

MOOMIN: Deep Molecular Omics Network for Anti-Cancer Drug Combination Therapy

Benedek Rozemberczki
AstraZeneca
Cambridge, United Kingdom

Anna Gogleva
AstraZeneca
Cambridge, United Kingdom

Sebastian Nilsson
AstraZeneca
Gothenburg, Sweden

Gavin Edwards
AstraZeneca
Cambridge, United Kingdom

Andriy Nikolov
AstraZeneca
Cambridge, United Kingdom

Eliseo Papa
AstraZeneca
Cambridge, United Kingdom

ABSTRACT

We propose the molecular omics network (MOOMIN) a multimodal graph neural network used by AstraZeneca oncologists to predict the synergy of drug combinations for cancer treatment. Our model learns drug representations at multiple scales based on a drug-protein interaction network and metadata. Structural properties of compounds and proteins are encoded to create vertex features for a message-passing scheme that operates on the bipartite interaction graph. Propagated messages form multi-resolution drug representations which we utilized to create drug pair descriptors. By conditioning the drug combination representations on the cancer cell type we define a synergy scoring function that can inductively score unseen pairs of drugs. Experimental results on the synergy scoring task demonstrate that MOOMIN outperforms state-of-the-art graph fingerprinting, proximity preserving node embedding, and existing deep learning approaches. Further results establish that the predictive performance of our model is robust to hyperparameter changes. We demonstrate that the model makes high-quality predictions over a wide range of cancer cell line tissues, out-of-sample predictions can be validated with external synergy databases, and that the proposed model is data efficient at learning.

ACM Reference Format:

Benedek Rozemberczki, Anna Gogleva, Sebastian Nilsson, Gavin Edwards, Andriy Nikolov, and Eliseo Papa. 2022. MOOMIN: Deep Molecular Omics Network for Anti-Cancer Drug Combination Therapy. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

A large number of human diseases are treated more effectively by the concurrent use of multiple drugs together in *combination therapy* [18, 58]. It is a particularly powerful approach in combating cancer, bacterial and viral infections [25, 47, 66] which can develop resistance when treated by *monotherapy* – using a single drug. The application of monotherapy in oncology has several downsides in

addition to the development of resistance: high toxic dosages of the drug [23], long-term side effects and poorly targeted treatment which destroys healthy cells [63]. In contrast, combination therapy requires lower dosages [4], the drugs can have synergies at combating cancer [8] and the intelligent targeting of multiple biological pathways can lead to increased efficacy and reduced toxicity [98]. Hence, finding synergistic drug combinations in oncology is a highly relevant healthcare problem.

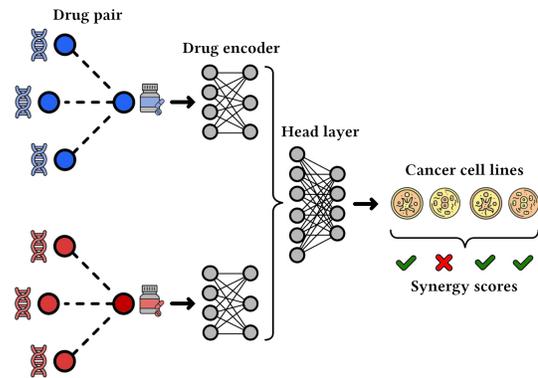


Figure 1: Given two drugs (red and blue nodes) MOOMIN represents drugs using protein and compound structures that appear in the neighbourhood of the pair in an interaction graph. Using the representations its head predicts synergistic pairs which are effective at combating cancer.

The manual design of effective combination therapy regimes for cancer therapy is a non-trivial task. Even on a single targeted cancer cell type, the number of potential drug pairings (the simplest type of combinations) is quadratic in the number of compounds. This means that only a limited number of potential candidate pairs can be tested by in-vitro and in-vivo experiments. In addition, the experimental evaluation of combinations is hindered by drug and cell line availability [44], conflicting experimental results [56] and the fact that certain outcomes (e.g. polypharmacy side effect) are not the primary interest of experiments [103]. Moreover, the synergistic effect of drug pairs depends on the context in which they are applied – the biological pathways the drugs interact with and the cancer cells being targeted [55]. Finally, drug combination experiments are done in a sparse and skewed manner: a limited number of drug pairs are tested on a few cells out of the combinatorial range of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, July 2017, Washington, DC, USA

© 2022 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

possibilities [55, 56]. A machine learning-based approach that pre-screens drug combinations would be valuable, shortening the time required to find therapeutic combinations to use in the clinic.

Present work. In this paper we introduce the molecular omics network (MOOMIN), a semi-supervised graph neural network for predicting the synergistic nature of drug pairings for combating cancer cells. We design a custom-tailored multi-resolution multi-modal feature extraction model which contextualizes drugs based on molecular features and structural properties of proteins that they interact with in a bipartite graph. Contextualization happens by traversing a drug-protein interaction graph at multiple scales with truncated random walks [35]. Using the extracted multi-resolution representations we define a scoring layer that can output synergy scores for pairs of drugs and a targeted cancer cell line. The components of the model are trained end-to-end by maximizing the likelihood of correct synergy predictions.

Our main contribution is the definition of multi-scale multimodal drug representations. These learned features integrate molecular information about a given drug and about the drugs and proteins that are in the neighborhood of the source drug at multiple proximity scales in a drug-protein interaction graph. By using a bipartite interaction graph for modeling, the information about drugs and proteins at multiple scales is distilled by separate blocks of features. Therefore, the synergy scoring performance lift gained by the inclusion of multimodal drug-protein information is quantifiable with our model. We also propose an algorithm to approximately contextualize the drugs. This allows scalable model training and efficient synergy scoring inference when a dimension of the dataset such as the number of drugs considered is large.

The empirical evaluation of our model focuses on the synergy scoring task: given a pair of drugs and a cancer cell line the models have to predict whether the drug combination is synergistic. We compare the predictive performance of MOOMIN with state-of-the-art graph fingerprinting, node embedding, tensor factorization, and deep learning techniques. Using a real-world drug pair synergy database [56] we show that MOOMIN outperforms competing methods by as much as 7.2% and 2.5% in terms of F_1 and PR AUC scores on the test set. The predictive performance of MOOMIN is robust across cancer cells conditioned on source tissues and competitive with other methods with a fraction of training data. Out-of-sample predictions made by AstraZeneca scientists on standard drug combination databases demonstrate that MOOMIN can identify novel synergistic drug combinations of approved drugs.

Our contributions. Overall the main contributions of our paper can be summarized as the following:

- We propose MOOMIN an empirically motivated multimodal graph neural network that can predict synergistic drug pairs using protein and molecule features.
- We design an efficient mini-batch training algorithm to scale the training and inference of MOOMIN to large-scale drug synergy prediction problems.
- We support evidence using real-world data that MOOMIN can effectively find synergistic drug pairings.

The remainder of this paper has the following structure. We overview the related work about network representation learning and machine-aided combination therapy in Section 2. We define

relevant theoretical concepts, introduce the datasets used in the experiment and show the empirical regularities motivating our model design in Section 3. The details of the model are discussed in Section 4. We experimentally evaluate the model in Section 5 on synergy prediction tasks. The paper concludes with Section 6 where we summarize our main findings and point out directions for future research.

2 RELATED WORK

Our work touches on machine learning approaches to identify combination therapies as well as various existing graph representation learning techniques at the node and graph levels. We are going to discuss these methods and contrast MOOMIN based on various desired characteristics of the models.

2.1 Machine learning for combination therapy

The application of machine learning to combination therapy is commonly done by solving a scoring problem: given a pair of drugs (combination) we want to predict the probability that the pair has a property. The predicted property can be a certain side-effect caused by the combination [94, 103], an interaction of drugs [1, 95] or the synergistic behavior of the pair [53, 69]. We summarized recent algorithmic developments on these tasks in Table 1.

Existing models are differentiated by the task solved and whether predictions can be made about drugs that are not present at training time (induction). Another differentiating factor is the type of input data ingested by the models. Most models exploit heterogeneous graphs with various node types (e.g. drugs, proteins) and structural features of drugs and proteins. On the synergy scoring task MOOMIN is the first one to fuse drug-protein interaction information with actual structural properties of drugs and proteins.

Table 1: A machine learning task, induction and input data type based comparison of MOOMIN and existing deep learning models that solve combination therapy problems.

Method	Task	Node Types		Node Features	
		Inductive Drug	Protein	Drug	Protein
SkipGNN [43]	Side Effect	•			
DECAGON [103]	Side Effect	•	•		
ESP [11]	Side Effect	•	•		
TIP [94]	Side Effect	•	•		
DeepDDI [77]	Side Effect	•			•
SumGNN [101]	Side Effect	•	•		•
DeepCCI [1]	Interaction	•			•
MR-GNN [95]	Interaction	•			•
DrugCell [53]	Synergy	•		•	•
DeepSynergy [69]	Synergy	•			•
MOOMIN (ours)	Synergy	•	•	•	•

2.2 Node and graph representation learning

The architecture proposed in our work distills representations from graphs both at the node and graph level (interactome and molecules). To position our work we give a general overview of machine learning techniques that can learn these representations.

