STRUCTURE-BASED NETWORKS FOR DRUG VALIDATION

Cătălina Cangea[1], **Arturas Grauslys**[2], **Pietro Liò**[1] **and Francesco Falciani**[2]
[1]University of Cambridge [2]University of Liverpool

ABSTRACT

Classifying chemicals according to putative modes of action (MOAs) is of paramount importance in the context of *risk assessment*. However, current methods are only able to handle a very small proportion of the existing chemicals. We tackle this issue by exploring *deep learning architectures* that learn from molecular structures of drugs and their effects on human cells. Our choice of architectures is motivated by the significant influence of a drug's chemical structure on its MOA. We evaluate the performance of several models on two datasets of transcriptional responses: in one case, we improve on the strong ability of a unimodal architecture (F1 score of 0.803) to classify drugs by their *toxic MOAs* (Verhaar scheme) through adding another learning stream that processes transcriptional responses of human cells affected by drugs from the LINCS L1000 Phase II dataset. Our integrative model achieves an even higher classification performance—the error is reduced by 4.6%. However, subsequent experiments on the Phase I data from the same project show that learning only from the molecular structure yields the best generalization ability (F1 of 0.910). We conclude that the latter and more robust approach could be used to extend the current Verhaar scheme and constitute a basis for fast drug validation and risk assessment.

1 INTRODUCTION

Industrial chemistry is nowadays heavily centered around two topics of increasing concern: risk assessment and drug regulation. Tens of thousands of new chemicals are being synthesised every year—this calls for a faster validation process, in order to advance possible candidates to the next drug development stage or ascertain toxicity levels before releasing chemicals into the environment.

One well-established risk assessment method is the Verhaar scheme (Verhaar et al., 1992), which assigns a chemical one of four possible toxic modes of action: *inert*, *less inert*, *reactive* or *specifically acting*. This method is entirely based on chemistry principles and determines the class of a compound using a sequence of structural triggers that attempt to place a compound in one of the four classes (Enoch et al., 2008). However, the scheme can only be applied to a very small percentage of the existing chemical space, as the triggers do not have any effect on most compounds, which makes it impossible to determine their toxicity.

We explore an integrative method of learning from the *transcriptional responses of human cells exposed to chemicals* and *structural representations of the chemicals*. Our architecture achieves a high Verhaar classification performance (F1 score of 0.812) on chemicals used in the LINCS L1000 project[1], *without requiring access to the classification rules themselves*. Instead, the model learns only from the chemicals' effects on human cells and their unbiased, raw structure given by their *molecular fingerprint* (Cereto-Massagué et al., 2015). However, subsequently evaluating this model on the data from Phase I[2] shows a much weaker performance, with the integrative model failing to improve much beyond only learning from transcriptional responses alone. Consequently, we propose using the molecular fingerprint alone to place a chemical in one of the Verhaar classes, as this method achieves very strong performances on both sets of chemicals (Phase I and II).

[1]L1000 Connectivity Map Phase II perturbational profiles from Broad Institute LINCS Center for Transcriptomics: `https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE70138`

[2]`https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE92742`

2 RELATED WORK

Existing research using the LINCS transcriptomic data is based in areas which include repurposing drugs, investigating their properties and predicting their adverse effects. Aliper et al. (2016) used deep neural networks (DNNs) and support vector machines (SVMs) to learn from LINCS data corresponding to 678 drugs across three cell lines. They predicted 12 therapeutic use categories derived from the MeSH database and proposed a DNN confusion matrix approach for drug repurposing. Iwata et al. (2017) also used the LINCS data to elucidate MOAs of bioactive compounds. They first performed pathway enrichment analysis to reveal similarly classified drugs, using previously adopted procedures (computing the p-value of two sets of genes intersecting), then predicted the target protein using known compound-target interactions as ground truth (via cell-based similarity search using the Pearson correlation coefficient) and finally revealed new such interactions.

