Menstrual cramps; Muscle cramps; Neuromyelitis optica; Pain in legs; Peripheral arterial disease; Pneumonia; Prevent migraine; Restless feeling in legs; Rheumatoid arthritis; Support digestive health; Support heart health
4	Bipolar I disorder; Bipolar II disorder; Chronic headache disorder; Cluster headaches; Difficulty staying asleep; Elevated blood sugar (hyperglycemia); Emotional lability; Essential tremor; Improve communication; Migraine with aura; Mood swings; Panic attacks; Persistent depressive disorder (dysthymia); Post-traumatic stress disorder; Promote relaxation; Ringing in ears (tinnitus); Schizoaffective disorder; Seizures; Social anxiety disorder; Support weight loss; Tremor(s)
5	Polycystic ovary syndrome; Eczema; Irregular menstrual periods; Breathing difficulty with activity; Asthma; Ulcerative colitis; Psoriasis; Lichen sclerosus; Cluster headaches; Discoid eczema; Sjogrens syndrome; Chronic obstructive pulmonary disease; Perennial allergy; Antiphospholipid antibody syndrome; Support hormone balance; Psoriasis on head; Depressed mood; Dyshidrotic eczema; Emphysema; Crohns disease
6	Parkinsons disease; Stiffness/Spasticity; Migraine; Nerve pain (Neuralgia)
7	Complex partial seizures; Stiffness/Spasticity; Epilepsy; Migraine
8	Hypersalivation; Drooling; Irritable Bowel Syndrome
9	Emotional lability; Difficulty staying asleep ; Essential tremor; Hyperglycemia; Chronic headache disorder; Post traumatic stress disorder; Mood swings; Bipolar I disorder; Cluster headache; Ttinnitus; Bipolar II disorder; Seizures; Social anxiety disorder; Migraine with aura; Dysthymia; Panic attacks; Schizoaffective disorder;
10	Birth control; Cold sensation in feet; Acid reflux; Heart attack (myocardial infarction); Pain in ankles; Polycystic ovary syndrome; Diabetes type II; Elevated blood sugar (hyperglycemia); Diabetes type I; Migraine; Prinzmetal angina; Angina pain; Migraine
11	High blood pressure (hypertension); Tachycardia (fast heart rate); Irregular heartbeat (cardiac arrhythmia); Atrial fibrillation; Heart palpitations

Table A.1: List of possibly associated diseases clustered through the proposed approach.

Group	Drugs
1	Liraglutide; Naltrexone; Saxagliptin metformin; Warfarin; Sitagliptin; Insulin detemir; Gliclazide; Clobetasone topical; Sitagliptin metformin; Empagliflozin; Exenatide; Estradiol norethindrone; Formoterol; Glyburide; Ketorolac; Cetirizine; Acarbose; Oxycodone; Glimepiride; Nabilone; Glipizide ; Indomethacin; Hydroxychloroquine; Esterified estrogen methyltestosterone; Ipratropium
2	Darifenacin; Nitrofurantoin; Hydromorphone; Orphenadrine aspirin caffeine; Metformin; Azithromycin; Lubiprostone; Nadolol; Fluticasone nasal spray; Mometasone topical; Hydrocodone; Isradipine; Clindamycin; Triamterene hydrochlorothiazide; Morphine; Glycopyrrolate; Chlordiazepoxide clidinium; Amoxicillin; Metoprolol succinate er; Amoxicillin clavulanate; Lactulose; Polyethylene glycol 3350 with electrolytes; Beclomethasone nasal; Docusate senna; Albuterol; Candesartan; Tolterodine; Aspirin ; Atenolol; Cephalexin; Oxycodone; Doxycycline; Potassium citrate; Sulfamethoxazole trimethoprim cotrimoxazole; Meloxicam; Terazosin; Benazepril; Montelukast; Levofloxacin; Penicillin; Azelastine
3	Ethinyl estradiol norgestimate ; Haelan tape; Levalbuterol; Ethinyl estradiol norethindrone; Dimethyl fumarate; Tiotropium inhaler; Mometasone topical; Ethinyl estradiol norgestrel; Spironolactone; Clobetasol topical; Desipramine; Ethinyl estradiol etonogestrel; Halobetasol topical; Ketoconazole topical;
4	Oxaprozin; Methadone; Warfarin; Imipramine; Fulvestrant; Beta blockers; Paclitaxel; Topiramate; Sumatriptan subcutaneous; Cisplatin; Mycophenolate mofetil; Nabumetone; Potassium chloride; Simvastatin; Vinorelbine; Domperidone; Leflunomide; Ketorolac; Fludarabine; Tamoxifen; Anastrozole; Mexiletine; Clopidogrel; Atorvastatin; Letrozole; Diclofenac misoprostol; Piroxicam; Butalbital acetaminophen caffeine; Rituximab; Methotrexate; Medroxyprogesterone acetate; Hydroxychloroquine; Capecitabine; Clozapine; Raloxifene; Zolpidem; Dapsone

