# Structured Multi-View Representations for Drug Combinations

**Shengchao Liu**[1,2]*, **Andreea Deac**[1,2]*, **Zhaocheng Zhu**[1,2], **Jian Tang**[1,3]
[1] Mila-Quebec AI Institute, [2] Université de Montréal, [3] HEC Montreal

## Abstract

A central task in computational medicine is discovering novel effective treatments in the form of drug combinations. This is necessary for complex medical conditions and is difficult to screen empirically. So far, computational approaches have focused on using either an *chemical* view of the drug (the chemical structure) or an *protein* view of the drug (the proteins each drug is affecting). Conversely, we introduce a multi-view framework which leverages information from the drugs' functional groups, while also matching the sets of proteins they target by using Graph Neural Networks. We empirically prove the benefit of our proposed multi-view solution, reaching state-of-the-art performance.

## 1 Introduction and Background

Growing amounts of data and better computational paradigms have allowed increasing use of neural networks for computational medicine [37]. However, treating co-existing conditions and complex diseases requires the analysis of *drug combinations*, a field still largely unexplored. Drug combinations refer to simultaneous administration of multiple drugs, and computational approaches can provide huge benefits – they can indicate novel treatments at a very low cost and short time frame.

Leveraging the drug chemical and the target protein graph at the same time would be equivalent to providing two views for each drug considered – an *chemical* view, derived from its chemical structure, and an *protein* view, based on the graph of proteins it targets.[1] Markedly, *joining the drug chemical and the target protein graph* in a multi-view framework is, to the best of our knowledge, a direction yet unstudied. While there exist some recent works where one of the views is a graph [2, 26], the full potential of the data is not explored – the graph is summarized to a flat representation before exchanging information with the other view. This observation is even more relevant when leveraging graphs of proteins – drugs not only affect the target proteins, but also the other proteins which are on the same pathways; thus the graph of proteins a drug affects is often a dense graph. This leads to our proposed framework, the Structured Multi-View Representation (SMVI). Next we will start with some notations before further discussion.

**Notations** Given a drug combination task, the $i$-th drug has two views, *i.e.*, $x^i = (x_c^i, x_p^i)$, where $x_c$ and $x_p$ correspond to the Chemical View and Protein View respectively. Each view has one corresponding representation function ($g_c$, $g_p$) and one prediction function ($f_c$, $f_p$).

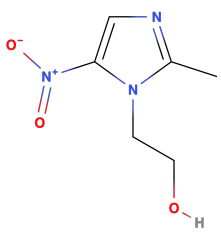
### 1.1 Single Views-Based Prediction

**Chemical View** In this view, the drug representation encodes the chemical feature from that drug. The common way is to treat drug as a molecular graph, with atoms as nodes and bonds as edges.

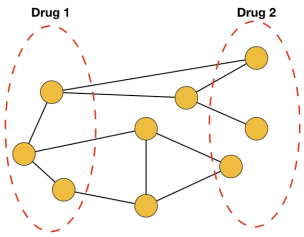
*Equal contribution

[1] It can include richer types of biological entities, like disease, assay, gene, etc. Here we only consider protein for simplicity.

Machine Learning for Molecules Workshop at NeurIPS 2020. https://ml4molecules.github.io




(a) Chemical View




(b) Protein View

Figure 1: Each drug can be represented by the chemical feature as Chemical View. While in Protein View, each drug can be represented as a graph from the large PPI graph, as marked in the dashed line.

Recent works [31, 34, 28] explored different representation strategies, and here we may as well take Extended Connectivity FingerPrints (ECFP) [31] which maps the molecular graph into a fix-length bit vector.

**Protein View** Protein View focuses more on the relation between drug and its connected proteins, and therefore it can reveal the pharmaceutical aspect for each drug, as a higher-level information. In doing so, a biomedical knowledge graph is used for exploring the pharmaceutical relation. Here we simplify it as a single knowledge graph: if two proteins can have an interaction, we add a link between them and call it as protein-protein interaction (PPI). Thus, each drug induces its own graph of the PPI network over the protein set it targets. We denote $x_p = (V, E)$ the induced graph of proteins, which will further be used as an input to representation function $g_p$, to obtain a latent representation of the view, $g_p(x_p) \in \mathbb{R}^{d_p}$. More details are followed in Section 1.2.