Our approach is mostly similar to the one adopted by Wang et al. (2016), who integrated gene expression (GE) data with cell multiplex-cytological (MC) profiles and the chemical structure (CS) of drugs in the form of fingerprints to predict adverse drug reactions (ADRs). The three data types were combined for feature selection, which yielded the 50 most predictive features that were used to learn an L1 regularized logistic regression model. The function coefficients were averaged across 200 runs to indicate feature importances. Extra Trees (ETs) classifiers were then trained for each ADR on GE, MC and CS data, using the 251 drugs shared among these three data sources. The authors discovered that the GE data had the highest prediction ability, whereas the CS and MC profiles showed similar AUROC (area under ROC) values. Additional findings showed that integrating MC profiles with other attributes did not significantly improve the classification performance, whereas combining CS features with GE data resulted in better predictions of ADRs. Unlike Wang et al., we do not restrict the model to a certain number of features and allow it to learn the best ones, providing only the raw data (entire gene expression vectors and fingerprints) to the classifier.

3 MODEL AND INPUTS

3.1 DATASET

We used two types of data for classification: (a) *gene expression levels* encoding the transcriptional responses of chemicals across human cell lines, represented by a vector of 978 landmark genes, and (b) the respective *molecular fingerprints*, which are bit strings encoded by binary vectors of length 1024 (see Figure 1). The goal is to place each compound (represented by the two modalities) in one of 4 Verhaar classes.

The gene expression levels were taken from the LINCS Phase II and I data. The Phase II dataset contains 118050 samples, corresponding to individual experiments with a single compound across 12328 genes, out of which 978 are *landmark genes* (used to infer the remaining 11350). A portion of the dataset (53976 examples) has been labelled according to the Verhaar classification scheme (Verhaar et al., 1992), which indicates the toxicity of compounds based on their MOAs. The labels correspond to 4 classes and a fifth category—chemicals that cannot be classified by the Verhaar rules.

The architecture was trained using the 14116 examples across the first four classes enumerated in Table 1, each of them only containing perturbations corresponding to the 978 landmark genes. We have used the Phase I dataset in a similar fashion—statistics are presented in the same table.

Table 1: Verhaar (Verhaar et al., 1992) class distribution for the labelled LINCS data, across the first 4 classes. The Phase I and II datasets are unbalanced and each chemical is used in several experiments.

Chemical class	PII Experiments	PII Chemicals	PI Experiments	PI Chemicals
Inert	1092	26	4488	227
Less inert	1949	37	7255	345
Reactive	6474	104	20483	1090
Specifically acting	4601	31	5119	293
Total	**14116**	**198**	**37345**	**1955**

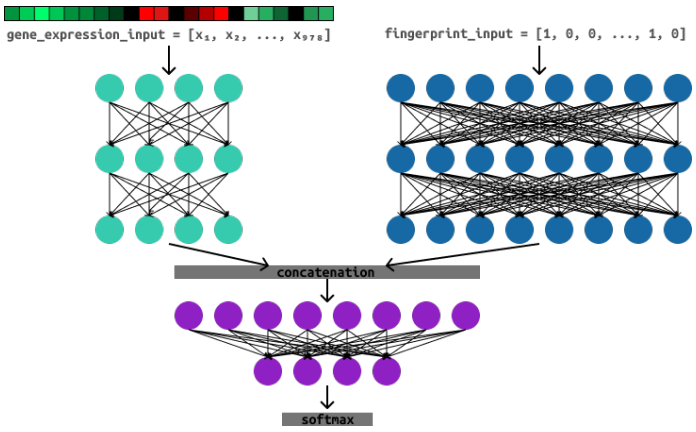

Figure 1: Illustration of the integrative architecture used to classify chemical compounds. Each example contains two inputs (vector of thresholded and averaged gene expression levels and fingerprint binary vector) and is placed into one of the four Verhaar classes.

