ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Higher-order Drug Interactions with Hypergraph Neural Networks

Bachelor Thesis

Jonathan Maillefaud

January 19, 2025

Advisors: Dr. Nabil Abubaker, Prof. Dr. Torsten Hoefler
Scalable Parallel Computing Lab (SPCL)
Department of Computer Science, ETH Zürich

Abstract

Understanding drug interactions is crucial for ensuring patient safety and treatment efficacy, especially as polypharmacy (taking multiple medications simultaneously) becomes increasingly common. While existing methods typically analyze drug interactions in pairs, real-world scenarios often involve complex, multi-drug interactions that can lead to unexpected and sometimes harmful effects. Addressing this challenge requires advanced models that can capture the intricate relationships among multiple drugs, motivating the need for innovative approaches like hypergraph neural networks (HgNNs) to improve our understanding and prediction of these interactions. While HgNNs have been utilized in the past for the task of improving pairwise Drug-Drug interactions, to our knowledge none of the existing works consider N-way interactions.

Contents

Chapter 1

Introduction

A globally aging population and increasingly complex medical treatments have led to a rapid increase in polypharmacy usage in recent years [63]. Polypharmacy presents significant advantages over single-drug therapies with the potential for increased potency [34] and slower acquisition of drug resistance [79]. For example, a regimen for drug-sensitive tuberculosis involves taking four different drugs for two months, then taking a two-drug combination for four months [29]. Therefore, identification and analysis of treatments that involve multiple drugs is needed and essential to prevent non-optimized drug therapies (NOMT) [77].

NOMTs are a leading cause of disease, serious injury, and death and cost an estimated $528.4 billion yearly in the United States [45]. NOMTs may include adverse drug reactions (ADRs), often caused by interactions between simultaneous drug regimens. For example, patients with multiple chronic diseases can take up to 30 different medications prescribed by different physicians [77]. Uncoordinated polypharmacy can lead to ADRs and ultimately to NOMTs. Pharmacovigilance can help detect and prevent ADRs through clinical monitoring. However, these *in vitro* processes require the patient to be exposed to drug regimens for specific periods of time to allow detection of ADRs [55]. In comparison, *in silico* methods allow efficient computational screening of drug combinations and scale with combinatorial explosion when studying higher-order drug combinations [71].

We define two types of interactions in higher order drug combinations [22]:

- *Synergistic* drug interactions occur when the combined use of the drugs induces a better treatment outcome than the sum of the expected efficiencies of the individual drugs. Synergistic drug interactions are beneficial as they lower the required toxicity to the patient when taking the drug regimen.
- *Antagonist* drug interactions occur when some of the drugs counteract the effect of other drugs in the drug regimen, leading to a worse treatment outcome. Antagonist drug interactions are mostly avoided in clinical settings.

It should be noted that synergistic and antagonistic drug combinations may be useful in the treatment context. Several studies [79][73] have shown that a higher-order synergistic drug combination with subsets of antagonistic drug combinations may lead to a more potent drug regimen as it leads to a more complex fitness landscape and lower evolutionary drug resistance.

Multiple works have tried to identify higher-order drug interactions (HODIs) and subsequently predict them from order-2 drug interactions through theoretical approaches [65][26][53][31][51]. However, lack of a clear definition of the boundaries between the interaction (synergy and antagonism) and no interactions (additive effect of taking multiple drugs) has led to vast differences in the conclusions; with works concluding HODIs are common[43][33] and works concluding HODIs are extremely rare [26]. Notably, several works find a larger prevalence of antagonistic HODIs than synergistic HODIs [43][38].

In this work, we consider the semi-supervised learning problem of predicting HODIs. To the best of our knowledge, there is no work considering the problem of predicting HODIs as a semisupervised learning problem. To be precise, we study HODI prediction in a transductive setting (predict HODIs among known drugs) and in an inductive setting (predict HODIs among known and unknown drugs).

Specifically, we formulate our HODI prediction problem as a hypergraph edge prediction task. Hypergraphs are a generalization of graphs in which an edge can join any number of nodes. They are thus suitable for modeling higher-order interactions among elements. Our methods assume that drugs with similar substructures in their molecule are likely to interact with the same drugs [25][9][12]. The standard way to encode the structure of a molecule is through its SMILES string, a standardized specification, and a basic feature in drug learning. Because SMILES are publicly available for a large amount of drugs, we decided to use it as our drug feature, allowing our methods to be widely applicable, especially on newly synthesized drugs.

The main contributions of this work are summarized as follows:

- *Assembling a dataset*: We assembled a dataset of higher-order drug interactions from multiple sources. Higher-order drug interactions are stored as triplets with their corresponding DrugBank ID. In addition, we processed all of the drugs to extract different fingerprints from their molecular structure.
- *Transductive and inductive prediction*: To predict the HODIs, we use two different methods. In the transductive case, we use tensor decomposition with Laplacian regularization to complete the adjacency tensor of the HODI hypergraph. In the inductive case, we use an encoder-decoder pair of hypergraph neural networks. The encoder is based on a hypergraph convolutional network with a multi-head attention mechanism. Both methods solely use the SMILES string of the drug and are therefore applicable to any drug, without requiring other information.
- *Extensive training and experiments*: We perform a detailed hyperparameter optimization on both models and conduct extensive comparisons between the two

models. In addition, we show that our model can find new HODIs in the transductive and inductive cases.

The next chapter briefly reviews related work on HODIs and DDI predictions. Chapter III describes tensor decomposition with Laplacian regularization and attention-based hypergraph neural network method for HODI prediction. In chapter IV, we show experimental results and provide a detailed discussion comparing the two methods. The conclusion is given in Chapter V.

Chapter 2

Related Work

2.1 DDI prediction with Matrix Factorization

When considering drug-drug interactions (DDIs), the hypergraph edge prediction problem is reduced to a graph edge prediction problem. In this case, matrix factorization algorithms can be used to compute a low-rank decomposition and reconstruct the adjacency matrix of the graph. Standard algorithms include singular value decomposition (SVD) [24], non-negative matrix factorization (NMF) [32] and probabilistic matrix factorization (PMF) [41].

The approaches differ in the way drug data is included in the matrix decomposition algorithm [71]. One of the first approaches by Vilar et al. [28] used drug interaction profile fingerprinting (IPF) to create a second similarity matrix to compute the probability of DDI. Zhang et al. [48] added manifold regularization to their matrix factorization algorithm, ensuring that drug embeddings lie in a smooth manifold. Yu et al. [47], Shi et al. [42], and Shi et al. [54] used semi-nonnegative matrix factorization and triple matrix factorization to predict DDIs and their type (synergistic, antagonist). Rohani et al. [57] integrated multiple similarity matrices into their matrix decomposition procedure. Loyd et al. [87] used tensor factorization methods for knowledge graph embeddings to predict the type of ADR caused by the DDI.

2.2 DDI prediction with (Hyper)Graph Neural Networks

Another approach for edge prediction in graphs is to directly leverage the graph structure through graph convolution networks (GCN) to learn the node embeddings and then combine those node embeddings to compute the edge probabilities. Standard GCNs for node embedding include graph auto-encoders (GAEs) [30], graph sample and aggregate (GraphSAGE) [35] and graph attention networks (GATs) [44].

In addition to their abstractive capabilities as deep learning methods, GCN-based methods allow computation on the DDI graph and on the molecular graph of the drug;

this flexibility has made them the method of choice for DDI prediction. An exhaustive list of all GCN-based DDI prediction models would be outside the scope of this work, so we refer readers to the following reviews: [71], [78], [89], and [86]. One of the first models to use GCNs was GAE-based Decagon [49], based on a GCN encoder and a matrix factorization decoder to predict the type of ADR caused by the DDI. GCNMK [66] predicts DDIs and their type (synergistic or antagonistic). MIRACLE [66] used both the molecular graph and the DDI graph for the DDI predictions. 3DGT-DDI [72] used the 3D structure of the molecule encoded in the molecular graph. HyGNN [80] used hypergraphs with attention in the encoder to represent the molecular graphs and used the DDI graph in the readout decoder. HUF-DCIE [81] created a knowledge hypergraph to include other data sources in their prediction, such as drug synergy.

To the best of our knowledge, no previous work has tried to predict HODIs (synergistic and antagonistic) of order 3 or higher, especially using tensor factorization methods or hypergraph neural networks.

Chapter 3

Presentation of the approaches and algorithms

We start by presenting the general framework in which both methods are based. We then present the tensor decomposition with Laplacian regularization method for the transductive case. Finally, we present the hypergraph convolution network method for the inductive case.

3.1 HODI Hypergraph

Let $V \subset \mathbb{N}$ be a finite set of objects. Let E be a family of subsets $e \subseteq V$ with $|E| \geq 2$. Let $|V| = n$ and $|E| = m$. We call $G_H = (V, E)$ a *hypergraph* with *vertex* set V and *hyperedge* set E. We call G_H *k-uniform* if every hyperedge contains precisely k vertices. A 2-uniform hypergraph reduces to a graph. A hypergraph G_H is connected if there is a path between every pair of vertices, all hypergraphs in what follows are assumed to be connected.

A hypergraph G_H can be represented by a *incidence* matrix $H \in \{0,1\}^{n \times n}$ where $H_{v,e} = 1$ if and only if $v \in e$. We can then define the *degree* of a vertex v, $d(v) = \sum_{e \in E} H_{v,e}$, and the *degree* of a hyperedge e, $d(e) = \sum_{v \in V} H_{v,e}$. Notice that this corresponds to the expansion of the hypergraph to a bipertite graph, also called the star expansion [74]. Zhou et al. [13] introduced semi-supervised learning with the incidence matrix.

A k-uniform hypergraph G_H can also be represented by a k-order symmetric n-dimensional *adjacency* tensor $\mathcal{H} = (h_{i_1 i_2 \ldots i_k})$, where $h_{i_1 i_2 \ldots i_k} = \frac{1}{(k-1)!}$ if and only if $\{i_1, i_2, \ldots, i_k\} \in E$. Notice that $\mathcal{H}$ is very sparse as $h_{i_1 i_2 \ldots i_k} \neq 0$ only if all indices are different. The adjacency tensor is the starting point for the study of spectral properties of hypergraphs [36].

We define the higher-order drug interaction (HODI) hypergraph as the hypergraph in which the vertices are drugs and a hyperedge represents an interaction (synergistic and antagonistic) between drugs. The problem this work addresses is the semi-supervised

learning problem of classifying possible drug combinations as interacting or not. In the HODI hypergraph, this problem can be formulated as predicting new hyperedges. Both the incidence matrix and the adjacency tensor can be used to predict new hyperedges in a hypergraph [84][75][62][58].

For clarity of exposition of the following methods and due to the limitations of our dataset, we present the case where the HODI hypergraph is a 3-uniform graph. It is important to mention that all the presented methods can also be applied to the k-uniform case.