**Drug Pair Prediction** Given Chemical View and Protein View $(g_c(x_c), g_p(x_p))$, we can perform the single-view final prediction on the drug-pair $(x_i, x_j)$. Following the common practice in matching networks [9], we predict the pairwise combination on the concatenation of the two drug representations, *i.e.*, $[g_c(x_c^i) \oplus g_c(x_c^j)]$ or $[g_p(x_p^i) \oplus g_p(x_p^j)]$, depending on the view we are using, where $\oplus$ is the concatenation.

### 1.2 Node-Level and Graph-Level Representation of the Protein View

In Protein View, each drug induces a graph $x_p = (V, E)$, with $V = \{p_1, p_2, \dots p_{|V|}\}$, the proteins targeted by the drug and $E = \{(p_i, p_j) | p_i, p_j \in V \wedge p_i \leftrightarrow p_j\}$, edges between these proteins, where $p_i \leftrightarrow p_j$ represents an existing link in the PPI network. We denote by $\mathcal{N}_i$ the one-hop neighborhood of protein $p_i$ in this subgraph. Each protein is represented by a learnable embedding vector $p_i^0 \in \mathbb{R}^{d_p}$. The graph structure then allows us to apply a GCN [24], which updates the protein representation as follows:

$$p_i^{t+1} = \sigma\Big(W_s^t p_i^t + \sum_{(i,j)\in E} \frac{1}{\sqrt{|\mathcal{N}_i||\mathcal{N}_j|}} W^t p_j^t\Big) \qquad g_p(x_p) = \sum_{i=1}^{|V|} p_i^T \tag{1}$$

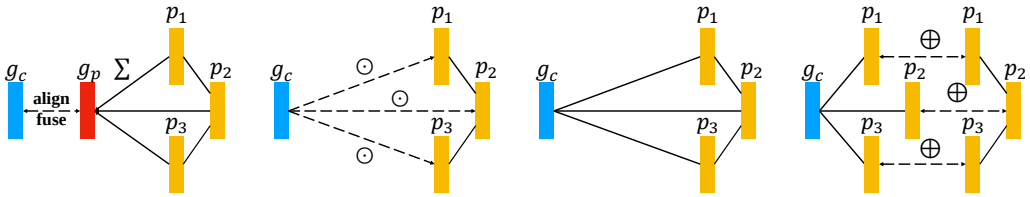
where $\sigma$ is an activation function and $W$ and $W_s$ are a learnable linear transformations
After $T$ GCN layers, we summarise the protein representations through a graph-level readout function, resulting in $g_p(x_p)$, the representation for drug in Protein View.

### 1.3 Unstructured Multi-View Interaction

Classic multi-view representation methods first extract a flat representation from each view ($g_c$ and $g_p$ in our framework), then try to align or fuse these multiple representations. Deep multi-view representation includes aligning features through correlation [19, 3, 41] or fusing the representation learned by neural networks [32, 22, 12]. Recent progress is contrastively learning the multi-view representation [5, 40] in a self-supervised manner. To put them into our setting, these methods are indeed jointly learning the representation between $g_c(x_c)$ and $g_p(x_p)$, while they are unaware of the drug-protein graph. Thus we categorize them as unstructured multi-view interaction methods.

## 2 Structured Multi-View Interaction

As pointed out in Section 1.2, Protein View is a graph of drug and proteins. To be more specific, existing multi-view learning methods are focusing on interactions between $g_c(x_c)$ and $g_p(x_p)$, and ignore that $g_p(x_p)$ can be factorized into proteins the drug targets, *i.e.*, $p_i^T$. Conversely, we are interested in utilizing such graph structure to allow $g_c(x_c)$ and $p_i^T$ to interact. Motivated by this, we design three structured approaches for modeling their interactions, described in Sections 2.1 to 2.3. The high-level pipelines are given in Figure 2.




(a) UMVI (b) CPA (c) MPMP (d) DPMP

Figure 2: Pipelines for UMVI and three SMVI pipelines. The blue, red, and yellow rectangles are drug representations from Chemical View ($g_c(x_c)$), Protein View ($g_p(x_p)$) and protein representations from Protein View ($p_i$) respectively. $\sum$, $\odot$ and $\oplus$ are summation, element-wise dot product, and concatenation. Solid line corresponds to an edge, while dashed line means it is a directed operation. In Figure 2a, we illustrate UMVI between $x_c$ and $x_p$, with feature alignment or fusion. In Figures 2b to 2d, we visualise the three proposed structured interaction methods CPA, MPMP and DPMP.