2.2.1 Proximity preserving node representation learning. Node embedding techniques in this general category map nodes of a graph

into a Euclidean embedding space where *pairwise proximity* between nodes is approximately preserved [83]. Models are differentiated from each other by the notion of proximity such as truncated random walk transition probabilities [35, 67, 70, 75], neighbourhood overlap [3, 79] or personalized PageRank [62]. Importantly encoding nodes in this embedding space does not optimize for the preservation of structural properties directly [39, 40, 71, 72], but this information can be correlated with the location in the embedding space. Using the node embeddings as input features for a downstream model allows various supervised machine learning tasks such as node classification.

2.2.2 Graph fingerprints. These statistical fingerprinting techniques create embeddings of whole graphs which preserve *pairwise structural similarities*. Graph fingerprints are commonly created by the factorization of graph-structural feature matrices [17, 60], scattering transforms [29, 88], spectral feature extraction [28, 82, 84] and structural property summarization [12, 76]. The fingerprints created are fed to a downstream model which ingests these features to predict graph level properties.

2.2.3 Inductive node representation learning. In contrast with proximity-based node embedding techniques these methods map nodes into an embedding space based on the similar distribution of vertex attributes in neighborhoods [36, 38]. Inductive node representation learning techniques can be seen as methods that perform *neural message passing* [7, 33]. In this paradigm nodes in the graph generate hidden representations with a parametric trainable function and the messages are aggregated in neighborhoods using a weighting scheme and an update function. Models are differentiated from each other based on the message generating function and the message aggregation scheme applied. Messages can be generated by linear models [91], single hidden layer [22, 49] or deep neural networks [9, 51]. Aggregation weights can be defined based on exact [19, 49, 91] or sampled adjacency relations [37, 99], personalized PageRank [9, 51] or neural network parametrization [73, 87].

2.2.4 Pooling for graph representations. Node level representations outputted by graph neural networks can be pooled by permutation invariant functions to generate whole graph representations. This permutation invariant function can be a neural network itself which can perform node sorting [102], top-k node retrieval [13, 30, 52] or calculate a weighted average of representations using an attention mechanism [52, 54, 100] to generate the graph level representations.

3 PRELIMINARIES

Our work considers contextualizing drugs at multiple scales by structural and molecular properties of compounds and proteins to predict synergetic pairings of drugs. The introduction of a machine learning model which can achieve this requires an appropriate notation of related concepts and a discussion of the datasets integrated.

3.1 Formal definitions

The architecture design description extensively uses the definition of drug set $\mathcal{D} = \{d_1, \dots, d_n\}$, cancer cell line set $\mathcal{C} = \{c_1, \dots, c_k\}$ and protein set $\mathcal{P} = \{p_1, \dots, p_m\}$. Introducing the sets of drugs that can be used to combat cancer cells, proteins that interact with these

drugs, and cancer cell lines allows us to formulate the definition of a drug pair synergy set which is our primary interest.

DEFINITION 1. Drug pair synergy set. A drug pair synergy set defined on \mathcal{D} and \mathcal{C} is the set \mathcal{S} containing tuples of the form (d, d', c, y) where $d, d' \in \mathcal{D}$, $c \in \mathcal{C}$ and $y \in \{0, 1\}$ is the synergy label of the drug pairing.

The entries in the drug pair synergy set are pairs of drugs which are known to be synergistic or antagonistic at combating certain cancer cells. A positive label means that the pair of drugs are more effective together at fighting the cancer cell, while a zero label shows a lack of synergy.

DEFINITION 2. Drug pair synergy predictor. A drug pair synergy predictor defined on \mathcal{D} and \mathcal{C} is the function $\hat{y} = h(d, d', c)$ where $d, d' \in \mathcal{D}$, $c \in \mathcal{C}$ and $0 \leq \hat{y} \leq 1$ is the predicted synergy score of the drug pair.

A drug pair synergy predictor is a function that predicts the probability of the pair being synergistic at combating a cancer cell.

DEFINITION 3. Drug-protein interaction graph. The interaction graph defined on \mathcal{D} and \mathcal{P} is the bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}) = (\mathcal{D} \cup \mathcal{P}, \mathcal{E})$ where \mathcal{V} and \mathcal{E} are the vertex and edge sets respectively.

Our definition of the drug-protein interaction graph does not allow the existence of edges within the drug and protein sets. This hedges against potential data leakage on drug-drug interactions.

3.2 Dataset

The dataset [32] which we used to evaluate MOOMIN was created by the fusion of multiple data sources which we discuss in detail.

3.2.1 Synergy database. The database which we used comes from DrugCombDB which is a public dataset [56]. After the data fusion, our drug pair synergy set contained 10,911 pairings, 987 unique drugs, and 110 cell lines coming from 14 source tissues. Drug pairs have a positive label when the pair slowed the growth of cancer cells more than what would be expected, based on the effect of the compounds when used alone (synergy), while a zero label is associated with increased cell growth when combined (antagonism).

3.2.2 Drug-protein interaction graph. The drug and protein interaction network was composed of several sources and has 1,160 drugs and 17,582 human proteins with 232,524 edges between vertices. Drug-protein relations were mainly taken from three public sources: (i) ChEMBL [31], which included both the verified targets against which the compound was developed as well as interactions discovered experimentally, (ii) CTD [21], which aggregates information from research publications, and (iii) Hettinet [41], which itself accumulates a small set of sources to cover all the most relevant domain concepts. These public reference datasets were extended with internal as well as proprietary data from two drug development database products: MetaBase [10] and Pharmaprojects [80].

3.2.3 Molecular descriptors. The drug molecules in DrugCombDB are represented as SMILES strings [90], a common notation for describing molecular graphs in string form. From the SMILES strings, we extracted the molecular structure of the drugs (connectivity graph and atoms). The structure was extracted using PaDELPy [46], a Python wrapper of the PaDEL-Descriptor [97] package.

3.2.4 Protein features. Protein sequence information can be represented by various classes of numeric descriptors. In this study, we used amino acid composition descriptors (dipeptide composition), along with pseudo-amino acid composition descriptors (amphiphilic pseudo-amino acid composition). Amino acid composition descriptors reflect the frequency of dipeptide combinations of amino acids in a protein sequence. Pseudo-amino acid composition descriptors in addition incorporate local properties of protein sequences such as the correlation between residues or a certain distance between groups of residues [20]. All descriptors were computed on a full set of human proteins using the protpackage [92].

3.3 Exploratory observations and motivation

The effects of drug-drug combinations are widely studied experimentally, the number of parameter combinations to test makes such validation very time-consuming and expensive [56]. Thus, available experimental data on synergistic and antagonistic effects are necessarily sparse. Moreover, it is hard to extrapolate from existing pairs to new pairs of drugs or new types of cells due to several factors.

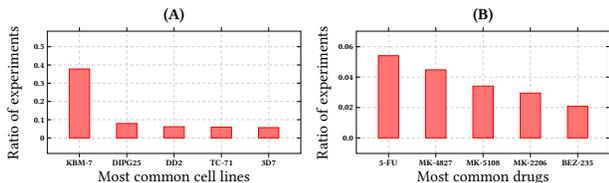


Figure 2: Drug combination pairs are frequently (A) tested on popular cell lines and (B) include popular drugs.

3.3.1 Popular cell lines and drugs. Experiments are often performed only with popular drugs on popular cell lines, while effects in some tissues are rarely studied. This also makes it difficult to train separate predictive models for different cell lines and drugs. Looking at Figure 2 it can be seen that 63% of experiments in DrugCombDB are done on the 5 most popular cell lines and 18% of synergy tests are carried out with the top 5 drugs.

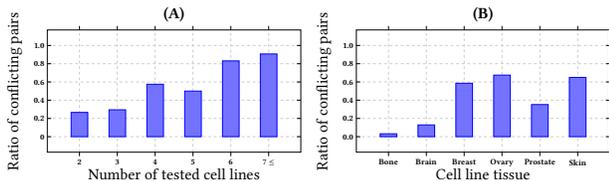


Figure 3: Drug combination pairs often produce opposite effects on different cell lines (synergy vs antagonism). The share of such pairs (A) grows with the number of tested cell lines and (B) varies greatly depending on the tissue.

3.3.2 Opposing effects. In DrugCombDB third of drug pairs that have been tested on different cell lines have shown a synergistic effect on one cell line, but antagonistic on another. As seen in Figure 3, as the same pair of drugs are tested against multiple cell lines, the chance of discovering opposite effects grows: i.e., to predict all effects of applying drugs in combination it is important to take into account the whole range of potentially affected cells

and tissues. This problem is made even more complex by the fact that the same pair of drugs often exhibit opposite effects within the same tissue: e.g., this was the case for 67% of drug pairs tested on ovary cell lines, but only less than 4% for bone.

4 THE FRAMEWORK DESIGN

The design of MOOMIN was influenced by the empirical regularities which we discussed in the preliminaries: dataset imbalance, sparsity of the synergy database, and cell line conditional synergistic and antagonistic behavior of drug pairs.