3.2 Tensor Decomposition with Laplacian Regularization (TDLR)

Let $\mathcal{H} \in \mathbb{R}^{n \times n \times n}$ be the order-3 symmetric n-dimensional adjacency tensor of our HODI hypergraph. Following [1], [2] and [3], we use the Tucker decomposition to decompose A as a core tensor $\mathcal{C}$ and a set of factor matrices $U^{(1)}, U^{(2)}, U^{(3)}$:

$$\mathcal{H} = \mathcal{C} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}$$

where $\mathcal{C} \in \mathbb{R}^{r \times r \times r}$, $U^{(1)}, U^{(2)}, U^{(3)} \in \mathbb{R}^{r \times n}$ and $\times_i$ denotes the contraction along the i-th mode:

$$(\mathcal{C} \times_1 U^{(1)})(i_1, j_2, j_3) = \sum_{j_1=1}^{r} \mathcal{C}(j_1, j_2, j_3) U^{(1)}(j_1, i_1)$$

$$(\mathcal{C} \times_2 U^{(2)})(j_1, i_2, j_3) = \sum_{j_2=1}^{r} \mathcal{C}(j_1, j_2, j_3) U^{(2)}(j_2, i_2)$$

$$(\mathcal{C} \times_3 U^{(3)})(j_1, j_2, i_3) = \sum_{j_3=1}^{r} \mathcal{C}(j_1, j_2, j_3) U^{(3)}(j_3, i_3)$$

3.2.1 Loss function

This rank decomposition r (r is a hyperparameter) can be interpreted as a form of multilinear principal component analysis (MPCA) [7] or higher-order singular value decomposition (HOSVD) [8], where the columns of the factor matrices are lower-dimensional (we assume $r \ll n$) latent feature vectors.

We optimize our decomposition by minimizing the squared Frobenius norm,

$$\mathscr{L}_{dec} = \frac{1}{2} \| \mathcal{H} - \mathcal{C} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)} \|_F^2$$

To regularize the ambient space, we add a Tikhonov regularization parameter, with hyperparameter λ,

$$\mathscr{L}_{tik} = \frac{\lambda}{2} \left(\|\mathcal{C}\|_F^2 + \sum_{i=1}^{3} \|U^{(i)}\|_F^2 \right)$$

Inspired by [11], [19] and [48], we assume that our drug fingerprints form a low-dimensional nonlinear latent manifold $M \subset R^{2048}$, also called intrinsic space. A manifold M is a topological space that locally resembles Euclidian space. We want to smooth out the row vectors of our factor matrices with respect to the gradient ∇_M on the manifold. To be precise, using the fact that the Laplacian L of the point cloud generated by our fingerprints approximates the Laplace-Beltrami operator Δ_M on the manifold[10], we minimize an approximation of the norm in the intrinsic space, with hyperparameter μ,

$$\mathscr{L}_{lap} = \frac{\mu}{2} \sum_{i=1}^{3} \mathrm{Tr}({U^{(i)}}^T L U^{(i)})$$

To prevent the Laplacian of the point cloud from introducing too much noise, we approximate it using the Laplacian of the k-nearest-neighbors graph (k is another hyperparameter) on our fingerprint point cloud [61]. We use different distance metrics on our fingerprints to compute the nearest-neighbors graph.

- Tanimoto/Jaccard distance: measures the ratio of the intersection to the union of the two fingerprints. Specifically,

$$\mathrm{Tanimoto}(A, B) = 1 - \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

- Dice similarity: similar to Tanimoto but emphasizes common features by doubling the intersection term. Specifically,

$$\mathrm{Dice}(A, B) = 1 - \frac{2|A \cap B|}{|A| + |B|}$$

- Hamming distance: counts the number of differing bits between two fingerprints. Specifically,

$$\mathrm{Hamming}(A, B) = \sum_{i=1}^{n} |A_i - B_i|$$

We optimize the complete loss function simultaneously,

$$\underset{\mathcal{C}, U^{(1)}, U^{(2)}, U^{(3)}}{\arg\min} \mathscr{L} = \underset{\mathcal{C}, U^{(1)}, U^{(2)}, U^{(3)}}{\arg\min} \mathscr{L}_{dec} + \mathscr{L}_{tik} + \mathscr{L}_{lap}$$

3.2.2 Optimization procedure

We study three different initialization algorithms for our decomposition, the HOSVD decomposition computed through higher-order orthogonal iteration [8], the non-negative Tucker decomposition through higher-order orthogonal iteration [37] and the non-negative Tucker decomposition through hierarchical alternating least squares [15]. We then minimize the loss $\mathscr{L}$ using an alternating least squares procedure on the

columns of our factor matrices. The loss is individually minimized on every column with the Newton–Raphson method. We compute the graph adjacency matrix of the nearest-neighbor fingerprint point cloud with the NN-Descent algorithm [20].

We can rewrite our loss function,

$$\begin{aligned}\underset{\mathcal{C},U^{(1)},U^{(2)},U^{(3)}}{\arg\min}\ & \frac{1}{2}\sum_{i,j,k=1}^{n}\left(\mathcal{H}_{ijk}-\mathcal{H}^{pred}_{ijk}\right)^2 \\ & +\frac{\lambda}{2}\left(\|\mathcal{C}\|_F^2+\sum_{i=1}^{3}\sum_{i=1}^{n}\|u_i^{(i)}\|_2^2\right) \\ & +\frac{\mu}{2}\sum_{i=1}^{3}\sum_{j,k=1}^{n}\|u_j^{(i)}-u_k^{(i)}\|_2^2 W_{jk}\end{aligned}$$

where $u_i^{(j)}$ is the i-th column of factor j, W_{jk} is the similarity between fingerprint j and fingerprint k and $\mathcal{H}^{pred}$ is the tensor predicting the new hyperedges,

$$\mathcal{H}^{pred}=\mathcal{C}\times_1 U^{(1)}\times_2 U^{(2)}\times_3 U^{(3)}$$

Then, the first derivatives are

$$\begin{aligned}\frac{\delta}{\delta u_i^{(1)}}\mathscr{L} &= \sum_{j,k=1}^{n}\left(\mathcal{H}^{pred}_{ijk}-\mathcal{H}_{ijk}\right)M^{(1)}_{jk}+\lambda u_i^{(1)} \\ &\quad +\mu\left(\sum_{p=1}^{n}\left(u_i^{(1)}-u_p^{(1)}\right)W_{ip}+\sum_{p=1}^{n}\left(u_p^{(1)}-u_i^{(1)}\right)W_{pi}\right) \\ &= \left(\sum_{j,k=1}^{n}M^{(1)}_{jk}{M^{(1)}_{jk}}^T+\lambda I+\mu\left(\sum_{p=1}^{n}W_{ip}+W_{pi}\right)I\right)u_i^{(1)} \\ &\quad -\sum_{j,k=1}^{n}\mathcal{H}_{i,j,k}M^{(1)}_{jk}-\mu\sum_{p=1}^{n}\left(W_{ip}+W_{pi}\right)u_p^{(1)}\end{aligned}$$

where $M^{(1)}_{jk}=\mathcal{C}\times_2 u_j^{(2)}\times_3 u_k^{(3)}$

$$\begin{aligned}\frac{\delta}{\delta u_j^{(2)}}\mathscr{L} &= \sum_{i,k=1}^{n}\left(\mathcal{H}^{pred}_{ijk}-\mathcal{H}_{ijk}\right)M^{(2)}_{ik}+\lambda u_j^{(2)} \\ &\quad +\mu\left(\sum_{p=1}^{n}\left(u_j^{(2)}-u_p^{(2)}\right)W_{jp}+\sum_{p=1}^{n}\left(u_p^{(2)}-u_j^{(2)}\right)W_{pj}\right) \\ &= \left(\sum_{i,k=1}^{n}M^{(2)}_{ik}{M^{(2)}_{ik}}^T+\lambda I+\mu\left(\sum_{p=1}^{n}W_{jp}+W_{pj}\right)I\right)u_j^{(2)} \\ &\quad -\sum_{i,k=1}^{n}\mathcal{H}_{ijk}M^{(2)}_{ik}-\mu\sum_{p=1}^{n}\left(W_{jp}+W_{pj}\right)u_p^{(2)}\end{aligned}$$

where $M_{ik}^{(2)} = \mathcal{C} \times_1 u_i^{(1)} \times_3 u_k^{(3)}$

$$\begin{aligned}\frac{\delta}{\delta u_k^{(3)}}\mathscr{L} &= \sum_{i,j=1}^{n}\left(\mathcal{H}_{ijk}^{pred} - \mathcal{H}_{ijk}\right) M_{ij}^{(3)} + \lambda u_k^{(3)} \\ &\quad + \mu\left(\sum_{p=1}^{n}\left(u_k^{(3)} - u_p^{(3)}\right) W_{kp} + \sum_{p=1}^{n}\left(u_p^{(3)} - u_k^{(3)}\right) W_{pk}\right) \\ &= \left(\sum_{i,j=1}^{n} M_{ij}^{(3)} M_{ij}^{(3)^T} + \lambda I + \mu\left(\sum_{p=1}^{n} W_{kp} + W_{pk}\right) I\right) u_k^{(3)} \\ &\quad - \sum_{i,j=1}^{n} \mathcal{H}_{ijk} M_{ij}^{(3)} - \mu \sum_{p=1}^{n}\left(W_{kp} + W_{pk}\right) u_p^{(3)}\end{aligned}$$

where $M_{ij}^{(3)} = \mathcal{C} \times_1 u_i^{(1)} \times_2 u_j^{(2)}$

We can then compute the second derivatives,

$$\frac{\delta^2}{\delta {u_i^{(1)}}^2}\mathscr{L} = \sum_{j,k=1}^{n} M_{jk}^{(1)} M_{jk}^{(1)^T} + \lambda I + \mu\left(\sum_{p=1}^{n} W_{ip} + W_{pi}\right) I$$

where $M_{jk}^{(1)} = \mathcal{C} \times_2 u_j^{(2)} \times_3 u_k^{(3)}$

$$\frac{\delta^2}{\delta {u_j^{(2)}}^2}\mathscr{L} = \sum_{i,k=1}^{n} M_{ik}^{(2)} M_{ik}^{(2)^T} + \lambda I + \mu\left(\sum_{p=1}^{n} W_{jp} + W_{pj}\right) I$$

where $M_{ik}^{(2)} = \mathcal{C} \times_1 u_i^{(1)} \times_3 u_k^{(3)}$

$$\frac{\delta^2}{\delta {u_k^{(3)}}^2}\mathscr{L} = \sum_{i,j=1}^{n} M_{ij}^{(3)} M_{ij}^{(3)^T} + \lambda I + \mu\left(\sum_{p=1}^{n} W_{kp} + W_{pk}\right) I$$

where $M_{ij}^{(3)} = \mathcal{C} \times_1 u_i^{(1)} \times_2 u_j^{(2)}$

We compute the update rule for the Newton Raphson method as follows,

$$\left[\frac{\delta^2}{\delta {u_i^{(1)}}^2}\mathscr{L}\right] s = -\frac{\delta}{\delta u_i^{(1)}}\mathscr{L} \rightarrow {u_i^{(1)}}' = u_i^{(1)} + s$$

$$\left[\frac{\delta^2}{\delta {u_j^{(2)}}^2}\mathscr{L}\right] s = -\frac{\delta}{\delta u_j^{(2)}}\mathscr{L} \rightarrow {u_j^{(2)}}' = u_j^{(2)} + s$$

$$\left[\frac{\delta^2}{\delta {u_k^{(3)}}^2}\mathscr{L}\right] s = -\frac{\delta}{\delta u_k^{(3)}}\mathscr{L} \rightarrow {u_k^{(3)}}' = u_k^{(3)} + s$$

Notice that the second derivatives of the columns are always positive definite, and thus the update s can be efficiently computed with high numerical stability through the Cholesky decomposition. The column updates can be massively parallelized as they are all independent of each other. The sparsity of $\mathcal{H}$ and W should be leveraged appropriately through a sparse Tucker decomposition [14]. For an algorithm, see Algorithm 1.