### 2.1 Context-Protein Attention (CPA)

One way to do structured multi-view learning is by providing the Chemical View as a context for the Protein View representation learning. We call this Context-Protein Attention (CPA).

Recall that $g_p(x_p) = \sum_{i=1}^{|V|} p_i^T$. We provide Chemical View context through attention [6], providing the Chemical View $g_c(x_c)$ as the query, and the protein embeddings $p_i^T$ from Protein View as the keys. We learn coefficient vectors $\alpha_i$ for each drug-protein pair, $\alpha_i = s(x_c, p_i)$, where $s$ is the attention mechanism, for which we tested (1) element-wise product: $s(x_c, p_i) = g_c(x_c) \odot p_i^T$ and (2) linear transformation on concatenation: $s(x_c, p_i) = W[g_c(x_c) \oplus p_i]$, where $\odot$ is an element-wise product of two vectors and $\oplus$ is vector concatenation. With this learned weight vector, the final sub-component and Protein View representation are given in Equation (2).

$$\tilde{g}_p(p_i) = \alpha_i \odot p_i^T \qquad \tilde{g}(x_p) = \sum_{i=1}^{|V|} \tilde{g}_p(p_i) \tag{2}$$

### 2.2 Master-Protein Message Passing (MPMP)

In Master-Protein Message Passing (MPMP), the Chemical View is added as a master node to the Protein View graph, which is directly connected with all proteins. Combining the information from the protein-protein (PPI) network and the drug-protein interaction (DPI) network, we create a master-protein graph $x_{mp} = (V_{mp}, E_{mp})$, where $V_{mp} = V \cup \{x_c\}$ and $E_{mp} = E \cup \{(x_c, p_i) | p_i \in V\}$. The update rule is:

$$v_i^{t+1} = \sigma\Big(W_s^t v_i^t + \sum_{(i,j)\in E_{mp}} \frac{1}{\sqrt{|\mathcal{N}_i||\mathcal{N}_j|}} W^t v_j^t\Big) \tag{3}$$

where $v_i^{i+1}$ is the representation after applying layer $t+1$ of node $v$ (which is either $p_i^t$ or $x_c$). The initial features of $x_c$ within $V_{mp}$ are given by $g_c(x_c)$. This enables direct propagation of information from the chemical to the individual proteins through the GNN update and, also allows the master node to summarise the protein information and adapt at the same time.

### 2.3 Drug Protein Message Passing (DPMP)

We explicitly incorporate heterogeneity in the Drug Protein Message Passing (DPMP) model. DPMP learns two different sets of parameters in order to distinctly model protein-protein and drug-protein

interactions. We use two graphs to model the two types of interactions – drug-protein (DP graph) and protein-protein (the previously introduced $x_p$ graph). We denote $G_{DP} = (V_{DP}, E_{DP})$, the graph modelling the drug-protein interactions, with $V_{DP} = V \cup \{x_c\}$ and $E_{DP} = \{(x_c, p_i) | p_i \in V_{DP}\}$, while the graph $x_p$, described in Section 1.2, models the protein-protein interactions. This yields two protein representations: 1) $\tilde{p}_i$ from its interaction with the chemical, as described in Equation (4); 2) $p_i$ from interaction with proteins in $x_p$, described in Equation (1). The update rule in the DP graph is:

$$\tilde{p}_i^{t+1} = \sigma\Big({}^{dp}W_s^t \tilde{p}_i^t + \frac{1}{\sqrt{|V|}} {}^{dp}W^t \tilde{x}_c^t\Big) \qquad \tilde{x}_c^{t+1} = \sigma\Big({}^{dp}W_s^t \tilde{x}_c^t + \sum_{i \in V} \frac{1}{\sqrt{|V|}} {}^{dp}W^t \tilde{p}_i^t\Big) \tag{4}$$

where the mixing coefficient is determined by the fact that the protein neighborhood size is 1 ($|\mathcal{N}_i| = 1$) and the drug neighborhood size is the number of proteins ($|\mathcal{N}_j| = |V|$). Similarly to MPMP, the initial drug chemical embedding $\tilde{x}_c^0$ is obtained from $g_c(x_c)$.