4.1 Encoders

The decision about a pair of drugs being synergistic at effectively shrinking a cancer cell line depends on information about the pair of drugs, protein-drug interactions, and the targeted cell line.

4.1.1 Drug encoder. Learning non-contextualized representations of an individual drug $d \in \mathcal{D}$ is handled by the drug encoder $f_{\Theta_D}(\cdot)$ which is parametrized by Θ_D , uses the molecular structure \mathcal{M}_d to generate \mathbf{h}_d a vector representation of drugs – see Equation (1).

$$\mathbf{h}_d = f_{\Theta_D}(\mathcal{M}_d), \quad \forall d \in \mathcal{D} \quad (1)$$

A drug encoder, for example, could use a graph neural network to learn node embeddings of the atoms in the molecule, pool these atom-level representations, and concatenate the pooled representations to create a hidden representation learned from the molecule structure and atom-level features. Our implementation of MOOMIN used APPNP [9, 51] for learning the atom embeddings. Atom representations were fed to a fully connected feed-forward neural network with two hidden layers and concatenated together after pooling with mean, maximum, and minimum functions to generate molecular descriptors.

4.1.2 Protein encoder. Creating representations of proteins in the interaction graph is done by the encoder $f_{\Theta_P}(\cdot)$ which is parametrized by Θ_P and uses structural information \mathbf{x}_p to generate \mathbf{h}_p a vector representation of the protein $p \in \mathcal{P}$ – see Equation (2).

$$\mathbf{h}_p = f_{\Theta_P}(\mathbf{x}_p), \quad \forall p \in \mathcal{P} \quad (2)$$

Protein encoders in MOOMIN are feedforward neural networks with a single hidden layer and take structural feature vectors of proteins as input.

4.1.3 Cell line encoder. The cancer cell lines on which the drugs are tested also need representations which are generated by the cell encoder $f_{\Theta_C}(\cdot)$, parametrized by Θ_C which uses cancer cell features as input in Equation (3).

$$\mathbf{h}_c = f_{\Theta_C}(\mathbf{x}_c), \quad \forall c \in \mathcal{C} \quad (3)$$

The cell line encoders were conceptualized as trainable parametric vector embeddings of cancer cells in our implementation of MOOMIN. This way induction concerning new cell lines is not possible. However, by choosing an encoder that uses exogenous cell features this limitation can be lifted.

4.2 Multimodal drug representation

The representations of drugs in combinations are learned from multimodal data and the drug-protein graph. Individual representations

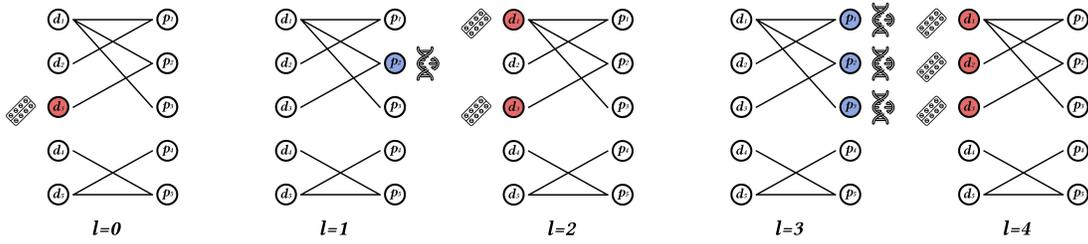


Figure 4: In this example the drug d_3 has a representation formed by concatenating aggregated drug and protein representations. These representation blocks at even scales $l = 0, 2, 4$ are learned by aggregating the representations output by the drug-encoder using geometric properties of the compounds in the sets $\{d_1\}$, $\{d_1; d_3\}$ and $\{d_1; d_2; d_3\}$. The odd scale representations at $l = 1, 3$ are learned by combining the protein representations generated by the protein encoder using the structural properties of the proteins in the sets $\{p_1\}$ and $\{p_1; p_2; p_3\}$.

of the compounds are contextualized by those protein and drug representations which can be reached at various proximity scale $l \in \{0, \dots, r\}$ by truncated random walks. In practice this design choice allows the model to learn *difference operators* [2] at various scales. This idea is described with a toy example in Figure 4.

4.2.1 Contextualization by neighbors at jumps. The multi-scale representation for drug $d \in \mathcal{D}$ is the vector \mathbf{m}_d defined by Equation (4) – where \parallel is the column-wise concatenation operator.

$$\mathbf{m}_d = \parallel_{l \in \{0, \dots, r\}} \sum_{w \in V} P(v_{j+l} = w | v_j = d) \cdot \mathbf{h}_w, \quad \forall d \in \mathcal{D} \quad (4)$$

In Equation (4) the probability $P(v_{j+l} = w | v_j = d)$ describes the probability that a truncated discrete random walk on \mathcal{G} which started at drug d after l steps terminates at node w . The vector \mathbf{h}_w is a learned representation of $w \in V$ outputted by respectively the drug or protein encoders described by Equations (1) and (2). Because \mathcal{G} is bipartite, when the scale is even the walker terminates at drugs, and at an odd scale, it terminates at proteins. In simple terms, \mathbf{m}_d is formed by concatenating together proximity-weighted drug and protein representations in an alternating fashion.

The probability scores in Equation (4) can be expressed algebraically. Let us denote the row normalized adjacency matrix of \mathcal{G} as $\hat{\mathbf{A}}$ and we can reformulate the multi-scale representation definition of drugs as Equation (5).

$$\mathbf{m}_d = \parallel_{l \in \{0, \dots, r\}} \hat{\mathbf{A}}_{d,w}^l \cdot \mathbf{h}_w, \quad \forall d \in \mathcal{D} \quad (5)$$

The exact calculation of the multi-scale multimodal drug representations described by Equation (5) is not always feasible in practical settings. One limiting condition is an indirect result of the *low effective diameter phenomenon* – as the length of the truncated random walks being considered is increased the normalized adjacency matrix powers lose sparsity [16]. In practice this means $\mathcal{O}(|V|^3)$ time and $\mathcal{O}(|V|^2)$ space requirement for calculating and storing $\hat{\mathbf{A}}^l$ when l is large. Another considerable issue is the space

requirement needed for all of the protein and drug representations when the drug-protein graph is large.

```

Data:  $\mathcal{G}$  – Bipartite drug – protein interaction graph.
          $d$  – Source drug of interest.
          $s$  – Sample size.
Result:  $\bar{\mathbf{m}}_d$  – Sampled multi-scale representation of drug  $d$ .
1  $\tilde{\mathbf{m}}_d^0 \leftarrow f_{\Theta_D}(\mathcal{M}_d)$ 
2  $\bar{\mathbf{m}}_d \leftarrow \mathbf{0}$ 
3 for  $i \in \{1, \dots, s\}$  do
4    $\tilde{d} \leftarrow d$ 
5    $\tilde{\mathbf{m}}_d \leftarrow \tilde{\mathbf{m}}_d^0$ 
6   for  $l \in \{1, \dots, r\}$  do
7     if  $l \bmod 2 = 1$  then
8        $\tilde{p} \leftarrow \text{Random neighbor}(\mathcal{G}, \tilde{d})$ 
9        $\tilde{\mathbf{m}}_d^l \leftarrow f_{\Theta_P}(\mathbf{x}_{\tilde{p}})$ 
10       $\tilde{\mathbf{m}}_d \leftarrow [\tilde{\mathbf{m}}_d \parallel \tilde{\mathbf{m}}_d^l]$ 
11     else
12        $\tilde{d} \leftarrow \text{Random neighbor}(\mathcal{G}, \tilde{p})$ 
13        $\tilde{\mathbf{m}}_d^l \leftarrow f_{\Theta_D}(\mathcal{M}_d)$ 
14        $\tilde{\mathbf{m}}_d \leftarrow [\tilde{\mathbf{m}}_d \parallel \tilde{\mathbf{m}}_d^l]$ 
15     end
16   end
17    $\bar{\mathbf{m}}_d \leftarrow \bar{\mathbf{m}}_d + \tilde{\mathbf{m}}_d$ 
18 end
19  $\bar{\mathbf{m}}_d \leftarrow \bar{\mathbf{m}}_d / s$ 

```

Algorithm 1: Approximate truncated random walk sampled multi-scale drug representation forward pass with MOOMIN.

4.2.2 A sampling-based representation approach. The computational constraints discussed earlier can be resolved by approximating the multi-scale multimodal drug representations with a uniform truncated random walk-based sampling approach. This representation sampling technique is described by Algorithm 1. The algorithm uses the bipartite graph \mathcal{G} and the starting drug d as input and outputs $\bar{\mathbf{m}}_d$ an approximation of \mathbf{m}_d . Before sampling starts we create $\tilde{\mathbf{m}}_d^0$ the self representation of the drug d using the drug encoder $f_{\Theta_D}(\cdot)$ (line 1) and $\bar{\mathbf{m}}_d$ the vector to store the sum of the sampled multi-scale multimodal drug representations (line 2).