Algorithm 1 The tensor Decomposition with Laplacian regularization algorithm

procedure MatrixDecomposition($F \in \mathbb{R}^{n \times f}, \mathcal{H} \in \mathbb{R}^{n \times n \times n}, r, k, \lambda, \mu$)
  $W \leftarrow$ NN-descent(F, k) ▷ See [20]
  $\mathcal{C}, U^{(1)}, U^{(2)}, U^{(3)} \leftarrow$ Sparse-Tucker$(\mathcal{H}, r)$ ▷ See [14]
  while $U^{(1)}, U^{(2)}, U^{(3)}$ have not converged **do**
    Parallel $u_i^{(1)} \in \mathbb{R}^r$ $(1 \leq i \leq n)$ **do**
      $M_{jk}^{(1)} \leftarrow \mathcal{C} \times_2 u_j^{(2)} \times_3 u_k^{(3)}$
      $\nabla_{(1)}^2 \leftarrow \sum_{j,k=1}^n M_{jk}^{(1)} {M_{jk}^{(1)}}^T + \lambda I + \mu \left(\sum_{p=1}^n W_{ip} + W_{pi} \right) I$
      $\nabla_{(1)} \leftarrow \sum_{j,k=1}^n \left(\mathcal{H}_{ijk}^{pred} - \mathcal{H}_{i,j,k} \right) M_{jk}^{(1)} + \lambda u_i^{(1)}$
      $+\mu \left(\sum_{p=1}^n \left(u_i^{(1)} - u_p^{(1)} \right) W_{ip} + \sum_{p=1}^n \left(u_p^{(1)} - u_i^{(1)} \right) W_{pi} \right)$
      $u_i^{(1)} \leftarrow u_i^{(1)} + \text{solve}(\nabla_1^2, \nabla_1)$
    End
    Parallel $u_j^{(2)} \in \mathbb{R}^r$ $(1 \leq i \leq n)$ **do**
      $M_{ik}^{(2)} \leftarrow \mathcal{C} \times_1 u_i^{(1)} \times_3 u_k^{(3)}$
      $\nabla_{(2)}^2 \leftarrow \sum_{j,k=1}^n M_{jk}^{(1)} {M_{jk}^{(1)}}^T + \lambda I + \mu \left(\sum_{p=1}^n W_{ip} + W_{pi} \right) I$
      $\nabla_{(2)} \leftarrow \sum_{i,k=1}^n \left(\mathcal{H}_{ijk}^{pred} - \mathcal{H}_{ijk} \right) M_{ik}^{(2)} + \lambda u_j^{(2)}$
      $+\mu \left(\sum_{p=1}^n \left(u_j^{(2)} - u_p^{(2)} \right) W_{jp} + \sum_{p=1}^n \left(u_p^{(2)} - u_j^{(2)} \right) W_{pj} \right)$
      $u_j^{(2)} \leftarrow u_j^{(2)} + \text{solve}(\nabla_2^2, \nabla_2)$
    End
    Parallel $u_k^{(3)} \in \mathbb{R}^r$ $(1 \leq i \leq n)$ **do**
      $M_{ij}^{(3)} \leftarrow \mathcal{C} \times_1 u_i^{(1)} \times_2 u_j^{(2)}$
      $\nabla_{(3)}^2 \leftarrow \sum_{i,j=1}^n M_{ij}^{(3)} {M_{ij}^{(3)}}^T + \lambda I + \mu \left(\sum_{p=1}^n W_{kp} + W_{pk} \right) I$
      $\nabla_{(3)} \leftarrow \sum_{i,j=1}^n \left(\mathcal{H}_{ijk}^{pred} - \mathcal{H}_{ijk} \right) M_{ij}^{(3)} + \lambda u_k^{(3)}$
      $+\mu \left(\sum_{p=1}^n \left(u_k^{(3)} - u_p^{(3)} \right) W_{kp} + \sum_{p=1}^n \left(u_p^{(3)} - u_k^{(3)} \right) W_{pk} \right)$
      $u_k^{(3)} \leftarrow u_k^{(3)} + \text{solve}(\nabla_3^2, \nabla_3)$
    End
  end while
end procedure

3.3 Attention-based Hypergraph Neural Network (AHNN)

As in Section 3.2, we need to capture a latent representation of the drugs to predict their HODIs. To compute the embeddings over the structure of the hypergraph, we use a pair of hypergraph convolutional neural networks:

1. *Attention convolution hyperedge encoder*: drugs are represented as hyperedges and latent representations are found through attention-based hypergraph convolutions.
2. *Hyperedge pooling decoder*: the latent representations of the drugs are aggregated to predict HODIs.

3.3.1 Hypergraph edge encoder

For the encoder to generate meaningful latent representations from the fingerprints, a new hypergraph is assembled from the drug fingerprints. Specifically, we create a hypergraph $H_{enc} = (V, E)$ where E is the set of drugs and V is the set of possible bits in our fingerprints. Then, for a drug $e \in E$ with fingerprint f^e, $v_i \in e$ if and only if $f_i^e = 1$.

For the hypergraph neural network to learn over H_{enc}, we modify the hyperedge attention convolution from [60] and [80]. Let $|V| = n$ and $|E| = m$. We create a vertex feature matrix $L^{(0)} \in \mathbb{R}^{n \times d_n^{(0)}}$ and an edge feature matrix $M^{(0)} \in \mathbb{R}^{m \times d_e^{(0)}}$. $d_n^{(0)}$ and $d_e^{(0)}$ are the initial embedding dimensions of the vertices and hyperedges, respectively.

Attention convolution consists of two steps: hyperedge-level attention and vertex-level attention. Hyperedge-level attention selectively aggregates the latent representation of the drugs and obtains the latent representation of the fingerprint bit. Vertex-level attention selectively aggregates latent representations of the fingerprint bits and obtains the latent representation of the drug. We perform attention with multiple heads to allow for different selection. This convolution operation is equivalent to a heterogeneous graph attention network [44] on the star expansion of the hypergraph [74]. The usage of attention is important as it allows the model more flexibility to choose which bit of the fingerprint is relevant to the probability of HODI.

Hyperedge-level attention

Hyperedge level attention for vertex i, layer p, head k is defined as

$$\ell_i^{(p+1)^k} = \sigma\left(\frac{1}{\sqrt{\mathcal{D}(v_i)}} \sum_{e_j \in \mathcal{N}(v_i)} H_{ij}^k \frac{1}{\sqrt{\mathcal{D}(e_j)}} m_j^{(p)} P_1^k\right)$$

where σ is a non-linear activation function like LeakyReLU, P_1^k is the linear weight matrix for head k. $\mathcal{N}(v_i)$ is the neighborhood of vertex i, that is $\mathcal{N}(v_i) = \{e_i \mid v_i \in e_i\}$

We normalize by the degrees of vertices and hyperedges to avoid the vanishing/exploding gradient problem, there $\mathcal{D}(v_i)$ is the degree of vertex v_i and $\mathcal{D}(e_j)$ is the degree of hyperedge e_j.

H^k is the attention matrix for head k, it has the same shape as the hypergraph incidence matrix. The attention score can be interpreted as a transition probability between the hyperedge e_j and the vertex v_i or as the selection of relevance between the hyperedge e_j and vertex v_i. The attention score are a probability distribution over the neighborhood computed as

$$H_{ij} = \frac{\exp(\text{sim}(\ell_i, m_j))}{\sum_{e_k \in \mathcal{N}(v_i)} \exp(\text{sim}(\ell_i, m_k))}$$

We define two types of attention similarity metrics:

- *Concatenation similarity*: $\text{sim}(\ell_i, m_j) = \sigma(A^T \left[\ell_i P_2 \| m_j P_3\right])$ where $\|$ is the concatenation operation, σ is a non-linear activation function, A is the matrix used to output the k similarity values and P_2, P_3 are linear weight matrices.
- *Dot similarity*: $\text{sim}(\ell_i, m_j) = \sigma(A^T(\ell_i P_2 \cdot m_j P_3))$ where $\cdot$ is the element-wise multiplication operation, σ is a non-linear activation function, A is the matrix used to output the k similarity values and P_2, P_3 are linear weight matrices.

Lastly, we aggregate the multiple heads through concatenation or taking the mean.

Vertex-level attention

Similarly to hyperedge-level attention, vertex level attention for hyperedge j, layer p, head k is defined as

$$m_j^{(p+1)^k} = \sigma\left(\frac{1}{\sqrt{\mathcal{D}(e_j)}} \sum_{v_i \in e_j} H_{ji}^k \frac{1}{\sqrt{\mathcal{D}(v_i)}} \ell_i^{(p+1)} P_4^k\right)$$

where P_4^k is the linear weight matrix for head k. H^k is the attention matrix for head k, it has the same shape as the transpose of the hypergraph incidence matrix.

The hypergraph edge encoder is based on there two attention steps that selectively aggregate vertex embeddings and hyperedge embeddings. We first apply hyperedge level attention and finish by applying vertex-level attention. The hyperedge embeddings then capture a latent representation of the drugs.

3.3.2 Hypergraph edge decoder

The hyperedge decoder operates directly on the HODI graph. It takes the drug embeddings generated from the encoder and pooles them to predict new edges. Specifically, for drugs with embeddings m_1, m_2 and m_3, we compute $\gamma(m_1, m_2, m_3)$. We consider two forms of pooling:

- *Dot pooling*: $\gamma(m_1, m_2, m_3) = a^T(m_1 \cdot m_2 \cdot m_3)$
- *MLP pooling*[68]: $\gamma(m_1, m_2, m_3) = \text{MLP}([m_1 \| m_2 \| m_3])$

We pass the output of $\gamma(m_1, m_2, m_3)$ through a sigmoid layer to obtain a probability between 0 and 1, and results closer to 1 indicate a higher chance of a HODI occurring.

3.3.3 Optimization procedure

As we have a small dataset, we initialize our edge feature matrix with one hot encoding (every row and column has exactly one nonzero entry). We initialize the node feature matrix with Glorot initialization [16]. We train the encoder-decoder architecture end-to-end using gradient descent with the Adam otpimizer and the binary cross entropy loss:

$$\mathscr{L} = -\frac{1}{N}\sum_{n=1}^{N} y_n \log \hat{y}_n + (1 - y_n) \log(1 - \hat{y}_n)$$

where y_n is the label of the hyperedge and $\hat{y}_n$ is the predicted probability of the model.

Since the convolutions can be computed in parallel, the complexity of the model is linear in the number of edges and nodes of the hypergraph. The handling of the sparsity of the hypergraphs can be done using standard geometric deep learning libraries [50]. Normalization can be performed efficiently using the function provided by deep learning libraries [83].

Chapter 4

Experiments & Results

In this chapter, we evaluate our proposed TDLR and AHNN. We first present our dataset and experimental setup. Then, we provide individual in-depth analysis of the models. Finally, we discuss and contrast the two approaches.