For the PP graph, similarly as before:

$$p_i^{t+1} = \sigma\Big({}^{pp}W_s^t p_i^t + \sum_{(i,j) \in E} \frac{1}{\sqrt{|\mathcal{N}_i||\mathcal{N}_j|}} {}^{pp}W^t p_j^t\Big) \tag{5}$$

Critically, the $pp$ weights are distinct from the $dp$ weights. Note that the MPMP model can be seen as a special case of the DPMP model, when we force ${}^{dp}W_s^t = {}^{pp}W_s^t$ and ${}^{dp}W^t = {}^{pp}W^t$ for all $t$.

One layer is then defined by performing one GCN propagation in the drug-protein graph and one GCN propagation in the protein-protein graph in parallel. This is then followed by the fusion of $\tilde{p}_i$ and $p_i$ through a linear layer: $\hat{p}_i^{t+1} = W[\tilde{p}_i^{t+1} \oplus p_i^{t+1}]$, where $\oplus$ is the concatenation on vectors. At the end of the GCN layer, we reassign the protein representation as: $\tilde{p}_i^{t+1} = p_i^{t+1} = \hat{p}_i^{t+1}$.

## 3 Experiments

We only list the brief descriptions of baselines here. More details can be found in appendix. For **Chemical View**, we apply Extended Connectivity FingerPrints (ECFP). For **Protein View**, we use Graph Convolutional Network (GCN) [24] on the PPI knowledge graph. For **UMVI**, we include Simple Mean, Deep CCA [3], Deep Fusion [27], and InfoNCE [40]. For **SMVI**, our proposed methods include Context-Protein Attention (CPA), Master-Protein Message Passing (MPMP), Drug Protein Message Passing (DPMP).

Table 1: 5-fold cross-validation results on NCI-SS and NCI-GP. The best results are marked in **bold**.

| View | | Model | NCI-SS | | NCI-GP | |
|---|---|---|---|---|---|---|
| | | | RMSE | MAE | RMSE | MAE |
| Chemical View | | ECFP | 27.11 ± 1.58 | 15.22 ± 0.88 | 28.00 ± 3.93 | 17.24 ± 1.86 |
| Protein View | | Local PPI | 25.95 ± 1.04 | 15.41 ± 0.43 | 27.37 ± 3.64 | 16.65 ± 1.74 |
| Chemical View & Protein View | UMVI | Simple Mean | 26.19 ± 1.74 | 15.30 ± 0.71 | 27.12 ± 3.17 | 16.99 ± 1.33 |
| | | Deep CCA | 31.03 ± 1.22 | 22.32 ± 0.73 | 31.47 ± 3.22 | 22.23 ± 2.59 |
| | | Deep Fusion | 26.55 ± 2.22 | 15.33 ± 0.81 | 26.50 ± 3.12 | 17.00 ± 1.19 |
| | | InfoNCE | 26.61 ± 1.61 | 15.69 ± 0.68 | 27.97 ± 3.33 | 16.88 ± 1.30 |
| | SMVI | CPA | **24.42 ± 1.13** | **13.70 ± 0.33** | **25.94 ± 4.51** | **15.34 ± 1.56** |
| | | MPMP | 25.06 ± 2.33 | 14.89 ± 0.64 | 27.11 ± 2.99 | 16.66 ± 0.77 |
| | | DPMP | 24.79 ± 1.87 | 15.26 ± 0.81 | 26.39 ± 3.93 | 16.60 ± 1.33 |

We can observe that generally, all three SMVI methods can perform better than the baselines. Specifically, CPA is reaching consistently state-of-the-art performance on both datasets. Due to its indirect nature, CPAis the model requiring the smallest number of learnable parameters out of the three, making it a better candidate for small datasets.

## 4 Conclusions

To sum up, we propose Structured Multi-View Interaction (SMVI), a novel and general framework for learning multi-view representations from a Chemical View and Protein View. We introduce three approaches for leveraging the interaction between the two views and all demonstrate better performance. Moreover, the three structured approaches outperform UMVI baselines which combine or contrast the two views after having a flat representation.

## References


[1] PubChem Identifier Exchange Service, 2020.

[2] Mona Alshahrani and Robert Hoehndorf. Drug repurposing through joint learning on knowledge graphs and literature. 2018.

[3] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255, 2013.

[4] Masaki Asada, Makoto Miwa, and Yutaka Sasaki. Enhancing drug-drug interaction extraction from texts by molecular structure information. *arXiv preprint arXiv:1805.05593*, 2018.

[5] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems 32*, pages 15535–15545. Curran Associates, Inc., 2019.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[7] David M Blei and Michael I Jordan. Modeling annotated data. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 127–134, 2003.