We start s truncated random walks on \mathcal{G} starting from the drug d (lines 3-4). A sampled drug representation $\tilde{\mathbf{m}}_d$ is formed by first taking the self representation $\tilde{\mathbf{m}}_d^0$. At each scale, the random walker takes a step on the bipartite graph. If the step is at an odd scale the walker samples the protein \tilde{p} , a representation is created for the protein with the protein encoder $f_{\Theta_P}(\cdot)$ and \tilde{m}_d^l is concatenated to the sampled multi-scale multimodal representation (lines 7-10). When the step is at an even scale the walker samples a drug \tilde{d} , a representation is distilled with the drug encoder $f_{\Theta_D}(\cdot)$ and it is concatenated to the sample vector (lines 12-14).

After the random walk is terminated, the sampled vector $\tilde{\mathbf{m}}_d$ is added to the running sum of vectors (line 17). Finally, when all of the truncated random walks have terminated $\bar{\mathbf{m}}_d$ which stores the sum of the sampled multi-scale multimodal drug representations is normalized by the sample size (line 19).

<p>Data: \mathcal{B} – Batch of (drug A, drug B, cell line, label) tuples. \mathcal{G} – Graph of interest. s – Sample size for multi-scale approximation.</p> <p>Result: \mathcal{L} – Average loss on batch \mathcal{B}.</p> <pre> 1 $\mathcal{L} \leftarrow 0$ 2 for $(d, d', c, y) \in \mathcal{B}$ do 3 $\tilde{\mathbf{m}}_d \leftarrow \text{Sample Representation}(\mathcal{G}, d, s)$ 4 $\tilde{\mathbf{m}}_{d'} \leftarrow \text{Sample Representation}(\mathcal{G}, d', s)$ 5 $\mathbf{h}_{(d,d',c)} \leftarrow [\tilde{\mathbf{m}}_d \parallel \tilde{\mathbf{m}}_{d'} \parallel \mathbf{h}_c]$ 6 $\hat{y}_{(d,d',c)} \leftarrow f_{\Theta_H}(\mathbf{h}_{(d,d',c)})$ 7 $\mathcal{L} \leftarrow \mathcal{L} + \ell(y; \hat{y}_{(d,d',c)}) / \mathcal{B}$ 8 end</pre>

Algorithm 2: Mini-batch based forward pass of MOOMIN-A with approximate multi-scale multimodal representations.

4.3 Synergy scoring layer and training

The primary purpose of MOOMIN is to predict the synergistic nature of drug pairings conditioned on the targeted cancer cell lines. We achieve this with $f_{\Theta_H}(\cdot)$ the synergy scoring layer of our model defined by Equation (7) which depends on the drug pair-cell representation $\mathbf{h}_{(d,d',c)}$ defined by Equation (6). The layer outputs $\hat{y}_{d,d',c}$ the probability that the drug pair d, d' is effective at killing the cancer cell c . In practice, the synergy scoring layer is a feedforward neural network parametrized by Θ_H . It has a single hidden layer and a final neuron with a sigmoid activation function.

$$\mathbf{h}_{(d,d',c)} = [\mathbf{m}_d \parallel \mathbf{m}_{d'} \parallel \mathbf{h}_c] \quad (6)$$

$$\hat{y}_{(d,d',c)} = f_{\Theta_H}(\mathbf{h}_{(d,d',c)}) \quad (7)$$

Using the synergy probability outputted by the synergy scoring layer we compute the loss for a single entry in the drug synergy database based on the binary cross-entropy in Equation (8).

$$\ell(y; \hat{y}) = -[y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})] \quad (8)$$

These losses can be averaged out over the synergy database and used to calculate the mean binary cross-entropy on the synergy database \mathcal{S} defined by Equation (9).

$$\mathcal{L} = \sum_{(d,d',c,y) \in \mathcal{S}} \ell(y_{d,d',c}; \hat{y}_{d,d',c}) / |\mathcal{S}| \quad (9)$$

The model which calculates the transition probabilities in Equation (4) and the average loss in Equation (9) exactly is referenced as

MOOMIN-E. By minimizing this average loss, $\Theta_D, \Theta_P, \Theta_C$ and Θ_H the parameters of the encoders and the synergy scoring layer can be learned jointly. We also propose an efficient mini-batching-based approximate model called MOOMIN-A which uses the approximation technique described by Algorithm 1 for creating multi-scale representations and batching procedure of Algorithm 2.

5 EXPERIMENTAL EVALUATION

In this section, we evaluate the predictive performance of MOOMIN under various training and testing regimes. Furthermore, we provide results about the efficiency of learning, stability of the approximate inference, validate model predictions on external datasets, and analyze the hidden representations visually.

5.1 Predictive performance

The drug synergy scoring problem can be solved by a range of existing machine learning techniques. Our goal is to have an extensive predictive performance comparison with the existing set of tools.

5.1.1 Baselines. The benchmark methods cover a range of graph and node level representation learning algorithms. Unsupervised methods use a two-stage pipeline; (i) drug embedding generation and pairwise representation concatenation (ii) gradient boosted machine training and scoring with scikit-learn [65]. Specifically, the baseline techniques are:

- *Molecule statistical descriptors.* We calculate graph descriptors for each drug molecule using the default settings of the *Karate Club* [74] library.
- *Node embeddings.* We create higher order proximity-preserving node embeddings from the drug-protein interaction graph with the baseline hyperparameters of *Karate Club* [74].
- *Tensor factorization.* We factorize the heterogeneous interaction graph with *PyKeen* [5] and use the drug node representations as features with the default settings.
- *Graph neural networks.* Using the molecular structure extracted from the SMILES strings we train pooled graph neural networks with PyTorch Geometric [26].

5.1.2 Experimental settings. We used default hyperparameters from [49] and [51] in order to ensure fair and comparable evaluation. The APPNP *drug encoders* in MOOMIN uses 10 approximate Personalized PageRank [9, 51] iterations with a return probability of 0.2 and has two hidden layers with 32 neurons separated by ReLU [59] activations. Atom representations are pooled by mean, max and min pooling functions, and the graph representations are concatenated together. *Protein encoders* in MOOMIN had a single hidden layer with 32 neurons and the same activation functions. *Cancer cell line encoders* are 16 dimensional *embeddings* initialized with the GloRot method [34]. The *synergy prediction layer* uses 16 hidden layer neurons, ReLU activations and applies a dropout of 0.5 during training time [78]. We consider truncated random walks at scales $r = \{0, 1, 2\}$ to assess the utility of multiple modalities and higher-order information.

The exact and approximate MOOMIN variants are optimized using the Adam optimizer [48] and use a learning rate of $5 \cdot 10^{-3}$ with a weight decay of $5 \cdot 10^{-5}$. The number of samples used by MOOMIN-A is 2^7 and the batch size is set to be 2^5 . Both MOOMIN-E

and MOOMIN-A are implemented with the PyTorch and PyTorch Geometric Temporal automatic differentiation libraries [26, 64]. All of the baseline models and MOOMIN variants are trained with 80% of the drug synergy database and evaluated on the remaining 20% of entries. We performed 5-fold cross-validation within the training set to find the optimal number of epochs based on early stopping. The mean synergy scoring performance of models calculated from 10 train/test splits is in Table 2 with standard errors.

Table 2: The synergy scoring performance of MOOMIN, statistical fingerprinting, node embedding and deep learning approaches. We report mean performance and the standard error around the mean calculated from 10 splits. Bold numbers denote the best performing model on a metric.

	ROC AUC	PR AUC	F ₁ Score
Graph2Vec [60]	.745 ± .002	.632 ± .004	.563 ± .002
NetLSD [82, 84]	.723 ± .003	.604 ± .004	.490 ± .005
FEATHER [76]	.732 ± .003	.610 ± .005	.520 ± .004
GeoScattering [29]	.726 ± .004	.612 ± .004	.499 ± .004
NetMF [70]	.751 ± .001	.678 ± .006	.553 ± .004
GraRep [14]	.760 ± .001	.684 ± .006	.595 ± .003
Walklets [68]	.762 ± .002	.689 ± .001	.581 ± .002
LINE [79]	.760 ± .002	.686 ± .005	.576 ± .003
HOPE [62]	.757 ± .003	.679 ± .006	.578 ± .005
DeepWalk[67]	.761 ± .002	.686 ± .003	.595 ± .004
GCN [50]	.714 ± .005	.613 ± .009	.534 ± .053
GAT [87]	.700 ± .006	.593 ± .005	.563 ± .021
MixHop [2]	.731 ± .002	.633 ± .004	.537 ± .066
ComplEx [81]	.764 ± .003	.691 ± .002	.581 ± .003
RESCAL [61]	.762 ± .002	.684 ± .004	.577 ± .003
Tucker [6]	.748 ± .003	.664 ± .004	.521 ± .003
ComboFM [45]	.760 ± .002	.681 ± .005	.563 ± .018
DualE [15]	.758 ± .003	.683 ± .004	.576 ± .002
SEEK [96]	.755 ± .002	.676 ± .003	.548 ± .004
ReinceptionE [93]	.758 ± .002	.685 ± .003	.553 ± .003
CompGCN [86]	.764 ± .003	.688 ± .002	.584 ± .006
MOOMIN-E $r = 0$.705 ± .003	.639 ± .004	.533 ± .002
MOOMIN-E $r = 1$.777 ± .002	.702 ± .002	.632 ± .003
MOOMIN-E $r = 2$.770 ± .003	.708 ± .002	.656 ± .003
MOOMIN-A $r = 0$.718 ± .003	.656 ± .005	.545 ± .006
MOOMIN-A $r = 1$.775 ± .004	.699 ± .005	.588 ± .005
MOOMIN-A $r = 2$.753 ± .003	.674 ± .006	.568 ± .008

5.1.3 Hyperparameters for fair evaluation. The baselines' hyperparameters ensure a fair evaluation with respect to expressive power (number of free parameters) and the optimizer settings because:

- (1) Tensor factorization, node embedding, and graph fingerprinting techniques have the same embedding dimensions.
- (2) Drug pair classifier graph neural network models are trained with the same optimizer settings as MOOMIN.
- (3) It holds at $r < 3$ that MOOMIN has fewer embedding dimensions and free trainable parameters than the node embedding, graph fingerprinting, and tensor factorization techniques.