4.1 Datasets

Several works have identified HODIs through large-scale screening of administrative healthcare datasets, notably opoids [67], benzodiazepine [69], antidepressants [76], skeletal muscle relaxants [70] and anticoagulants [82]. We extracted all drug-drug-drug interactions and kept the drugs composed only of a single molecule. Drugs composed of multiple molecules, such as conjugated estrogens (DB00286), often lack a clear specification and do not have a single SMILES string. A summary of the sizes of the individual datasets and the number of unique drugs is shown in 4.1.

For all the remaining drugs, we get the DrugBank ID and the corresponding SMILES string. The DrugBank database [46] contains an extensive amount of pharmaceutical information (chemical properties, identifiers, pharmacodynamics, drug-drug interactions, side effects); we only use the SMILES string to allow our model to work with a wide variety of drugs.

Table 4.1: Summary of the sizes of the datasets

Dataset	Number of HODIs	Number of drugs
Opoids [67]	166	69
Benzodiazepine [69]	79	69
Skeletal Muscle Relaxant [70]	29	40
Antidepressants [76]	334	114
Total	608	129

Table 4.2: Grid Search parameters

(a) TDLR

Parameter	Values
λ	0.001, 0.3, 0.6, 1, 1.3 1.6, 2, 2.3, 2.6, 3
μ	0.001, 0.3, 0.6, 1, 1.3 1.6, 2, 2.3, 2.6, 3
Neighbors	40%, 45%, 50%, 55% 60%, 65%, 70%
Rank	20%, 33%, 50%

(b) Hypergraph Neural Network

Parameter	Values
Learning Rate	0.001, 0.1, 0.2 0.3, 0.4, 0.5
Hidden features	32, 64, 128
Number of heads	1, 2, 4
Dropout	0, 0.5

We further process the SMILES string by computing different fingerprints for each drug molecule with the RDKit Cheminformatics software [85]. The idea behind generating a fingerprint is to apply a kernel to the molecule structure and hash it to obtain a bit vector. Fingeprints have found wide acceptance in substructure searching [18] and computing molecule similarity [40]. We compute the following fingerprints:

- *RDKit (Topological) Fingerprints* [85][21]: a daylight-like substructure fingerprint, based on the identification of subgraphs of the molecule graph.
- *Atom-Pairs Fingerprints* [4]: a similarity fingerprint based on properties of pairs of atoms in the drug molecule and their topological distance.
- *Topological-Torsion Fingerprints* [5]: a similarity fingerprint based on the structural information of subsets of size 4 in the drug molecule.
- *Morgan Fingerprints* [17][6]: a similarity fingerprint considering the environment and properties of every atom in the drug molecule.

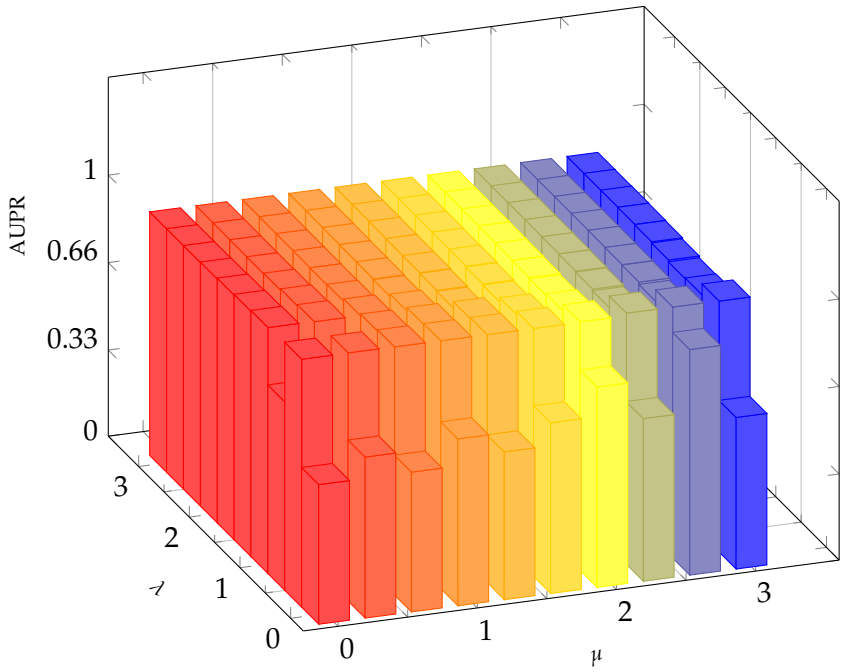
We compute a dense bit fingerprint vector of length 2048 for every drug. We keep the default parameters provided by the RDKit library.

We perform negative hyperedge sampling [88] by selecting three random drugs and adding a hyperedge containing these three drugs. We generate as many negative hyperedges as positive ones to ensure a balanced classification dataset.

4.2 Experimental Setup

Every data set is shuffled and divided into a train set (90%) and a holdout test set (10%). We obtain the optimal hyperparameters with an exhaustive grid search with 5-fold cross-validation on a validation set (10%). The parameters for the grid search can be seen in Tables 4.2a and 4.2b. In addition, we perform threshold tuning with 5-fold cross-validation on TDLR. We measure the accuracy, harmonic mean of precision and recall (F1 score), area under the receiver-operating curve (AUROC), and area under the

Figure 4.1: Impact of regularization parameters on AUPR



precision-recall curve (AUPR) metrics on the test set. Higher values always indicate a better score.

We take 65% of the neighbors to build the Laplacian of the point cloud and choose a rank of 33% of the size of the data to decompose our HODI tensor. We choose regularization hyperparameters of $\lambda = 0.66$ and $\mu = 2.66$ for regularization in the ambient space and intrinsic space, respectively.

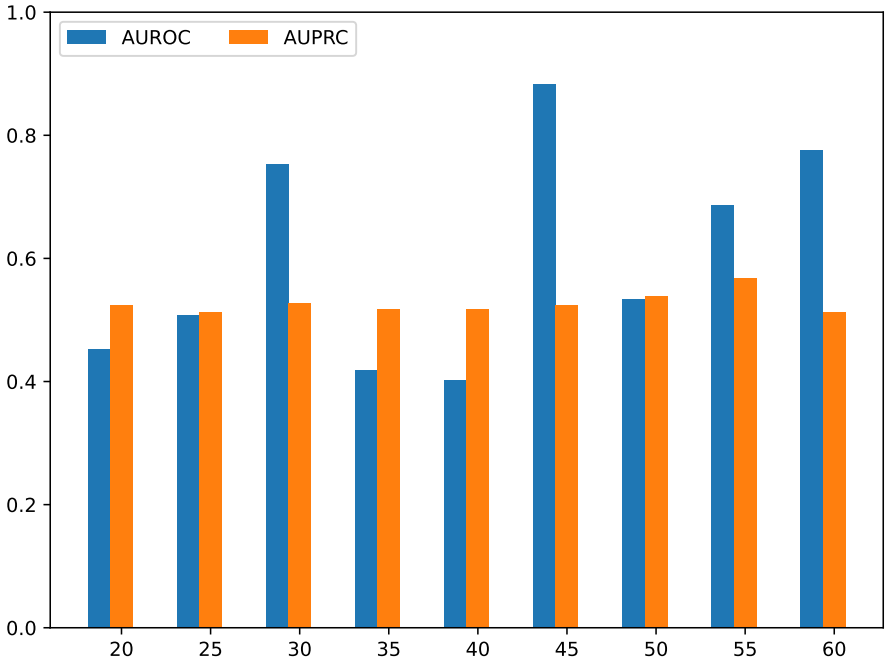
TDLR was implemented in Numpy [56][59] with the methods provided by the Tensorly library [52] and trained with an AMD EPYC 7742 processor. AHNN was implemented in PyTorch [83] with the method provided by the PyTorch Geometric library [50] and trained on an NVIDIA A100 40GB SMX GPU. Both models were integrated into the Scikit-Learn API for an equal testing ground [23][39] and trained on the facilities of the Swiss National Supercomputing Center [64].

4.3 Tensor Decomposition with Laplacian Regularization (TDLR)

4.3.1 Impact of hyperparameters

We first analyze the impact of regularization in the ambient space (hyperparameter λ) and regularization in the intrinsic space (μ). As can be seen in Figure 4.1, the hyperparameters must be carefully balanced with the optimal values of $\lambda = 0.33$ and $\mu = 2.33$. Large values of λ would crush the decomposition factors to zero and would

Figure 4.2: Score relative to the number of neighbors in point cloud Laplacian (Dice similarity, $r = 22$, $\lambda = 0.33$, $\mu = 2.33$)



lead to the loss of meaningful data. However, too small values of λ would allow for an explosion of values in Newton updates and ultimately lead to divergence. Similarly, an overwhelming intrinsic space regularization with a high μ could counteract the representation gained through the Tucker decomposition and lead to divergence.

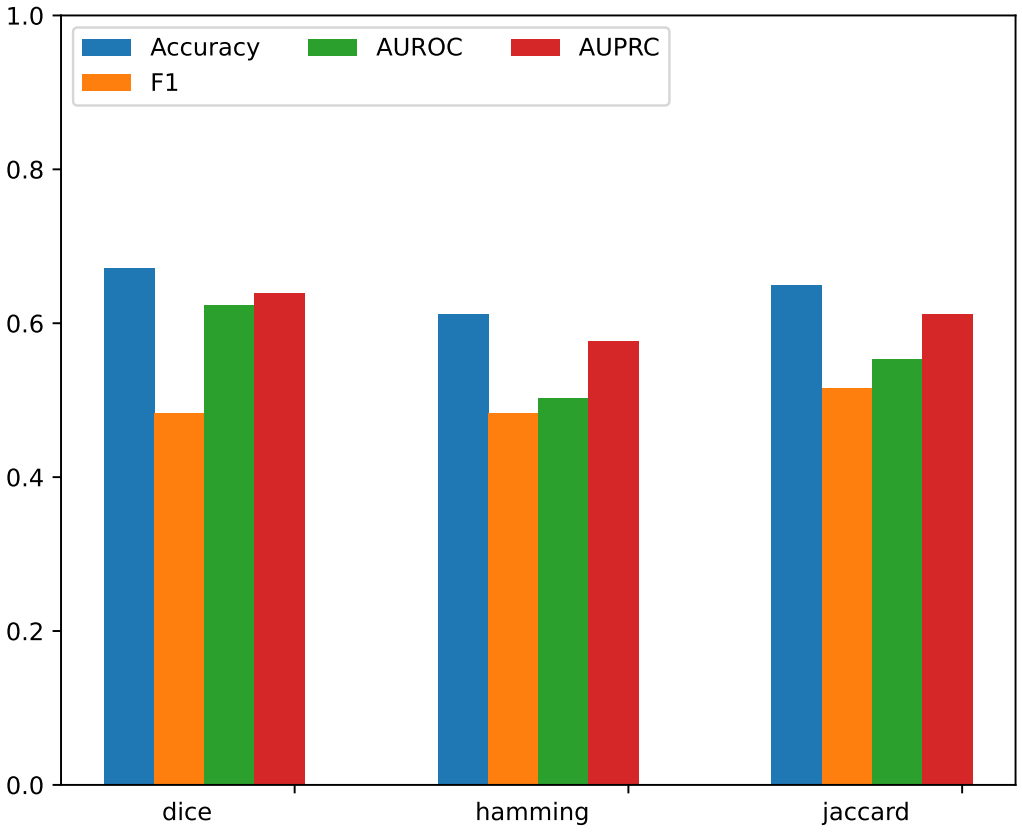
The strength of the parameters μ also depends on the number of neighbors chosen k when constructing the Laplacian point cloud for regularization in the intrinsic scape. Figure 4.2 illustrates the trade-off when choosing k. Choosing too few neighbors when building the Laplacian does not give enough data to the predictive model, whereas too many neighbors in the point-cloud Laplacian lead to a noisy environment when the learning process is hampered.