[8] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100, 1998.

[9] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. In *Advances in neural information processing systems*, pages 737–744, 1994.

[10] Feixiong Cheng, István A Kovács, and Albert-László Barabási. Network-based prediction of drug combinations. *Nature communications*, 10(1):1–11, 2019.

[11] Andreea Deac, Yu-Hsiang Huang, Petar Veličković, Pietro Liò, and Jian Tang. Drug-drug adverse effect prediction with graph co-attention. *arXiv preprint arXiv:1905.00534*, 2019.

[12] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

[13] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013.

[14] Assaf Gottlieb, Gideon Y Stein, Yoram Oron, Eytan Ruppin, and Roded Sharan. Indi: a computational framework for inferring drug interactions and their associated recommendations. *Molecular systems biology*, 8(1):592, 2012.

[15] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[17] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[18] 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. The national cancer institute almanac: a comprehensive screening resource for the detection of anticancer drug pairs with enhanced therapeutic activity. *Cancer research*, 77(13):3564–3576, 2017.

[19] Harold Hotelling. Relations between two sets of variates. In *Breakthroughs in statistics*, pages 162–190. Springer, 1992.

[20] Hiroaki Iwata, Ryusuke Sawada, Sayaka Mizutani, Masaaki Kotera, and Yoshihiro Yamanishi. Large-scale prediction of beneficial drug combinations using drug efficacy and target profiles. *Journal of chemical information and modeling*, 55(12):2705–2716, 2015.

[21] Yangqing Jia, Mathieu Salzmann, and Trevor Darrell. Factorized latent spaces with structured sparsity. In *Advances in Neural Information Processing Systems*, pages 982–990, 2010.

[22] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.

[23] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem 2019 update: improved access to chemical data. *Nucleic acids research*, 47(D1):D1102–D1109, 2019.

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

[25] Dongge Li, Nevenka Dimitrova, Mingkun Li, and Ishwar K Sethi. Multimedia content processing through cross-modal association. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 604–611, 2003.

[26] Jia Li, Yu Rong, Hong Cheng, Helen Meng, Wenbing Huang, and Junzhou Huang. Semi-supervised graph classification: A hierarchical graph perspective. In *The World Wide Web Conference*, pages 972–982, 2019.

[27] Yingming Li, Ming Yang, and Zhongfei Zhang. A survey of multi-view representation learning. *IEEE transactions on knowledge and data engineering*, 31(10):1863–1883, 2018.

[28] Shengchao Liu, Mehmet F Demirel, and Yingyu Liang. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. pages 8464–8476, 2019.

[29] Tengfei Ma, Cao Xiao, Jiayu Zhou, and Fei Wang. Drug similarity integration through attentive multi-view graph auto-encoders. *arXiv preprint arXiv:1804.10850*, 2018.

[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013.

[31] HL Morgan. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of Chemical Documentation*, 5(2):107–113, 1965.

[32] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. 2011.

[33] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[34] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 991–1001. Curran Associates, Inc., 2017.

[35] Guy Shtar, Lior Rokach, and Bracha Shapira. Detecting drug-drug interactions using artificial neural networks and classic graph similarity measures. *PloS one*, 14(8), 2019.

[36] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 650–658, 2008.

[37] Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackerman, et al. A deep learning approach to antibiotic discovery. *Cell*, 180(4):688–702, 2020.

[38] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1):D607–D613, 2019.

[39] Damian Szklarczyk, Alberto Santos, Christian von Mering, Lars Juhl Jensen, Peer Bork, and Michael Kuhn. Stitch 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic acids research*, 44(D1):D380–D384, 2016.

[40] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

[41] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning. In *International Conference on Machine Learning*, pages 1083–1092, 2015.

[42] David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 46(D1):D1074–D1082, 2018.

[43] Fangfang Xia, Maulik Shukla, Thomas Brettin, Cristina Garcia-Cardona, Judith Cohn, Jonathan E Allen, Sergei Maslov, Susan L Holbeck, James H Doroshow, Yvonne A Evrard, et al. Predicting tumor cell line response to drug pairs with deep learning. *BMC bioinformatics*, 19(18):71–79, 2018.

[44] Eric P Xing, Rong Yan, and Alexander G Hauptmann. Mining associated text and images with dual-wing harmoniums. *arXiv preprint arXiv:1207.1423*, 2012.