- (4) When $r = 0$ the drug pair classifier graph neural networks and MOOMIN have the same number of trainable parameters and the optimizer settings are the same.

5.1.4 Experimental findings. Our results in Table 2 demonstrate that the MOOMIN models can significantly outperform a range of strong baselines on all evaluation metrics. In terms of F_1 score this advantage can be as big as 7.5%, while the gain with respect to AUC score is only 1.7%. Our results show that incorporating protein features at the 1st scale is a universally beneficial choice regardless of the metric used for evaluation. Higher than 1st order information only improves the performance of MOOMIN-E when PR AUC and F_1 scores are utilized. Our results also validate the efficacy of MOOMIN-A in the low-order proximity regime. In fact, for $r = 1$ based on the mean performance of MOOMIN-E and MOOMIN-A measured by ROC AUC and PR AUC is not significantly different.

5.2 Tissue conditional performance

Cancer cell lines on which the drug pairs are tested all have a source tissue from which they were originally extracted. We specifically analyze whether the predictive performance of MOOMIN-E depends on the source tissue of cancer cells.

5.2.1 Experimental settings. We utilize the hyperparameters from Subsection 5.1. Using 10 experimental runs we calculate mean predictive performance metrics with standard errors around the mean on the test set conditioned on the most common cell line tissues. We plotted the average PR AUC and F_1 scores with the standard error bars on subplots of Figure 5 for the proximity scales $r \in \{0, 1, 2\}$.

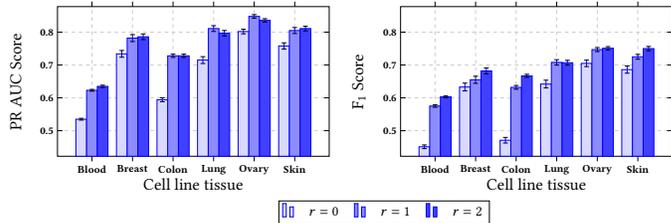


Figure 5: The synergy scoring performance of MOOMIN conditioned on cell line tissue. The bars are mean performance metrics with standard deviations around the mean calculated from 10 experimental repetitions .

5.2.2 Experimental findings. The bar charts on Figure 5 show that predictive performance is cell line tissue dependent: the model performs remarkably well on lung and ovary cells and relatively poorly on blood and colon cells. Including structural information about the interacting proteins can increase the test set F_1 scores on blood and colon cells by as much as 10% which is a remarkable predictive performance difference. The same performance gain pattern holds for the PR AUC based performance on the blood and colon tissue-specific cancer cell lines.

5.3 Data efficiency

The design of MOOMIN allows for semi-supervised learning; the characteristics of drugs that are not in the training database can be exploited to contextualize pairs of drugs used for training better.

We hypothesize that because of this MOOMIN has a reasonable predictive performance in the low training data ratio regime.

5.3.1 Experimental settings. We train MOOMIN-E with the hyperparameter settings detailed in Subsection 5.1. For each training data ratio, we do 10 splits and calculate the average predictive performance on the test set. These mean performance scores are plotted on subplots of Figure 6 as a function of training data ratio for the proximity scales $r \in \{0, 1, 2, 3\}$.

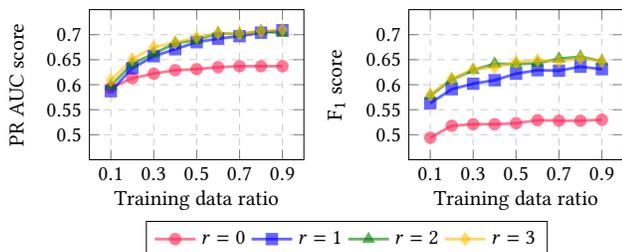


Figure 6: The mean synergy scoring performance of MOOMIN-E conditioned on the ratio of training data calculated from 10 experimental runs.

5.3.2 Experimental findings. The line charts in Figure 6 demonstrate that MOOMIN is efficient at utilizing the training data as the average test set performance increases steadily. In terms of PR AUC and F₁ scores, the higher order models can outperform baselines by using half of the training data when Table 2 and Figure 6 are cross-referenced. It is also evident that the purely molecular feature based 0th scale model has moderate gains with additional training data compared to true multimodal models.

Table 3: The molecule and window size conditional test set performance of MOOMIN-E. We report average predictive performances calculated from 10 experimental runs with standard errors around the mean.

Molecule Pair	Window	ROC AUC	PR AUC	F ₁ Score
Large - Large	0	.724 ± .007	.767 ± .007	.676 ± .022
	1	.785 ± .007	.808 ± .008	.733 ± .010
	2	.763 ± .013	.803 ± .008	.745 ± .006
Large - Small	0	.713 ± .005	.656 ± .008	.543 ± .022
	1	.778 ± .005	.713 ± .008	.656 ± .012
	2	.768 ± .006	.717 ± .012	.678 ± .007
Small - Small	0	.614 ± .008	.447 ± .011	.312 ± .016
	1	.715 ± .006	.554 ± .010	.477 ± .016
	2	.716 ± .007	.563 ± .010	.516 ± .013

5.4 Molecule size conditional performance

The drugs in DrugCombDB are heterogeneous concerning the number of atoms they contain and the pairings sometimes have drugs that are considerably different concerning size.

5.4.1 Experimental settings. Using the MOOMIN-E test set synergy scores obtained in Subsection 5.1 we calculate molecule size conditional performance metrics for various window sizes. We used the following definitions to categorize the drug pairs based on the molecule sizes for calculating the conditional performance metrics:

- (i) *Large - Large*: If both molecules have 50 or more non-hydrogen atoms;
- (ii) *Large - Small*: If only one of the molecules has 50 or more non-hydrogen atoms;
- (iii) *Small - Small*: If both molecules have less than 50 non-hydrogen atoms.

5.4.2 Experimental findings. Our results show that pairs with two small molecules gain the most in relative terms by adding the protein feature-based context. These pairs have gains even when the maximal window size is increased to 2. This heterogeneity of performances concerning the molecule size and maximal scale being considered implies that molecule sizes specific models should be trained. For example, small molecule pairs should be scored with a MOOMIN model with a large maximal scale.

5.5 Inference stability

The MOOMIN-A variant uses sampled truncated random walks to approximate the transition probabilities between the vertices in the interaction graph. Inference also uses a sampling-based approach – the variance of the test set performance is affected by sample size.

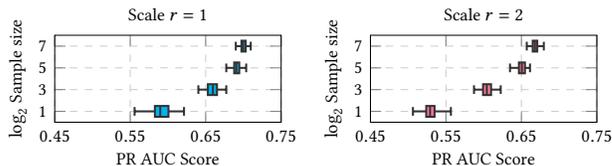


Figure 7: The stability of predictive performance under sampling based approximate inference with MOOMIN-A. Each boxplot represents the distribution of predictive performance scores calculated from 100 inference runs.

5.5.1 Experimental settings. We trained MOOMIN-A with the hyperparameter settings detailed in Subsection 5.1 at proximity scales $r \in \{1, 2\}$. Using the trained model we make a 100 sampling-based predictions on the test set with $s \in \{2^1, 2^3, 2^5, 2^7\}$. For each prediction, we calculate PR AUC scores and visualized the distribution of predictive performance metrics on subplots of Figure 7 conditioned on the sample size.

5.5.2 Experimental findings. Looking at Figure 7 we can conclude that increasing the number of truncated random walk samples increases the average predictive performance and reduces the variance of performance. The marginal average performance gains are decreasing with the sample size, but the order of decline depends on the resolution considered. Our results also support that the second-scale model could benefit from increasing the sample size at training and inference time.