As detailed in Figure 4.3, the model can learn with any fingerprint similarity metric used. Dice similarity shows a slight improvement in accuracy compared to Jaccard similarity and Hamming similarity. Dice similarity has gained wide acceptance in fingerprint similarity measurements [27], mainly thanks to it's higher sensitivity in heterogeneous data.

4.3.2 Convergence properties of Alternating Least Squares

Newton's method was chosen as an optimization algorithm for TDLR due to its solid convergence properties, it's fast convergence speed, and numerical stability. However, it still relies on a good starting position.

Figure 4.3: Score relative to similarity metric
($k = 45$, $r = 22$, $\lambda = 0.33$, $\mu = 2.33$)



We analyze the convergence properties of our algorithm for different initialization conditions. Especially with extremely sparse data, such as the HODI tensor, the overwhelming number of zero values requires careful handling to prevent numerical instabilities. As shown in Figure 4.4, Newton's method converges very quickly and the initialization does not impact the final convergence point.

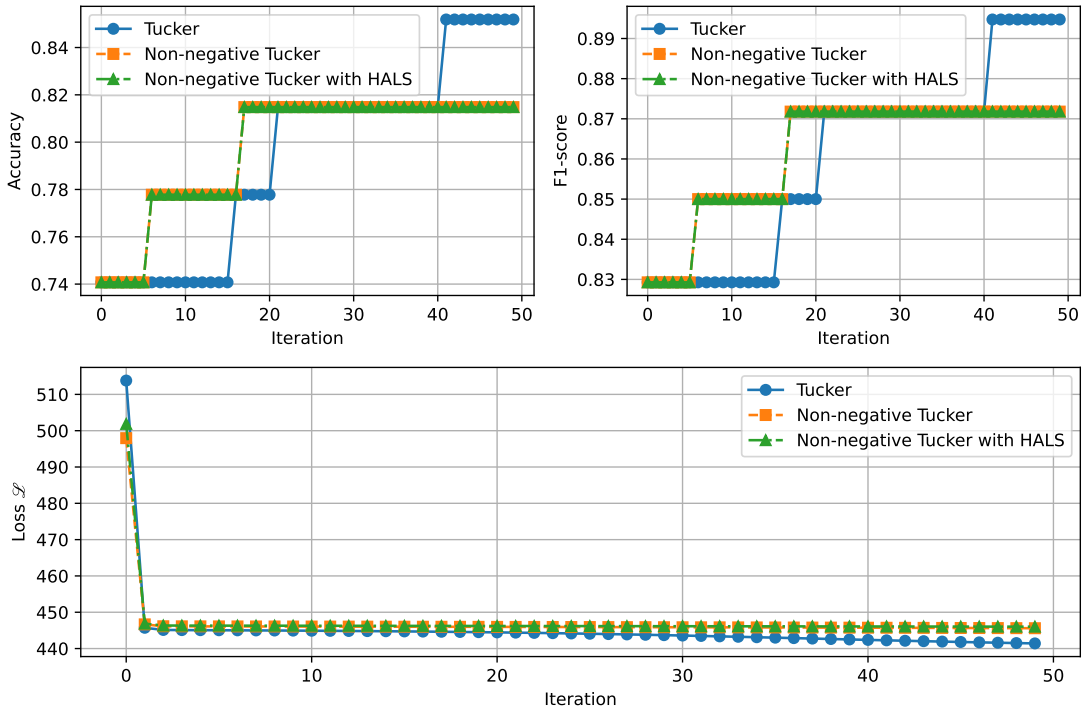
4.4 Attention-based Hypergraph Neural Network (AHNN)

4.4.1 Impact of Hyperparameters

As detailed in Figure 4.5, the decoder type used does not significantly affect the performance of the Hypergraph Neural Network. Although the MLP decoder naturally possesses more expressivity, the Dot decoder is enough to aggregate the latent embeddings obtained from the encoder. It should be noted that, while the Dot decoder offers better computational performance, the MLP decoder could learn a finer decision boundary in the case of a larger dataset.

We further investigate the effect of the attention mechanism in the encoder. As with

Figure 4.4: Loss, Accuracy and F1-score per iteration



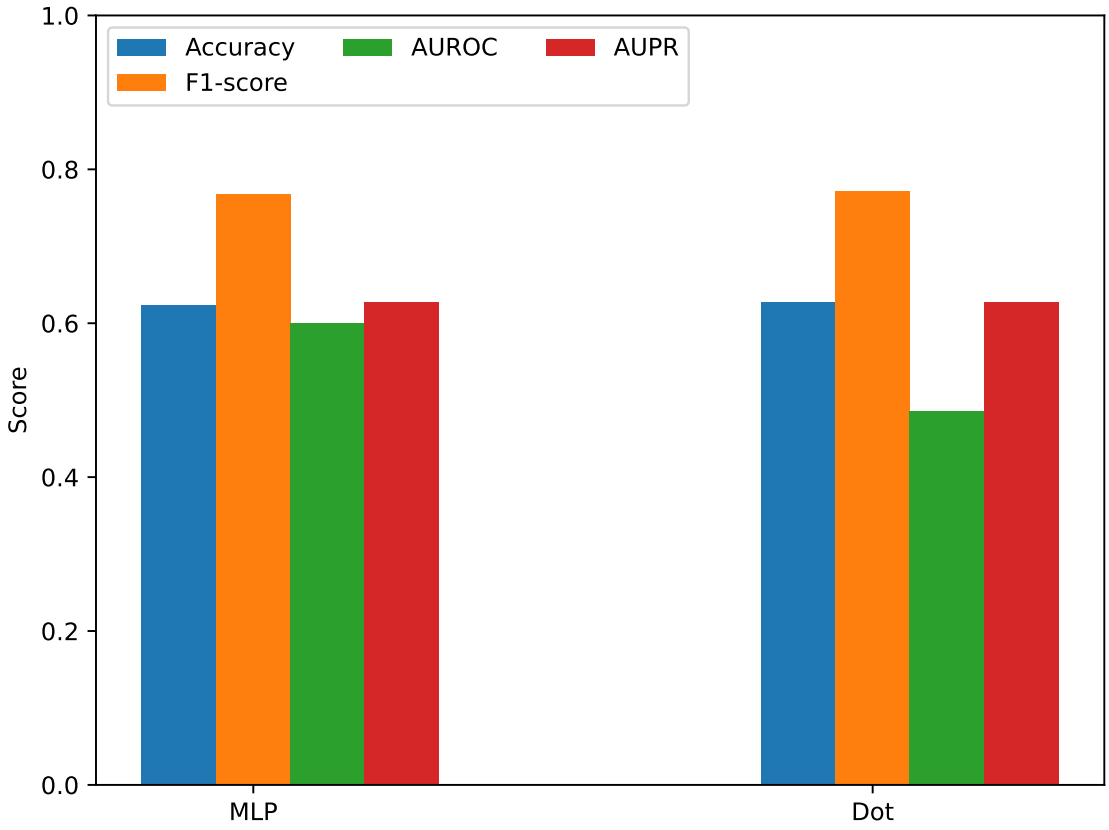
the decoder, Figure 4.6 shows that the more efficient dot attention has enough representation capacity to match concatenation-based attention. Experimentation with a larger dataset is necessary for a more conclusive answer on the relative performance of both types of attention. A slight improvement in discriminatory abilities can be noted when concatenating the attention heads instead of averaging the heads, which can be explained by the added flexibility concatenation provides.

4.5 Model Comparison

Figure 4.7 contains the evaluation of TDLR and AHNN for every fingerprint for every data set. Multiple patterns can be recognized:

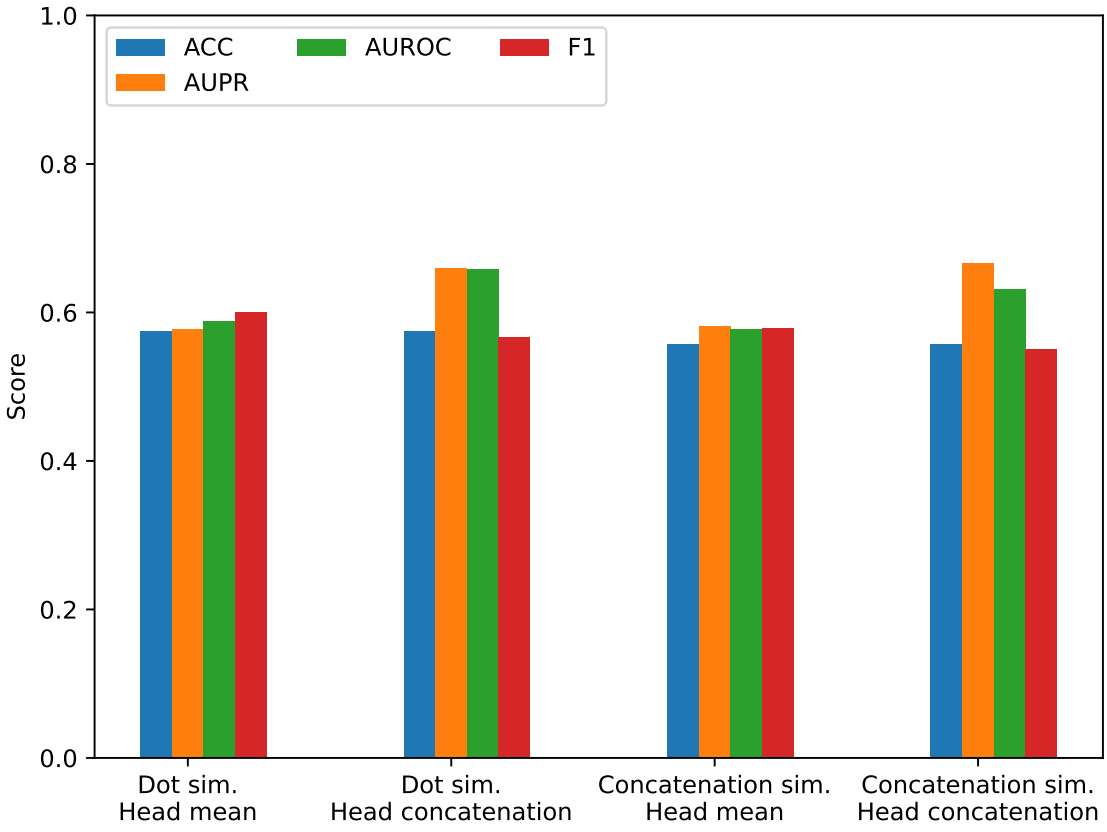
- Models have difficulty learning well with the RDKit fingerprint. As the RDKit fingerprint is constructed for substructure search and not for similarity measures, it leads to lower performance.
- Models have difficulty learning well with the Torsion fingerprint. AHNN does not have an AUPR greater than 0.6 for any dataset with the Torsion fingerprint. The low performance could indicate that the 3D information from the drug molecule is less representative for HODIs than expected.