[45] Marinka Zitnik, Monica Agrawal, and Jure Leskovec. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, 34(13):i457–i466, 2018.

# A Overview of Direct and Indirect Drug-Protein Interaction

Intuitively, there is a distinct difference between the CPA model and the remainder: the final obtained Protein View representation, $g_p(x_p)$, does not directly incorporate any content from the Chemical View, $g_c(x_c)$. Instead, CPA uses $g_c(x_c)$ as a mechanism for *indirectly* specifying the way in which contributions from various elements within $g_p(x_p)$ are scaled—hence making the final representation a weighted sum of *pure* protein representations, $p_i^T$. The only way in which the protein embeddings have an effect on (subsequent) chemical embedding computations is through backpropagating through the attention weights, $\alpha_i$.

In comparison, the MPMP and DPMP methods directly include $g_c(x_c)$ as a component of the message passing system, causing the protein representations $p_i^t$ to directly exchange content with corresponding Chemical View representations, $g_c(x_c)^t$ (which are themselves updated within this process). This allows the chemical information to *directly* influence the final Protein View vector.

Finally, we use the above analysis to propose another categorisation of multi-view interaction: *indirect* (influenced but not content-wise; only through backprop, such as CPA) and *direct* (exchanging content directly during forward propagation, such as MPMP and DPMP).

# B Related Work

## B.1 Drug Combination

Drug combination methods can be divided into two categories: chemical feature as Chemical View and network-based method as Protein View.

**Chemical-Based Method** An alternative method to DDI prediction is to directly study the similarity of the drugs. The work from Asada et al. [4] represents each drug separately using a textual and a GNN embedding and then predicts the side-effect after concatenating them. The multi-head co-attentive drug-drug interaction (MHCADDI) [11] model explicitly accounts for the interactions between molecular substructures of the two drugs at different scales, through an interleaved sequence of GNN layers within each drug's molecular graph and co-attentive layers that allow atom representations to be exchanged between the two drugs.

**Network-Based Method** Representing drugs as nodes in a graph allows identification of large overlap between two drugs' sets of neighbors, indicating positive or adverse drug-drug interactions (DDI). Therefore, it has been common to pose the task as a link prediction problem and use methods such as matrix factorization [35]. More recently, GNNs have been applied: AttSemiGAE [29] leverages multi-view graph auto-encoders, where each view corresponds to a different type of drug features, and a attentive mechanism to predict the weights corresponding to each view. Instead of multiple views, Decagon [45] uses a multi-modal graph by adding proteins as nodes in the graph.

## B.2 Multi-View Learning

Li et al. [27] concludes that the classic multi-view representation learning can be classified as feature alignment and feature fusion. For the first one, distance-based alignment [25] and similarity-based alignment [13, 22] target at minimizing the distance and maximizing the similarity among views from some measurable space. While correlation-based method like Canonical Correlation Analysis (CCA) [19] attempts to maximize the correlation between multiple views of representation. Deep CCA [3, 41] extends CCA from linear projection to non-linear mapping. For feature fusion, graphical model-based method [7, 21, 44, 36] tries to learn a distribution based on the observed multi-view data. Another direction is to use neural network [32, 22, 12], which first learns the representation from each view then aggregated for the final prediction. And specific to the drug discovery, some existing works [14, 20] can belong to the fusion methods, and put more emphasis on feature extraction from various sources.

Recently, contrastive learning [33, 17] has become prevalent in self-supervised learning. It encourages the representations getting close for the same data point while more distant to different data points in a contrastive way. Borrowing this idea to multi-view setting, [5, 40] apply the contrastive learning to learn representations among multiple views and reach promising results.

Another thing to mention is that the mainstream methods for multi-view learning include co-training [8] and knowledge distillation [16]. They can handle the multi-view (decision) learning, but they transfer knowledge among different views in the output level (or the decision layer). Thus they are beyond the scope of this paper.

**Multi-View Learning on Graph** Only a few works have been focusing on this direction. [2] takes the knowledge graph and literature description as two views and jointly learn the node representation under the word2vec [30] framework. [26] explores both the knowledge graph and entity-level graph, and proposes using the KL-divergence as the disagreement measure to improve the model performance. We will introduce a framework that can better utilize the property of the graph.