5.6 Out of sample model validation

Data sparsity is one of the main motivations for designing MOOMIN, so it can predict new synergy scores. Using out-of-sample combinations and the NCI Synergy Score Almanach [42] AstraZeneca oncologists validated a handful of pairings with the cancer drug Vemurafenib [27].

5.6.1 Experimental settings. We use a MOOMIN-E model with $r = 1$ and the hyperparameter settings from Subsection 5.1 and train the model on the whole drug synergy database. We scored

all of the triples which involved the drug *Vemurafenib* [27] – the predictions were sorted by the synergy scores and the 20 highest and lowest scoring triples were selected. Out of these combinations and cell targets, we listed seven in Table 4 which are present in the NCI Almanach [42] but are not included in the training database.

Table 4: Out of sample drug combination predictions on (Vemurafenib, Drug, Cell line) triples that can be externally validated in the NCI Almanach [42].

Drug A	Drug B	Cell line	Tissue	Score	Label
Vemurafenib	Cabazitaxel	MDA-MB-435	breast	.831	synergy
	Pazopanib	NCI-H522	lung	.803	synergy
	Crizotinib	UO-31	kidney	.807	synergy
	Raloxifene	SNB-75	brain	.860	synergy
	Axitimib	MDA-MB-231	breast	.043	antagonism
	Carboplatin	LOX IMVI	skin	.158	antagonism
	Chlorambucil	SR	blood	.022	antagonism

5.6.2 Experimental findings. Out of sample predictions in Table 4 which were manually validated come from a diverse set of drugs and cell lines that are derived from multiple source tissues (e.g kidney and brain). This shows that MOOMIN can identify new synergistic drug combinations without doing the actual experiments.

5.7 Runtime

The whole dataset used for training the MOOMIN variants is fairly small in terms of drug combinations. Training and scoring the exact or approximate model end-to-end on a Tesla V-100 GPU using all of the drug pairings *takes less than 2 seconds*. We investigate the relative runtime required by MOOMIN-A to make a weight update step using a single data batch. The average relative runtime for scales $r \in \{1, 2\}$ calculated from 100 repetitions is visualized on the subplots of Figure 8. When the batch size and sampling count values are small MOOMIN-A has a material advantage over MOOMIN-E, the runtime values converge for larger batch and sample sizes.

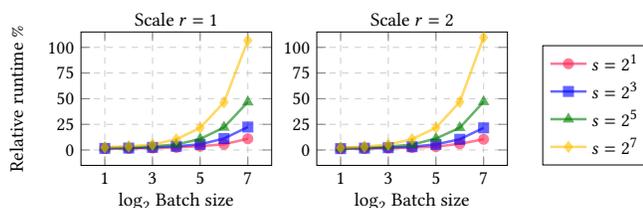


Figure 8: The relative runtime of a forward and backpropagation pass with MOOMIN-A compared to MOOMIN-E.

5.8 Visualizing representations

The architecture of MOOMIN consists of separate encoders and a downstream synergy scoring head layer which are trained jointly for solving the scoring task optimally.

5.8.1 Experimental settings. Using a fully trained MOOMIN-E model we extract the drug pair-cell and scoring head layer internal representations on the test set. We learn two-dimensional embeddings with the scikit-learn implementation [65] of t-SNE [85, 89] and visualized the location of drug pairs on Figure 9 and colored the points by class membership – synergistic and antagonistic pairings.

5.8.2 Experimental findings. The results in Figure 9 demonstrate that the separation of synergistic and antagonistic drug pairs starts before representations are passed through the scoring layer. Embeddings based on the synergy scoring layer show a clean class conditional separation of pairs. These findings suggest that multi-scale representations created with MOOMIN could allow transfer learning to other tasks e.g. polypharmacy side effect prediction.

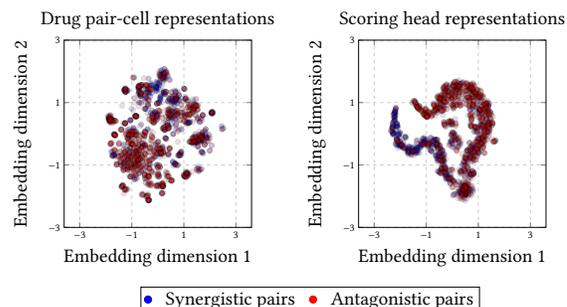


Figure 9: Test set drug pair-cell and scoring head representations embedded by t-SNE and colored by class membership.

6 CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we proposed the molecular omics network (MOOMIN), a multi-modal heterogeneous graph neural network custom-tailored to predict the synergistic and antagonistic nature of drug combinations in oncology. The model consists of protein feature and molecular structure encoders which contextualize drugs at multiple scales using a bipartite drug-protein interactions graph. Using the multi-scale representations and learned cancer cell embeddings a head layer outputs synergy scores for drug pairs conditioned on the cancer cell lines. The encoders, cancer cell embeddings, and the scoring head layer were trained jointly by minimizing a binary cross-entropy-based custom loss designed for the synergy scoring task. We introduced an approximation algorithm that makes memory-efficient model training/inference possible.

Our empirical evaluation of MOOMIN which used a publicly available drug combination synergy database had shown that the predictive performance of our framework outperforms state-of-the-art proximity-based node embedding, tensor factorization, statistical fingerprinting, and deep learning methods on the synergy scoring task. We also supported evidence that learning multi-modal representations that can distill information from molecular and protein structures improves predictive performance. Our results demonstrated that this gain holds even when the performance is conditioned on the tissue of the cell lines or the amount of training data is reduced by magnitudes. Further experiments characterized approximate inference and internal representations.

We are excited about the potential future developments of our work. The enrichment of the multi-modal data is one possibility. One avenue is considering other heterogeneous biological graphs which include other node types such as biological pathways that could generate better quality multi-scale representations. Another direction is the incorporation of advanced chemical and biological features such as molecular fingerprints [24] or information about the secondary and tertiary structure of proteins [57].