Figure 4.5: Scores relative to Decoder type
(Embedding size of 64, learning rate of 0.01)



- The results with Atom-Pair fingerprints have high variability. AHNN has an AUPR of 0.9 with Atom-pair fingerprints on the opoids dataset and an AUPR of 0.5 with Atom-Pair fingerprints on the skeletal muscle relaxant dataset. This leads to the conclusion that different drug features are used to predict HODIs for different families of drugs.
- The high variability on the Skeletal Muscle Relaxant dataset can be explained by the very small size of the dataset (29 samples), which is too small for the models to learn from.
- Similarly to a large number of other works, the fingerprints with the best mean performance are the Morgan fingerprint and the MACCS fingerprint (AUPR consistently above 0.7).
- Although it exhibits good AUPR and AUROC performance, TDLR has very low accuracy and F1 score. This is explained by the complexity of finding a well-performing threshold for a variety of datasets, as the sparsity induced in the HODI tensor by larger datasets leads to decision values with smaller. Finding a thresholding algorithm relative to the dataset warrants further investigation.

Figure 4.6: Score relative to Attention mechanism type
(Embedding size of 64, 4 attention heads, learning rate of 0.001)



- Overall, TDLR exhibits better performance than AHNN (consistently higher AUPR and AUROC values). As deep neural networks usually require larger datasets to learn, a definitive conclusion should only be drawn with both models trained on large datasets. Our current dataset contains only 0. 02% of all possible interactions with the 129 drugs.
- Lastly, the performance of the models on the combined datasets is lower than on the datasets individually, illustrating the complexity of learning to predict HODIs over different drug families.

Figure 4.7: Performance per model per dataset per fingerprint type
Blue: accuracy, Green: AUPR, Orange: AUROC, Red: F1

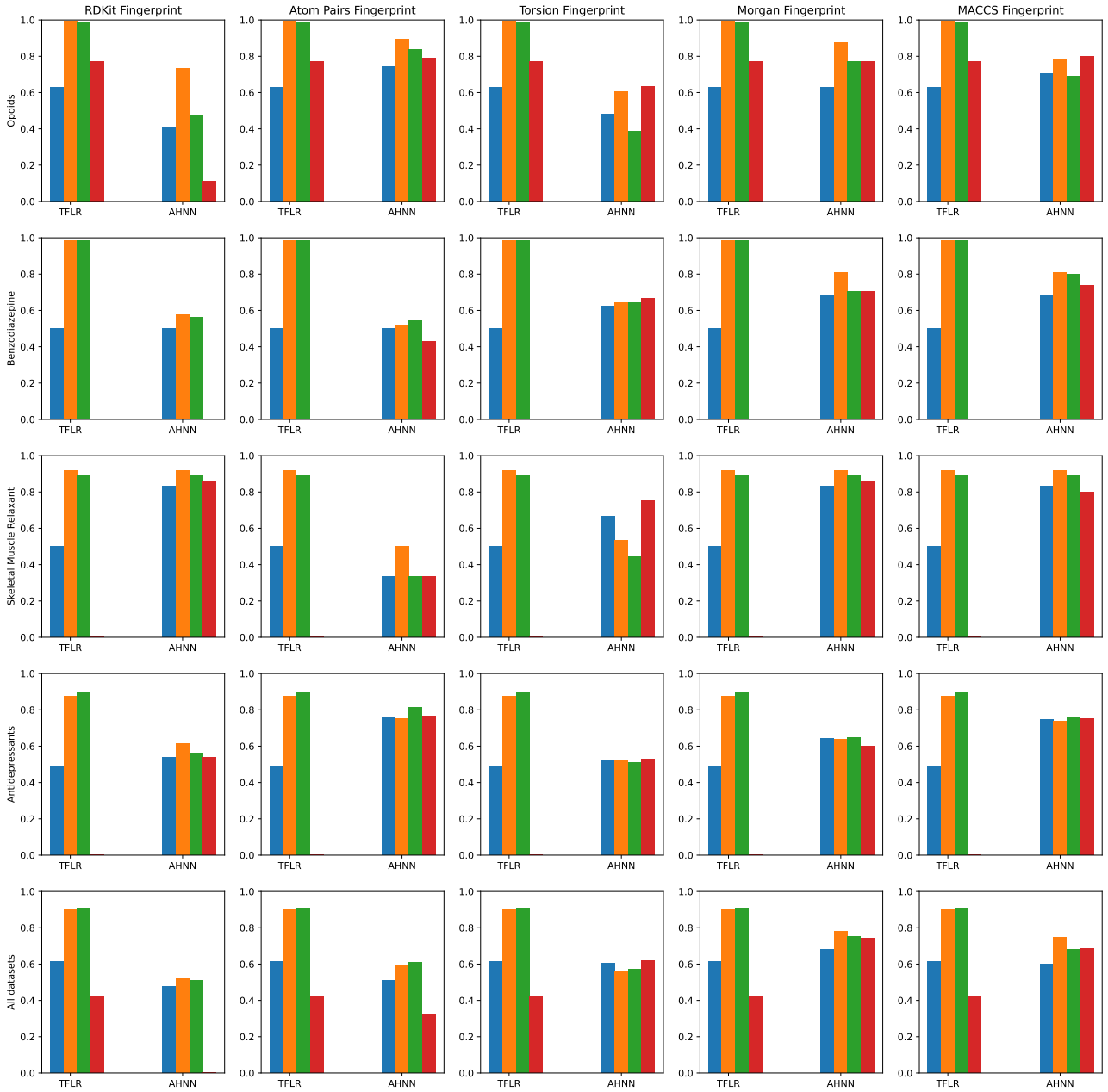


Table 4.3: Transductive predictions of TDLR and AHNN

Drug1	Drug2	Drug3	TDLR score	AHNN score	Label
Omeprazole	Cefuroxime	Cefdinir	0.00287844	0.578928	1
Methylprednisolone	Megestrol acetate	Topiramate	0.00253684	0.73728867	1
Amiodarone	Allopurinol	Citalopram	0.00153684	0.40559068	0

Table 4.4: Inductive predictions of AHNN

Drug1	Drug2	Drug3	AHNN score	Label
Baclofen	Lisinopril	Alprazolam	0.573078	1
Cyclobenzaprine	Fenofibrate	Gabapentin	0.63833857	1
Lisinopril	Tizanidine	Trimethoprim	0.470732	0

4.5.1 Model comparison for transductive learning

We evaluated TDLR and AHNN's ability to perform transductive predictions, that is, predictions of new HODIs on known drugs. For that, we train both models on 90% of all HODIs and ask them to predict the 10% remaining ones. Some of the results collected are shown in Table 4.3. As can be seen, both models predicted the potential existence of new HODIs correctly, illustrating the model's capacity to predict HODIs in the transductive case.

4.5.2 AHNN for inductive learning

In addition to the ability to predict HODIs among known drugs, AHNN is able to perform inductive learning and predict interactions among unknown drugs. Inductive prediction of HODIs is much more challenging as it potentially involves families of drugs that have not been seen before. To test the ability of AHNN to predict interaction among unseen drugs, we train the model on a dataset containing the opoids, benzodiazepine and antidepressants. We then ask it to predict HODIs in the skeletal muscle relaxant dataset.

With an accuracy of 0.6, F1 score of 0.7, AUROC of 0.6 and AUPR of 0.65, AHNN shows difficulties learning to predict HODIs among new drugs. As mentioned in Section 4.5, poorer performance could indicate a lack of training data to generalize over multiple types of drugs.

Chapter 5

Conclusions & Future Directions

We presented tensor decomposition with Laplacian regularization and attention-based hypergraph neural networks, two approaches for predicting higher-order drug interactions. TDLR is a tensor decomposition algorithm specifically tailored for regularization over the ambient space and intrinsic manifold defined by the drug fingerprints. We propose a fast and stable alternating least-squares optimization algorithm based on Newton's method to predict new HODIs. AHNN is an attention-based hypergraph neural network encoder-decoder pair that uses attention at the hyperedge level and at the node level to learn latent drug embeddings.

Both models show promising results on HODI prediction tasks, particularly when restricted to a specific drug family. We provide a detailed discussion of the different components of each model and provide a contrasted comparison of the two models. Both models give insights into the features and aspects of drug molecules that can lead to HODIs. In addition, both models can be applied to low-data scenarios, where only the SMILES string of the drug molecule is provided. AHNN shows particularly promising applications because it can be used on drugs on which the model has not been trained.

The main limitation of our work is the quantity and quality of the data. We have samples containing 0.002% of the drugs under study, which severely restricts the prediction capabilities, inducing extreme sparsity in the HODI tensor for TDLR and reducing the generalization capabilities of AHNN. Furthermore, as there is no current agreement on HODI detection methods, the quality of the data is constrained. More research is needed to handle these data sparsity and quality issues.

There are several directions for future study. Our approaches are restricted to fingerprints computed over drug SMILES strings for added flexibility, but integrating more data sources would allow for enhanced generalization capabilities. Specifically, the AHNN approach is suitable for knowledge hypergraphs which can be constructed from datasets such as DrugBank [46] and KEGG [90]. Building foundation models trained on knowledge hypergraphs would allow the model to extract more knowledge from

the small datasets presented in this work. As TDLR and AHNN are flexible in learning over different hypergraphs, it would be interesting to apply them to other domains and problems, for example, drug-drug-target interaction predictions, drug-drug-side-effect interaction predictions, or protein complex predictions.

Bibliography

[1] Frank L Hitchcock. "The Expression of a Tensor or a Polyadic as a Sum of Products". In: *Journal of Mathematics and Physics* 6.1-4 (1927), pp. 164–189. DOI: https://doi.org/10.1002/sapm192761164.

[2] Ledyard R. Tucker. "Some Mathematical Notes on Three-Mode Factor Analysis". In: *Psychometrika* 31.3 (Sept. 1966), pp. 279–311. ISSN: 0033-3123. DOI: 10.1007/BF02289464.

[3] Pieter M. Kroonenberg and Jan de Leeuw. "Principal Component Analysis of Three-Mode Data by Means of Alternating Least Squares Algorithms". In: *Psychometrika* 45.1 (Mar. 1980), pp. 69–97. ISSN: 0033-3123. DOI: 10.1007/BF02293599.

[4] Dennis H. Smith, Raymond E. Carhart, and R. Venkataraghavan. "Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications". In: *Journal of Chemical Information and Computer Sciences* 25.2 (May 1985), pp. 64–73. DOI: 10.1021/ci00046a002.

[5] Ramaswamy Nilakantan et al. "Topological Torsion: A New Molecular Descriptor for SAR Applications. Comparison with Other Descriptors". In: *Journal of Chemical Information and Computer Sciences* 27.2 (May 1987), pp. 82–85. ISSN: 00952338. DOI: 10.1021/ci00046a002.

[6] Alberto Gobbi and Dieter Poppinger. "Genetic optimization of combinatorial libraries". In: *Biotechnology and Bioengineering* 61.1 (1998), pp. 47–54. DOI: https://doi.org/10.1002/(SICI)1097-0290(199824)61:1<47::AID-BIT9>3.0.CO;2-Z.