5	Escitalopram; Hydromorphone; Nabiximols; Oxcarbazepine; Metaxalone; Apomorphine; Entacapone; Chlorzoxazone; Trihexyphenidyl; Zonisamide; Rotigotine; Verapamil; Quinine; Nabumetone; Dantrolene; Amantadine; Rasagiline; Desvenlafaxine; Pramipexole; Levodopa benserazide; Cabergoline ; Oxycodone; Meloxicam; Orphenadrine; Safinamide; Levetiracetam; Ropinirole; Methocarbamol; Pergolide
6	Baclofen intrathecal pump; Codeine guaifenesin syrup; Valacyclovir; Propoxyphene acetaminophen; Gabapentin; Fluticasone salmeterol; Omeprazole sodium bicarbonate; Hydrocodone
7	Lithium carbonate; Phenytoin; Nabilone; Topiramate; Primidone; Temazepam; Rizatriptan; Sumatriptan subcutaneous; Clobazam; Buspirone; Cariprazine; Dextromethorphan quinidine; Reboxetine; Acetaminophen isometheptene Dichloralphenazone
8	Hydroxyzine; Escitalopram; Duloxetine; Propranolol; Tramadol; Brexpiprazole; Fluoxetine; Lorazepam; Morphine; Valproic-acid; Chlorpromazine; Carbamazepine; Dronabinol; Clonazepam; Cyclobenzaprine; Sertraline; Pregabalin; Oxazepam; Codeine acetaminophen paracetamol; Gabapentin; Amitriptyline; Diazepam; Methylprednisolone; Alprazolam; Carisoprodol; Nortriptyline; Aluminum chloride hexahydrate topical; Levetiracetam; Botulinum toxin type A; Tizanidine; Venlafaxine; Citalopram
9	Fentanyl patch; Methylprednisolone; Naltrexone; Hydrocodone acetaminophen; Diclofenac topical; Celecoxib;Tramadol; Botulinum toxin type A; Gabapentin;Naproxen; Tizanidine; Ibuprofen-topical; Cyclobenzaprine; Steroid IV; Amitriptyline; Oxycodone acetaminophen
10	Nitroglycerin topical; Dexlansoprazole; Insulin nph; Esomeprazole; Insulin regular; Insulin glulisine; Pantoprazole; Metformin; Carvedilo ; Tramadol acetaminophen; Valproic acid; Nateglinide; Insulin aspart; Rabeprazole; Insulin glargine; Insulin lispro insulin lispro protamine; Diclofenac topical; Isosorbide mononitrate; Ketorolac; Cimetidine; Bupropion; Pioglitazone; Diltiazem; Butalbital acetaminophen caffeine; Lansoprazole; Medroxyprogesterone acetate; Ranitidine; Amlodipine pioglitazone; diltiazem; butalbital acetaminophen caffeine; lansoprazole; medroxyprogesterone acetate; ranitidine; amlodipine

11	Acetazolamide; Hydroxyzine; Cefdinir; Escitalopram; Naltrexone; Duloxetine; Morphine; Amoxicillin clavulanate; Lamotrigine; Clonazepam; Cyclobenzaprine; Sertraline; Aripiprazole; Codeine acetaminophen paracetamol; Gabapentin; Aspirin; Bupropion; Methylprednisolone; Alprazolam; Carisoprodol; Amphetamine; Dalfampridine; Botulinum toxin type A; Tizanidine; Cortisone injection; Zolpidem; Oxycodone acetaminophen

Table A.2: List of the drugs with repositioning possibility found through our approach.