REFERENCES

- [1] 2017. DeepCCI: End-to-End Deep Learning for Chemical-Chemical Interaction Prediction, author=Kwon, Sunyoung and Yoon, Sungroh. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. 203–212.
- [2] Sami Abu-El-Haija, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. 2019. Mixhop: Higher-order Graph Convolutional Architectures via Sparsified Neighborhood Mixing. In *International Conference on Machine Learning*. PMLR, 21–29.
- [3] Amr Ahmed, Nino Shervashidze, Shrawan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. 2013. Distributed Large-Scale Natural Graph Factorization. In *Proceedings of the 22nd International Conference on World Wide Web*. 37–48.
- [4] Kathy S Albain, Shona M Nag, German Calderillo-Ruiz, Johann P Jordaan, Antonio C Llombart, Anna Pluzanska, Janusz Rolski, Allen S Melemed, Jose M Reyes-Vidal, Jagdev S Sekhon, et al. 2008. Gemcitabine Plus Paclitaxel versus Paclitaxel Monotherapy in Patients with Metastatic Breast Cancer and Prior Anthracycline Treatment. *Journal of Clinical Oncology* 26, 24 (2008), 3950–3957.
- [5] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Sahand Sharifzadeh, Volker Tresp, and Jens Lehmann. 2021. PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. *Journal of Machine Learning Research* 22, 82 (2021), 1–6.
- [6] Ivana Balazevic, Carl Allen, and Timothy Hospedales. 2019. TuckER: Tensor Factorization for Knowledge Graph Completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 5188–5197.
- [7] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Flores Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. 2018. Relational Inductive Biases, Deep Learning, and Graph Networks. (2018).
- [8] Mikhail V Blagosklonny. 2004. Analysis of FDA Approved Anticancer Drugs Reveals the Future of Cancer Therapy. *Cell Cycle* 3, 8 (2004), 1033–1040.
- [9] Aleksandar Bojchevski, Johannes Klicpera, Bryan Perozzi, Amol Kapoor, Martin Blais, Benedek Rózemberczki, Michal Lukasik, and Stephan Günnemann. 2020. Scaling Graph Neural Networks with Approximate Pagerank. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2464–2473.
- [10] Dan M Bolser, Pierre-Yves Chibon, Nicolas Palopoli, Sungsam Gong, Daniel Jacob, Victoria Dominguez Del Angel, Dan Swan, Sebastian Bassi, Virginia Gonzalez, Prashanth Suravajhala, et al. 2012. MetaBase – the Wiki-Database of Biological Databases. *Nucleic acids research* 40, D1 (2012), D1250–D1254.
- [11] Hannah A. Burkhardt, Devika Subramanian, Justin Mower, and Trevor A. Cohen. 2019. Predicting Adverse Drug-Drug Interactions with Neural Embedding of Semantic Predications. In *AMIA 2019, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 16–20, 2019*. AMIA. <http://knowledge.amia.org/69862-amia-1.4570936/t005-1.4574828/t005-1.4574829/3203351-1.4574920/3203540-1.4574917>
- [12] Chen Cai and Yusu Wang. 2018. A Simple Yet Effective Baseline for Non-Attributed Graph Classification. *CoRR* (2018). arXiv:1811.03508
- [13] Cătălina Cangea, Petar Veličković, Nikola Jovanović, Thomas Kipf, and Pietro Liò. 2018. Towards Sparse Hierarchical Graph Classifiers. *arXiv preprint arXiv:1811.01287* (2018).
- [14] Shaosheng Cao, Wei Lu, and Qionghai Xu. 2015. GraRep: Learning Graph Representations with Global Structural Information. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*. 891–900.
- [15] Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2021. Dual Quaternion Knowledge Graph Embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 6894–6902.
- [16] Deepayan Chakrabarti and Christos Faloutsos. 2006. Graph Mining: Laws, Generators, and Algorithms. *ACM computing surveys (CSUR)* 38, 1 (2006), 2–es.
- [17] Hong Chen and Hisashi Koga. 2019. GL2vec: Graph Embedding Enriched by Line Graphs with Edge Features. In *International Conference on Neural Information Processing*. Springer, 3–14.
- [18] Margaret A Chesney, Jeannette Ickovics, Frederick M Hecht, Godfrey Sikipa, and Judith Rabkin. 1999. Adherence: a Necessity for Successful HIV Combination Therapy. *AIDS* 13 (1999), S271–8.
- [19] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. 2019. Cluster-GCN: An Efficient Algorithm for Training Deep and Large Graph Convolutional Networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 257–266.
- [20] Kuo-Chen Chou. 2001. Prediction of Protein Cellular Attributes Using Pseudo-Amino Acid composition. *Proteins: Structure, Function, and Bioinformatics* 43, 3 (2001), 246–255.
- [21] Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. 2020. Comparative Toxicogenomics Database (CTD): update 2021. *Nucleic Acids Research* 49, D1 (10 2020), D1138–D1143.
- [22] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. 3844–3852.
- [23] D Jeffrey Demanes, Alvaro A Martinez, Michel Ghilezan, Dennis R Hill, Lionel Schour, David Brandt, and Gary Gustafson. 2011. High-dose-rate Monotherapy: Safe and Effective Brachytherapy for Patients with Localized Prostate Cancer. *International Journal of Radiation Oncology* Biology* Physics* 81, 5 (2011), 1286–1292.
- [24] David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. 2015. Convolutional networks on graphs for learning molecular fingerprints. *arXiv preprint arXiv:1509.09292* (2015).
- [25] Richard J Fair and Yitzhak Tor. 2014. Antibiotics and Bacterial Resistance in the 21st Century. *Perspectives in medicinal chemistry* 6 (2014), PMC–S14459.
- [26] Matthias Fey and Jan Eric Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. *arXiv preprint arXiv:1903.02428* (2019).
- [27] Keith T Flaherty, Uma Yasothan, and Peter Kirkpatrick. 2011. Vemurafenib. *Nature Reviews Drug Discovery* 10, 11 (2011), 811–812.
- [28] Alexis Galland and Marc Lelarge. 2019. Invariant Embedding for Graph Classification. In *ICML 2019 Workshop on Learning and Reasoning with Graph-Structured Data*.
- [29] Feng Gao, Guy Wolf, and Matthew Hirn. 2019. Geometric Scattering for Graph Data Analysis. In *International Conference on Machine Learning*. PMLR, 2122–2131.
- [30] Hongyang Gao and Shuiwang Ji. 2019. Graph U-nets. In *international conference on machine learning*. PMLR, 2083–2092.
- [31] Anna Gaulton, Anne Hersey, Michal Nowotka, A. Patrícia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J. Bellis, Elena Cibrián-Uhalte, Mark Davies, Nathan Dedman, Anneli Karlsson, María Paula Magariños, John P. Overington, George Papadatos, Ines Smit, and Andrew R. Leach. 2016. The ChEMBL Database in 2017. *Nucleic Acids Research* 45, D1 (11 2016), D945–D954.
- [32] David Geleta, Andriy Nikolov, Gavin Edwards, Anna Gogleva, Richard Jackson, Erik Jansson, Andrej Lamov, Sebastian Nilsson, Marina Pettersson, Vladimir Poroshin, Benedek Rozemberczki, Timothy Scrivener, Michael Ughetto, and Eliseo Papa. 2021. Biological Insights Knowledge Graph: an Integrated Knowledge Graph to Support Drug Development. (2021).
- [33] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural Message Passing for Quantum Chemistry. In *International Conference on Machine Learning*. PMLR, 1263–1272.
- [34] Xavier Glorot and Yoshua Bengio. 2010. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 249–256.
- [35] Aditya Grover and Jure Leskovec. 2016. Node2Vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 855–864.
- [36] William Leif Hamilton. 2020. Graph Representation Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 14, 3 (2020), 1–159.
- [37] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 1025–1035.
- [38] William Leif Hamilton, Rex Ying, and Jure Leskovec. 2017. Representation Learning on Graphs: Methods and Applications. *IEEE Data Engineering Bulletin* 40, 3 (2017), 52–74.
- [39] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. 2012. RolX: Structural Role Extraction and Mining in Large Graphs. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1231–1239.
- [40] Keith Henderson, Brian Gallagher, Lei Li, Leman Akoglu, Tina Eliassi-Rad, Hanghang Tong, and Christos Faloutsos. 2011. It’s Who You Know: Graph Mining Using Recursive Structural Features. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 663–671.
- [41] Daniel Scott Himmelstein, Antoine Lizee, Christine Hessler, Leo Brueggeman, Sabrina L Chen, Dexter Hadley, Ari Green, Pouya Khankhanian, and Sergio E Baranzini. 2017. Systematic Integration of Biomedical Knowledge Prioritizes Drugs for Repurposing. 6 (sep 2017).
- [42] Susan L Holbeck, Richard Camalier, James A Crowell, Jeevan Prasaad Govindharajulu, Melinda Hollingshead, Lawrence W Anderson, Eric Polley, Larry Rubinstein, Apurva Srivastava, Deborah Wilsker, et al. 2017. The National Cancer Institute ALMANAC: A Comprehensive Screening Resource for the Detection of Anticancer Drug Pairs with Enhanced Therapeutic Activity. *Cancer research* 77, 13 (2017), 3564–3576.