# C Dataset Specification

The National Cancer Institute (NCI)-ALMANAC [18] provides Food and Drug Administration (FDA)-approved drug pairs on killing the cancer cells. It also includes the effect on various cell lines and panels, and here we are focusing on only one of them, the CCRF-CEM (cell lines) and Leukemia (panel). Besides, in NCI-ALMANAC, each drug pair and cell line corresponds to multiple records with different drug concentration combinations. Following [43], we are taking only the best growth inhibition for different concentration combinations, *i.e.*, the lowest growth fraction for each drug pair and cell line.

To align the NCI data with STITCH [39], STRING [38], and Cheng et al [10], the key is to map using the drug. NCI is using Cancer Chemotherapy National Service Center number (NCS ID) as drug identifier, and mapping it to the PubChem ID [23] takes the following steps:

1. We get all the valid drugs NSC ID from file 'ComboDrugGrowth_Nov2017.csv', removing NaN scores.
2. Then we map NSC ID to PubChem SID using 'NSC_PubChemSID.csv', then convert to PubChem CID using the PubChem Exchange website [1]. Only one PubChem SID (01178), cannot find PubChem CID. Note that we have some drugs with one PubChem SID mapping to multiple PubChem CIDs.
3. Each drug can have up to two drug names, one from file 'ComboCompoundNames_small.txt' and one by using NSC ID as drug name.[2] Then we map the drug names to PubChem CID using pubchempy.
4. Finally we merge all the drugs following NSC ID -> PubChem SID -> PubChem CID.
   (a) If we have only one mapping from PubChem SID -> PubChem CID, use this; otherwise:
   (b) If we have multiple mappings from PubChem SID -> PubChem CID, choose the most frequent CID in Drug Name -> CID; otherwise:
   (c) Prioritize the drug name mapping to PubChem CID using NSC ID to PubChem CID.

So now we have the mapping from NSC ID to PubChem CID for NCI-SS. Then we also use the PubChem Exchange website to map from PubChem CID to DrugBank ID [42] for NCI-GP.

## C.1 Statistics

Table 2: Dataset specification on merged NCI-ALMANAC with two sources of DPI and PPI.

| Dataset | # Drug | # Drug-Drug | # Drug-Protein | # Protein | # Protein-Protein |
|---|---|---|---|---|---|
| NCI-SS | 67 | 2,160 | 33,042 | 19,354 | 11,759,454 |
| NCI-GP | 46 | 979 | 735 | 2,132 | 217,160 |

We consider the drug combination effect on the cancer cell from National Cancer Institute (NCI) [18]. NCI-ALMANAC measure the best growth inhibition of each FDA-approved drug pair on killing the cancer cells, which can be viewed as a regression task. We merge the NCI-ALMANAC data with two sources of drug-proteins and protein-protein interaction network. Table 2 shows the key statistics of the two datasets. The complete pipeline of merging the two sources can be found in Appendix C.

[2]The drug name is 'NSCxxxxx', where xxxxx is the NSC ID.

**NCI-SS** STITCH [39] is a drug-protein interaction network dataset, and STRING [38] consists of protein-protein interactions. We align NCI-ALMANAC to STITCH-STRING on PubChem ID [23].

**NCI-GP** Cheng et al [10] studies the drug combination on different diseases and concludes that *graph-based proximity* can better measure the drug combination effect. It provides a cleaned-up version of protein-protein and drug-protein interaction network. We map this dataset to NCI-ALMANAC data on DrugBank ID [42].

### C.2 Distribution of the dataset

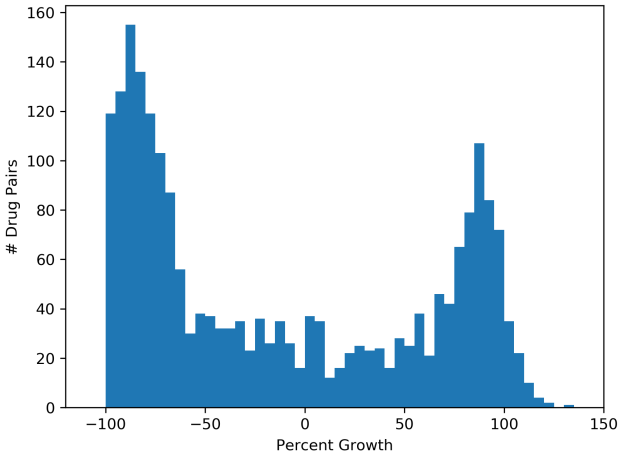



Figure 3: Distribution for NCI-SS.