[7] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. "A Multilinear Singular Value Decomposition". In: *SIAM Journal on Matrix Analysis and Applications* 21.4 (Mar. 2000), pp. 1253–1278. ISSN: 08954798. DOI: 10.1137/S0895479896305696.

[8] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. "On the Best Rank-1 and Rank-(R1,R2,. . .,RN) Approximation of Higher-Order Tensors". In: *SIAM Journal on Matrix Analysis and Applications* 21.4 (Mar. 2000), pp. 1324–1342. ISSN: 08954798. DOI: 10.1137/S0895479898346995.

[9] Yvonne C. Martin, James L. Kofron, and Linda M. Traphagen. "Do structurally similar molecules have similar biological activity?" In: *Journal of Medicinal Chemistry* 45.19 (Sept. 2002), pp. 4350–4358. ISSN: 00222623. DOI: /10.1021/jm020155c.

[10] Mikhail Belkin and Partha Niyogi. "Towards a Theoretical Foundation for Laplacian-Based Manifold Methods". In: *Proceedings of the 18th Annual Conference on Learning Theory*. Ed. by Peter Auer and Ron Meir. Vol. 3559 LNAI. Springer, Berlin, Heidelberg, July 2005, pp. 486–500. ISBN: 978-3-540-31892-7. DOI: 10.1007/11503415{_}33.

[11] Mikhail Belkin, Partha Niyogi, and Vika Sindhwani. "Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples". In: *The Journal of Machine Learning Research* 7 (Dec. 2006), pp. 2399–2434. DOI: 10.5555/1248547.1248632.

[12] Ansgar Schuffenhauer et al. "Relationships between molecular complexity, biological activity, and structural diversity". In: *Journal of Chemical Information and Modeling* 46.2 (Dec. 2006), pp. 525–535. ISSN: 1549960X. DOI: /10.1021/ci0503558.

[13] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. "Learning with Hypergraphs: Clustering, Classification, and Embedding". In: *Proceedings of the 19th International Conference on Neural Information Processing Systems*. Ed. by Bernhard Schölkopf, John Platt, and Thomas Hofmann. Cambridge: The MIT Press, Sept. 2007, pp. 1601–1608. DOI: 10.7551/mitpress/7503.003.0205.

[14] Morien Mørup, Lars Kai Hansen, and Sidse M. Arnfred. "Algorithms for sparse nonnegative tucker decompositions". In: *Neural Computation* 20.8 (Aug. 2008), pp. 2112–2131. ISSN: 08997667. DOI: 10.1162/NECO.2008.11-06-407.

[15] Tamara G Kolda and Brett W Bader. "Tensor Decompositions and Applications". In: *SIAM Review* 51.3 (2009), pp. 455–500. DOI: 10.1137/07070111X.

[16] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Yee Whye Teh and Mike Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, Jan. 2010, pp. 249–256.

[17] David Rogers and Mathew Hahn. "Extended-connectivity fingerprints". In: *Journal of Chemical Information and Modeling* 50.5 (May 2010), pp. 742–754. ISSN: 1549960X. DOI: 10.1021/ci100050t.

[18] Peter Willett. "Similarity Searching Using 2D Structural Fingerprints". In: *Chemoinformatics and Computational Chemical Biology*. Ed. by Jürgen Bajorath. 1st ed. Humana Totowa, NJ, Jan. 2010. Chap. 5, pp. 133–158. ISBN: 978-1-60761-839-3. DOI: 10.1007/978-1-60761-839-3{_}5.

[19] Zheng Xia et al. "Semi-supervised drug-protein interaction prediction from heterogeneous biological spaces". In: *BMC Systems Biology* 4.Supplement 2 (Sept. 2010). ISSN: 1752-0509. DOI: 10.1186/1752-0509-4-S2-S6.

[20] Wei Dong, Moses Charikar, and Kai Li. "Efficient K-nearest neighbor graph construction for generic similarity measures". In: *Proceedings of the 20th International Conference on World Wide Web*. New York: Association for Computing Machinery, Mar. 2011, pp. 577–586. DOI: 10.1145/1963405.1963487.

[21] "Fingerprints - Screening and Similarity". In: *Daylight Theory Manual*. Daylight Chemical Information Systems, Inc., Aug. 2011. Chap. 6.

[22] John R. Horn and Philip D. Hansten. "Triple Drug Interactions". In: *Pharmacy Times* 77.1 (Jan. 2011).

[23] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12.85 (2011), pp. 2825–2830.

[24] Qinxue Meng and Paul J Kennedy. "Using network evolution theory and singular value decomposition method to improve accuracy of link prediction in social networks". In: *Proceedings of the Tenth Australasian Data Mining Conference*. Australian Computer Society, Inc., Dec. 2012. ISBN: 9781921770142. DOI: 10.5555/2525373.2525394.

[25] Santiago Vilar et al. "Drug—drug interaction through molecular structure similarity analysis". In: *Journal of the American Medical Informatics Association* 19.6 (Nov. 2012), pp. 1066–1074. DOI: 10.1136/AMIAJNL-2012-000935.

[26] Kevin Wood et al. "Mechanism-independent method for predicting response to multidrug combinations in bacteria". In: *Proceedings of the National Academy of Sciences of the United States of America*. Vol. 109. 30. National Academy of Sciences, July 2012, pp. 12254–12259. DOI: 10.1073/pnas.1201281109.

[27] Sereina Riniker and Gregory A. Landrum. "Similarity maps - A visualization strategy for molecular fingerprints and machine-learning methods". In: *Journal of Cheminformatics* 5.9 (Sept. 2013), pp. 1–7. DOI: 10.1186/1758-2946-5-43.

[28] Santiago Vilar et al. "Detection of Drug-Drug Interactions by Modeling Interaction Profile Fingerprints". In: *PLOS ONE* 8.3 (Mar. 2013). ISSN: 1932-6203. DOI: 10.1371/JOURNAL.PONE.0058321.

[29] C. Robert Jr. Horsburgh, Clifton E. III Barry, and Christoph Lange. "Treatment of Tuberculosis". In: *New England Journal of Medicine* 373.22 (Nov. 2015). Ed. by Dan L. Longo, pp. 2149–2160. ISSN: 0028-4793. DOI: 10.1056/NEJMRA1413919.

[30] Thomas N. Kipf and Max Welling. "Variational Graph Auto-Encoders". In: (Nov. 2016).

[31] Elif Tekin et al. "Enhanced identification of synergistic and antagonistic emergent interactions among three or more drugs". In: *Journal of The Royal Society Interface* 13.119 (June 2016). ISSN: 17425662. DOI: 10.1098/RSIF.2016.0332.

[32] Bolun Chen et al. "Link prediction based on non-negative matrix factorization". In: *PLOS ONE* 12.8 (Aug. 2017). ISSN: 1932-6203. DOI: 10.1371/JOURNAL.PONE.0182968.

[33] Murat Cokol et al. “Efficient measurement and factorization of high-order drug interactions in Mycobacterium tuberculosis”. In: *Science Advances* 3.10 (Oct. 2017). DOI: 10.1126/sciadv.1701881.

[34] Brian G.M. Durie et al. “Bortezomib with lenalidomide and dexamethasone versus lenalidomide and dexamethasone alone in patients with newly diagnosed myeloma without intent for immediate autologous stem-cell transplant (SWOG S0777): a randomised, open-label, phase 3 trial”. In: *The Lancet* 389.10068 (Feb. 2017), pp. 519–527. ISSN: 0140-6736. DOI: 10.1016/S0140-6736(16)31594-X.

[35] William L Hamilton, Rex Ying, and Jure Leskovec. “Inductive Representation Learning on Large Graphs”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Ed. by Ulrike von Luxburg et al. Red Hook, NY, USA: Curran Associates Inc., Dec. 2017, pp. 1025–1035. DOI: 10.5555/3294771.3294869.

[36] Liqun Qi and Ziyan Luo. “Spectral Hypergraph Theory via Tensors”. In: *Tensor Analysis: Spectral Theory and Special Tensors*. Society for Industrial and Applied Mathematics, Apr. 2017. Chap. 4, pp. 121–172. ISBN: 978-1-611974-75-1. DOI: 10.1137/1.9781611974751.CH4.

[37] Stephan Rabanser, Oleksandr Shchur, and Stephan Günnemann. “Introduction to Tensor Decompositions and their Applications in Machine Learning”. In: (Nov. 2017).

[38] Elif Tekin, Van M. Savage, and Pamela J. Yeh. “Measuring higher-order drug interactions: A review of recent approaches”. In: *Current Opinion in Systems Biology* 4 (Aug. 2017), pp. 16–23. DOI: 10.1016/J.COISB.2017.05.015.

[39] Marian Tietz et al. *skorch: A scikit-learn compatible neural network library that wraps PyTorch*. July 2017.

[40] Yu Chen Lo et al. “Machine learning in chemoinformatics and drug discovery”. In: *Drug Discovery Today* 23.8 (Aug. 2018), pp. 1538–1546. ISSN: 1359-6446. DOI: 10.1016/J.DRUDIS.2018.05.010.

[41] Alberto Lumbreras, Louis Filstroff, and Cédric Févotte. “Bayesian Mean-parameterized Nonnegative Binary Matrix Factorization”. In: *Data Mining and Knowledge Discovery* 34.6 (Dec. 2018), pp. 1898–1935. ISSN: 1573756X. DOI: 10.1007/s10618-020-00712-w.

[42] Jian Yu Shi et al. “TMFUF: A triple matrix factorization-based unified framework for predicting comprehensive drug-drug interactions of new drugs”. In: *BMC Bioinformatics* 19.Suplement 14 (Nov. 2018), pp. 27–37. DOI: 10.1186/s12859-018-2379-8.

[43] Elif Tekin et al. “Prevalence and patterns of higher-order drug interactions in Escherichia coli”. In: *npj Systems Biology and Applications 2018 4:1* 4.1 (Sept. 2018), pp. 1–10. ISSN: 2056-7189. DOI: 10.1038/s41540-018-0069-9.

[44] Petar Veličković et al. "Graph Attention Networks". In: *Proceedings of the 6th International Conference on Learning Representations*. Curan Associates Inc., Feb. 2018. DOI: 10.1007/978-3-031-01587-8{_}7.

[45] Jonathan H. Watanabe, Terry McInnis, and Jan D. Hirsch. "Cost of Prescription Drug–Related Morbidity and Mortality". In: *Annals of Pharmacotherapy* 52.9 (Sept. 2018), pp. 829–837. ISSN: 15426270. DOI: 10.1177/1060028018765159/ASSET/IMAGES/LARGE/10.1177{_}1060028018765159-FIG2.JPEG.

[46] David S Wishart et al. "DrugBank 5.0: a major update to the DrugBank database for 2018". In: *Nucleic Acids Research* 46.D1 (Jan. 2018), pp. D1074–D1082. ISSN: 0305-1048. DOI: 10.1093/nar/gkx1037.

[47] Hui Yu et al. "Predicting and understanding comprehensive drug-drug interactions via semi-nonnegative matrix factorization". In: *BMC Systems Biology* 12.1 (2018), p. 14. ISSN: 1752-0509. DOI: 10.1186/s12918-018-0532-7.

[48] Wen Zhang et al. "Manifold regularized matrix factorization for drug-drug interaction prediction". In: *Journal of Biomedical Informatics* 88 (Nov. 2018), pp. 90–97. DOI: 10.1016/J.JBI.2018.11.005.