- [43] Kexin Huang, Cao Xiao, Lucas M Glass, Marinka Zitnik, and Jimeng Sun. 2020. SkipGNN: Predicting Molecular Interactions with Skip-Graph Networks. *Scientific reports* 10, 1 (2020), 1–16.
- [44] Aleksandr Ianevski, Anil K Giri, Prson Gautam, Alexander Kononov, Swapnil Potdar, Jani Saarela, Krister Wennerberg, and Tero Aittokallio. 2019. Prediction of Drug Combination Effects with a Minimal Set of Experiments. *Nature machine intelligence* 1, 12 (2019), 568–577.
- [45] Heli Julkunen, Anna Cichonska, Prson Gautam, Sandor Szedmak, Jane Douat, Tapio Pahikkala, Tero Aittokallio, and Juho Rousu. 2020. Leveraging Multi-way Interactions for Systematic Prediction of Pre-clinical Drug Combination Effects. *Nature Communications* 11, 1 (2020), 1–11.
- [46] Travis Kessler and Luis Oliveira. 2019. PadelPy. <https://github.com/ecrl/padelpy>.
- [47] Ayman Khedair, Di Chen, Yogesh Patil, Linan Ma, Q Ping Dou, Malathy PV Shekhar, and Jayanth Panyam. 2010. Nanoparticle-Mediated Combination Chemotherapy and Photodynamic Therapy Overcomes Tumor Drug Resistance. *Journal of Controlled Release* 141, 2 (2010), 137–144.
- [48] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).
- [49] Thomas N. Kipf and Max Welling. [n.d.]. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.
- [50] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [51] Johannes Klicpera, Aleksandar Bojchevski, and Stephan Günnemann. [n.d.]. Predict then Propagate: Graph Neural Networks meet Personalized PageRank. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.
- [52] Boris Knyazev, Graham W Taylor, and Mohamed Amer. 2019. Understanding Attention and Generalization in Graph Neural Networks. *Advances in Neural Information Processing Systems* 32 (2019), 4202–4212.
- [53] Brent M. Kuenzi, Jisoo Park, Samson H. Fong, Kyle S. Sanchez, John Lee, Jason F. Kreisberg, Jianzhu Ma, and Trey Ideker. 2020. Predicting Drug Response and Synergy Using a Deep Learning Model of Human Cancer Cells. *Cancer Cell* 38, 5 (2020), 672–684.e6.
- [54] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. 2019. Self-Attention Graph Pooling. In *International Conference on Machine Learning*. PMLR, 3734–3743.
- [55] Joseph Lehár, Andrew S Krueger, William Avery, Adrian M Heilbut, Lisa M Johansen, E Roydon Price, Richard J Rickles, Glenn F Short Iii, Jane E Staunton, Xiaowei Jin, et al. 2009. Synergistic Drug Combinations Tend to Improve Therapeutically Relevant Selectivity. *Nature biotechnology* 27, 7 (2009), 659–666.
- [56] Hui Liu, Wenhao Zhang, Bo Zou, Jinxian Wang, Yuanyuan Deng, and Lei Deng. 2020. DrugCombDB: A Comprehensive Database of Drug Combinations Toward the Discovery of Combinatorial Therapy. *Nucleic acids research* 48, D1 (2020), D871–D881.
- [57] Yang Liu, Qing Ye, Liwei Wang, and Jian Peng. 2018. Learning Structural Motif Representations for Efficient Protein Structure Search. *Bioinformatics* 34, 17 (2018), i773–i780.
- [58] Reza Bayat Mokhtari, Tina S Homayouni, Narges Baluch, Evgeniya Morgatskaya, Sushil Kumar, Bikul Das, and Herman Yeger. 2017. Combination Therapy in Combating Cancer. *Oncotarget* 8, 23 (2017), 38022.
- [59] Vinod Nair and Geoffrey E Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 807–814.
- [60] Annamalai Narayanan, Mahinthan Chandramohan, Lihui Rajasekar Venkatesan, Yang Liu Chen, and Shantanu Jaiswal. [n.d.]. Graph2Vec: Learning Distributed Representations of Graphs. ([n.d.]).
- [61] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A Three-Way Model for Collective Learning on Multi-Relational Data. In *International Conference on Machine Learning*. PMLR.
- [62] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. 2016. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1105–1114.
- [63] Ann H Partridge, Harold J Burstein, and Eric P Winer. 2001. Side Effects of Chemotherapy and Combined Chemohormonal Therapy in Women with Early-Stage Breast Cancer. *JNCI Monographs* 2001, 30 (2001), 135–142.
- [64] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems* 32 (2019), 8026–8037.
- [65] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
- [66] Pleuni S Pennings. 2013. HIV Drug Resistance: Problems and Perspectives. *Infectious disease reports* 5, 11 (2013), 21–25.
- [67] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 701–710.
- [68] Bryan Perozzi, Vivek Kulkarni, Haochen Chen, and Steven Skiena. 2017. Don't Walk, Skip! Online Learning of Multi-Scale Network Embeddings. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, 258–265.
- [69] Kristina Preuer, Richard PI Lewis, Sepp Hochreiter, Andreas Bender, Krishna C Bulusu, and Günter Klambauer. 2018. DeepSynergy: Predicting Anti-Cancer Drug Synergy with Deep Learning. *Bioinformatics* 34, 9 (2018), 1538–1546.
- [70] Jiezhong Qiu, Yuxiao Dong, Hao Ma, Jian Li, Kuansan Wang, and Jie Tang. 2018. Network Embedding as Matrix Factorization: Unifying Deepwalk, LINE, PTE, and Node2Vec. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, 459–467.
- [71] Leonardo FR Ribeiro, Pedro HP Saverese, and Daniel R Figueiredo. 2017. Struc2Vec: Learning Node Representations from Structural Identity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 385–394.
- [72] Ryan A Rossi, Di Jin, Sungchul Kim, Nesreen K Ahmed, Danai Koutra, and John Boaz Lee. 2020. On Proximity and Structural Role-Based Embeddings in Networks: Misconceptions, Techniques, and Applications. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 14, 5 (2020), 1–37.
- [73] Benedek Rozemberczki, Peter Englert, Amol Kapoor, Martin Blais, and Bryan Perozzi. 2021. Pathfinder Discovery Networks for Neural Message Passing. [arXiv:2010.12878](https://arxiv.org/abs/2010.12878) [cs.LG]
- [74] Benedek Rozemberczki, Oliver Kiss, and Rik Sarkar. 2020. Karate Club: An API Oriented Open-source Python Framework for Unsupervised Learning on Graphs. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 3125–3132.
- [75] Benedek Rozemberczki and Rik Sarkar. 2018. Fast Sequence-Based Embedding with Diffusion Graphs. In *International Workshop on Complex Networks*. Springer, 99–107.
- [76] Benedek Rozemberczki and Rik Sarkar. 2020. Characteristic Functions on Graphs: Birds of a Feather, from Statistical Descriptors to Parametric Models. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, 1325–1334.
- [77] Jae Yong Ryu, Hyun Uk Kim, and Sang Yup Lee. 2018. Deep Learning Improves Prediction of Drug–Drug and Drug–Food Interactions. *Proceedings of the National Academy of Sciences* 115, 18 (2018), E4304–E4311.
- [78] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* 15, 1 (2014).
- [79] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-Scale Information Network Embedding. In *Proceedings of the 24th ACM International Conference on World Wide Web*, 1067–1077.
- [80] GE Tao. 2011. Pharmaprojects Web Database and Its Application. *Progress in Pharmaceutical Sciences* (2011), 12.
- [81] T Trouillon, CR Dance, E Gaussier, J Welbl, S Riedel, and G Bouchard. 2017. Knowledge Graph Completion via Complex Tensor Factorization. *Journal of Machine Learning Research* 18, 130 (2017), 1–38.
- [82] Anton Tsitsulin, Davide Mottin, Panagiotis Karras, Alexander Bronstein, and Emmanuel Müller. 2018. NetLSD: Hearing the Shape of a Graph. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2347–2356.
- [83] Anton Tsitsulin, Davide Mottin, Panagiotis Karras, and Emmanuel Muller. 2018. VERSE: Versatile Graph Embeddings from Similarity Measures. In *Proceedings of the 2018 World Wide Web Conference*, 539–548.
- [84] Anton Tsitsulin, Marina Munkhoeva, and Bryan Perozzi. 2020. Just SLAQ When You Approximate: Accurate Spectral Distances for Web-Scale Graphs. In *Proceedings of The Web Conference 2020*, 2697–2703.
- [85] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing Data Using t-SNE. *Journal of Machine Learning Research* 9, 11 (2008).
- [86] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2019. Composition-based Multi-Relational Graph Convolutional Networks. In *International Conference on Learning Representations*.
- [87] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. [n.d.]. Graph Attention Networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.
- [88] Saurabh Verma and Zhi-Li Zhang. 2017. Hunt for the Unique, Stable, Sparse and Fast Feature Learning on Graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 87–97.
- [89] Martin Wattenberg, Fernanda Viégas, and Ian Johnson. 2016. How to Use t-SNE Effectively. *Distill* 1, 10 (2016), e2.
- [90] David Weininger. 1988. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *Journal of Chemical*

- information and computer sciences* 28, 1 (1988), 31–36.
- [91] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying Graph Convolutional Networks. In *International conference on machine learning*. PMLR, 6861–6871.
- [92] Nan Xiao, Dong-Sheng Cao, Min-Feng Zhu, and Qing-Song Xu. 2015. protr/ProtrWeb: R Package and Web Server for Generating Various Numerical Representation Schemes of Protein Sequences. *Bioinformatics* 31, 11 (2015), 1857–1859. <https://doi.org/10.1093/bioinformatics/btv042>
- [93] Zhiwen Xie, Guangyou Zhou, Jin Liu, and Xiangji Huang. 2020. ReInceptionE: Relation-Aware Inception Network with Joint Local-Global Structural Information for Knowledge Graph Embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 5929–5939.
- [94] Hao Xu, Shengqi Sang, and Haiping Lu. 2020. Tri-Graph Information Propagation for Polypharmacy Side Effect Prediction. *CoRR* abs/2001.10516 (2020). arXiv:2001.10516 <https://arxiv.org/abs/2001.10516>
- [95] Nuo Xu, Pinghui Wang, Long Chen, Jing Tao, and Junzhou Zhao. 2019. MR-GNN: Multi-Resolution and Dual Graph Neural Network for Predicting Structured Entity Interactions. *Proceedings of IJCAI* (2019).
- [96] Wentao Xu, Shun Zheng, Liang He, Bin Shao, Jian Yin, and Tie-Yan Liu. 2020. SEEK: Segmented Embedding of Knowledge Graphs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 3888–3897.
- [97] Chun Wei Yap. 2011. PaDEL-Descriptor: An Open Source Software to Calculate Molecular Descriptors and Fingerprints. *Journal of computational chemistry* 32, 7 (2011), 1466–1474.
- [98] Timothy A Yap, Aurelius Omlin, and Johann S De Bono. 2013. Development of Therapeutic Combinations Targeting Major Cancer Signaling Pathways. *Journal of Clinical Oncology* 31, 12 (2013), 1592–1605.
- [99] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 974–983.
- [100] Rex Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L Hamilton, and Jure Leskovec. 2018. Hierarchical Graph Representation Learning with Differentiable Pooling. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. 4805–4815.
- [101] Yue Yu, Kexin Huang, Chao Zhang, Lucas M Glass, Jimeng Sun, and Cao Xiao. 2021. SumGNN: Multi-Typed Drug Interaction Prediction via Efficient Knowledge Graph Summarization. *Bioinformatics* (2021).
- [102] Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. 2018. An End-to-End Deep Learning Architecture for Graph Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
- [103] Marinka Zitnik, Monica Agrawal, and Jure Leskovec. 2018. Modeling Polypharmacy Side Effects with Graph Convolutional Networks. *Bioinformatics* 34, 13 (06 2018), i457–i466.