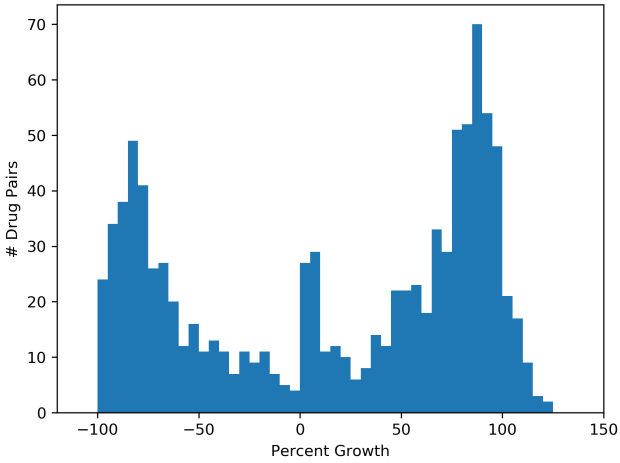



Figure 4: Distribution for NCI-GP.

## D Baselines

The **Extended Connectivity FingerPrints (ECFP)** [31] is a classic method to encode the chemical information. It follows the graph topology of the molecule and maps the substructure onto a bit vector with a hash function.

**Protein View Methods** We applied **Graph Convolutional Network (GCN)** [24] on the knowledge graph, where each node is the protein and edge is the interaction between proteins.

**UMVI** **Simple Mean** is taking the average of the two models. **Deep CCA** [3] attempts to maximize the correlation among different views. **Deep Fusion** [27] first integrates the representation from different views, and then passes through the fully-connected layers for the final prediction. **InfoNCE** [40] is contrasting the representation of different views in a self-supervised manenr.

# E Contrastive Representation Learning

Contrastive learning aims at distinguishing different data points in the feature space by maximizing the mutual information between them. Noise Contrastive Estimation (InfoNCE) [15, 33] is one of the most commonly used contrastive losses over two views $x_c$, $x_p$. Here we provide the definition of InfoNCE across the defined two views:

$$\mathcal{L}_{\text{InfoNCE}}(X_c, X_p) = -\log \mathbb{E}\left[\frac{\exp[s(x_c^i, x_p^i)/\tau)]}{\sum_{j=1}^{N}\exp[s(x_c^i, x_p^j)/\tau]}\right] \tag{6}$$

where $s$ is the cosine similarity between the representations of the two views, *i.e.* $s(x_c^i, x_p^j) = \frac{\langle g(x_c^i), g(x_p^j)\rangle}{\|g(x_c^i)\|\cdot\|g(x_p^j)\|}$, and $\tau$ is a temperature hyper-parameter.

Equation (6) presents a loss for contrasting $X_c$ against $X_p$, which is asymmetric. As in [40], we can recover a symmetric loss by also contrasting $X_p$ against $X_c$, which is $\mathcal{L}_{\text{InfoNCE}}(X) = \mathcal{L}_{\text{InfoNCE}}(X_c, X_p) + \mathcal{L}_{\text{InfoNCE}}(X_p, X_c)$.

# F Implementation Details

Since this is a regression task, we use root mean squared error (RMSE) and mean absolute error (MAE) for evaluation. For hyper-parameter tuning, we follow the rigorous pipeline for hyper-parameter tuning: we select the optimal hyper-parameter on a subset (80%) of the dataset. One thing to highlight is whether or not adding InfoNCE is one hyper-parameter for CPA, MPMPand DPMP. More details on hyperparameters can be found in Appendices E and G. Then with the optimal hyper-parameters, we train and test models on 80% and 20% data respectively, repeating this five times for the cross-validation.

# G Hyper-parameter Tuning

We are using 4 folds for the hyper-parameter tuning. Here we list the most important ones.

- **GCN**: we experiment on different GCN layer dimension, including {[1024, 128], [1024, 64], [128, 32], [128], [64]}.
- **ECFP**: the bit vector length and radius are usually set to 1024 and 2 respectively, and we tune the representation network for ECFP with NN structure including {[1024, 128], [1024 64]}.
- **Matching network**: this is after the concatenation of the drug pair, and we test {[128, 64], [128], [64]}.
- **Contrastive loss**: we tune the temperature $\tau$ in {0.01, 0.1, 1, 10, 100}.
- **Optimization**: we use Adam for optimizer, and check learning rate from {0.003, 0.001}, epochs from {100, 1000}.