[49] Marinka Zitnik, Monica Agrawal, and Jure Leskovec. "Modeling polypharmacy side effects with graph convolutional networks". In: *Bioinformatics* 34.13 (June 2018), pp. 457–466. DOI: 10.1093/BIOINFORMATICS/BTY294.

[50] Matthias Fey and Jan Eric Lenssen. "Fast Graph Representation Learning with PyTorch Geometric". In: (Mar. 2019).

[51] Itay Katzir et al. "Prediction of ultra-high-order antibiotic combinations based on pairwise interactions". In: *PLOS Computational Biology* 15.1 (Jan. 2019), e1006774. DOI: 10.1371/JOURNAL.PCBI.1006774.

[52] Jean Kossaifi et al. "TensorLy: Tensor Learning in Python". In: *Journal of Machine Learning Research* 20.26 (2019), pp. 1–6.

[53] Martin Lukačišin and Tobias Bollenbach. "Emergent Gene Expression Responses to Drug Combinations Predict Higher-Order Drug Interactions". In: *Cell Systems* 9.5 (Nov. 2019), pp. 423–433. ISSN: 24054720. DOI: 10.1016/j.cels.2019.10.004.

[54] Jian Yu Shi et al. "Detecting drug communities and predicting comprehensive drug–drug interactions via balance regularized semi-nonnegative matrix factorization". In: *Journal of Cheminformatics* 11.1 (Apr. 2019), pp. 1–16. ISSN: 17582946. DOI: 10.1186/S13321-019-0352-9.

[55] Melike Cokol-Cakmak et al. "Guided screen for synergistic three-drug combinations". In: *PLOS ONE* 15.7 (July 2020). DOI: 10.1371/JOURNAL.PONE.0235929.

[56] Charles R Harris et al. "Array programming with NumPy". In: *Nature* 585 (2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2.

[57] Narjes Rohani, Changiz Eslahchi, and Ali Katanforoush. "ISCMF: Integrated similarity-constrained matrix factorization for drug–drug interaction prediction". In: *Network Modeling Analysis in Health Informatics and Bioinformatics* 9 (Jan. 2020), p. 11. DOI: 10.1007/s13721-019-0215-3.

[58] Govind Sharma, Prasanna Patil, and M Narasimha Murty. "C3MM: Clique-Closure based Hyperlink Prediction". In: *Proceedings of the Twenty-Ninth International Joint Conference onArtificial Intelligence*. Ed. by Christian Bessiere. International Joint Conferences on Artificial Intelligence Organization, Jan. 2020, pp. 3364–3370. DOI: 10.24963/ijcai.2020/465.

[59] Pauli Virtanen et al. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[60] Song Bai, Feihu Zhang, and Philip H.S. Torr. "Hypergraph convolution and hypergraph attention". In: *Pattern Recognition* 110 (Feb. 2021), p. 107637. DOI: 10.1016/J.PATCOG.2020.107637.

[61] Vivien Cabannes et al. "Overcoming the curse of dimensionality with Laplacian regularization in semi-supervised learning". In: *Proceedings of the 35th International Conference on Neural Information Processing Systems*. NIPS '21. Red Hook, NY, USA: Curran Associates Inc., 2021, pp. 30439–30451. ISBN: 9781713845393. DOI: 10.5555/3540261.3542590.

[62] Deepak Maurya and Balaraman Ravindran. "Hyperedge Prediction Using Tensor Eigenvalue Decomposition". In: *Journal of the Indian Institute of Science* 101.3 (July 2021), pp. 443–453. DOI: 10.1007/S41745-021-00225-5.

[63] Farhad Pazan and Martin Wehling. "Polypharmacy in older adults: a narrative review of definitions, epidemiology and consequences". In: *European Geriatric Medicine* 12.3 (June 2021), pp. 443–452. ISSN: 18787657. DOI: 10.1007/S41999-021-00479-3/FIGURES/1.

[64] Swiss National Supercomputing Center. *CSCS, Hewlett Packard Enterprise and NVIDIA Announce World's Most Powerful AI-Capable Supercomputer*. Apr. 2021.

[65] Nhi Van, Yonatan N. Degefu, and Bree B. Aldridge. "Efficient Measurement of Drug Interactions with DiaMOND (Diagonal Measurement of N-Way Drug Interactions)". In: *Mycobacteria Protocols*. Humana Press Inc., July 2021. Chap. 30, pp. 703–713. DOI: 10.1007/978-1-0716-1460-0{_}30.

[66] Yingheng Wang et al. "Multi-view Graph Contrastive Representation Learning for Drug-Drug Interaction Prediction". In: *Proceedings of the World Wide Web Conference* (Oct. 2021), pp. 2921–2933. DOI: 10.1145/3442381.3449786.

[67] Emily K. Acton et al. "Opioid Drug-Drug-Drug Interactions and Unintentional Traumatic Injury: Screening to Detect Three-Way Drug Interaction Signals". In: *Frontiers in Pharmacology* 13.845485 (May 2022). DOI: 10.3389/fphar.2022.845485.

[68] David Buterez et al. "Graph Neural Networks with Adaptive Readouts". In: *Proceedings of the 36th Annual Conference on Neural Information Processing Systems*. Curran Associates, Inc., 2022, pp. 19746–19758.

[69] Cheng Chen et al. "Population-based screening to detect benzodiazepine drug-drug-drug interaction signals associated with unintentional traumatic injury". In: *Scientific Reports* 12.15569 (Sept. 2022). DOI: 10.1038/s41598-022-19551-4.

[70] Cheng Chen et al. "Skeletal muscle relaxant drug–drug–drug interactions and unintentional traumatic injury: Screening to detect three-way drug interaction signals". In: *British Journal of Clinical Pharmacology* 88.11 (Nov. 2022), pp. 4773–4783. DOI: 10.1111/BCP.15395.

[71] Ke Han et al. "A Review of Approaches for Predicting Drug–Drug Interactions Based on Machine Learning". In: *Frontiers in Pharmacology* 12.1 (Jan. 2022), p. 814858. DOI: 10.3389/fphar.2021.814858.

[72] Haohuai He, Guanxing Chen, and Calvin Yu Chian Chen. "3DGT-DDI: 3D graph and text based neural network for drug–drug interaction prediction". In: *Briefings in Bioinformatics* 23.3 (May 2022). ISSN: 14774054. DOI: 10.1093/BIB/BBAC134.

[73] Jonah Larkins-Ford et al. "Design principles to assemble drug combinations for effective tuberculosis therapy using interpretable pairwise drug response measurements". In: *Cell Reports Medicine* 3.9 (Sept. 2022). DOI: 10.1016/j.xcrm.2022.100737.

[74] Chaoqi Yang et al. "Semi-supervised Hypergraph Node Classification on Hypergraph Line Expansion". In: *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*. Association for Computing Machinery, Oct. 2022, pp. 2352–2361. ISBN: 9781450392365. DOI: 10.1145/3511808.3557447.

[75] Can Chen and Yang Yu Liu. "A Survey on Hyperlink Prediction". In: *IEEE Transactions on Neural Networks and Learning Systems* (June 2023). DOI: 10.1109/TNNLS.2023.3286280.

[76] Cheng Chen et al. "Antidepressant drug–drug–drug interactions associated with unintentional traumatic injury: Screening for signals in real-world data". In: *Clinical and Translational Science* 16.2 (Feb. 2023), pp. 326–337. DOI: 10.1111/CTS.13452.

[77] Pamela Dow, Veronique Michaud, and Jacques Turgeon. "Multidrug Interactions: Why Do They Occur and How to Handle?" In: *Clinical Therapeutics* 45.2 (Feb. 2023), pp. 99–105. DOI: 10.1016/J.CLINTHERA.2022.12.012.

[78] Xuan Lin et al. "Comprehensive evaluation of deep and graph learning on drug–drug interactions prediction". In: *Briefings in Bioinformatics* 24.4 (July 2023). DOI: 10.1093/BIB/BBAD235.

[79] Natalie Ann Lozano-Huntelman et al. "The evolution of resistance to synergistic multi-drug combinations is more complex than evolving resistance to each individual drug component". In: *Evolutionary Applications* 16.12 (Dec. 2023), pp. 1901–1920. ISSN: 1752-4571. DOI: 10.1111/EVA.13608.

[80] Khaled Mohammed Saifuddin et al. "HyGNN: Drug-Drug Interaction Prediction via Hypergraph Neural Network". In: *Proceedings of the 39th International Conference on Data Engineering*. Anaheim: IEEE, Apr. 2023, pp. 1503–1516. DOI: 10.1109/ICDE55515.2023.00119.

[81] Liqiao Yan and Yi Cao. "HUF-DCIE: A Hypergraph Based Unified Framework to Predict Drug Combination Integrated Effects". In: *Proceedings of the 5th International Conference on Machine Learning, Big Data and Business Intelligence*. New York: IEEE, 2023, pp. 274–279. DOI: 10.1109/MLBDBI60823.2023.10481969.

[82] Emily K. Acton et al. "Thinking Three-Dimensionally: A Self- and Externally-Controlled Approach to Screening for Drug–Drug–Drug Interactions Among High-Risk Populations". In: *Clinical Pharmacology & Therapeutics* 116.2 (Aug. 2024), pp. 448–459. DOI: 10.1002/CPT.3310.

[83] Jason Ansel et al. "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation". In: *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*. ASPLOS '24. New York, NY, USA: Association for Computing Machinery, 2024, pp. 929–947. ISBN: 9798400703850. DOI: 10.1145/3620665.3640366.

[84] Sunwoo Kim et al. "A Survey on Hypergraph Neural Networks: An In-Depth and Step-By-Step Guide". In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. New York: ACM, Aug. 2024, pp. 6534–6544. DOI: 10.1145/3637528.3671457.

[85] Greg Landrum et al. *rdkit/rdkit: 2024_03_5 (Q1 2024) Release*. July 2024. DOI: 10.5281/zenodo.12782092.

[86] Xinyue Li et al. "Deep learning for drug-drug interaction prediction: A comprehensive review". In: *Quantitative Biology* 12.1 (Feb. 2024), pp. 30–52. DOI: 10.1002/qub2.32.

[87] Oliver Lloyd, Yi Liu, and Tom R. Gaunt. "Tensor Factorisation for Polypharmacy Side Effect Prediction". In: (Apr. 2024), pp. 1–8. DOI: 10.5281/zenodo.

[88] Shilin Qu et al. "Scalable and Effective Negative Sample Generation for Hyperedge Prediction". In: (Nov. 2024).

[89] Ning Ning Wang et al. "Comprehensive Review of Drug-Drug Interaction Prediction Based on Machine Learning: Current Status, Challenges, and Opportunities". In: *Journal of Chemical Information and Modeling* 64.1 (Dec. 2024), pp. 96–109. DOI: 10.1021/acs.jcim.3c013.

[90] Minoru Kanehisa et al. "KEGG: biological systems database as a model of the real world". In: *Nucleic Acids Research* 53.D1 (Jan. 2025), pp. D672–D677. DOI: 10.1093/NAR/GKAE909.

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Higher-order Drug Interactions with
Hypergraph Neural Networks

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Maillefaud	Jonathan

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zürich, 19/01/2025	

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.