3.2 Model architecture

We design a multimodal neural network architecture to classify the chemicals by the Verhaar scheme (see Figure 1 for a graphical description). Two learning streams extract features separately from each of the modalities (gene expression levels and molecular fingerprint); the resulting 1D vectors are subsequently concatenated and a final, joint representation is learned at the tail of the model. The classification is achieved by a softmax layer.

3.3 Data pre-processing

One chemical is always used in multiple experiments—and therefore examples—across the LINCS dataset. As a consequence, the unique chemicals represented in the labelled portion are far fewer than the number of experiments; the distribution across the four classes is shown in Table 1. This implies that all training examples involving the same chemical would contain the same fingerprint data, which might result in *overfitting* on the corresponding learning stream in the network. In order to avoid this issue, we *summarized* the gene expression vectors from *all experiments* that were carried out using the same chemical. The resulting summary is then used as part of one example, along with the corresponding fingerprint vector—this reduces the Phase II dataset size to *198 examples*.

Summarization Upon inspection of all experiments that use a single chemical, we noticed that some of the transcriptional response vectors do not exhibit high variance and are thus less informative than others, potentially even confounding the discrimination across classes. Each such set of experiments was filtered by comparing the standard deviation of a gene expression level vector with a fixed threshold t to keep or remove the experiment—by experimenting with values in the range $[0..2]$, we found that $t = 1.25$ works best. The final summary is represented by a vector of length $2 \times$ *NUM_LANDMARK_GENES* which represents the concatenation of the element-wise *sum* and *maximum* across the filtered experiments. Figure 2 shows a depiction of the summarization pipeline.

4 Experiments

4.1 Evaluation procedure

We used k-fold cross-validation with $k = 5$ to assess the performance of our models. Both unimodal architectures were trained for 1000 epochs and the integrative model—for 2000 epochs. All models were trained on a batch size of 26, using the RMSprop optimizer (Hinton et al.), with a learning rate of 0.005 and default hyperparameters [3] otherwise. As the class distribution across the LINCS labelled data is *unbalanced* (see Table 1), we have used a weighted version of the F1 score to evaluate model

[3] https://pytorch.org/docs/stable/optim.html#torch.optim.RMSprop

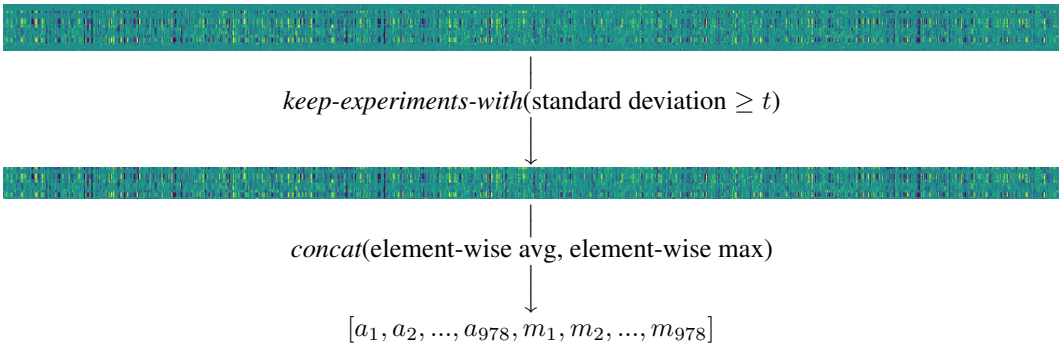

Figure 2: Schematic description of the summarization pipeline described in Section 3.3. Horizontal axis represents genes, vertical axis represents different experiments. The input is the entire set of gene expression levels corresponding to experiments with a single chemical across the LINCS dataset. The pipeline outputs the element-wise average and maximum of the filtered gene expression vectors.

performance. Accordingly, all models were trained with a weighted cross-entropy loss function, with weights being inversely proportional to the relative class proportions.

4.2 Model parameters

Our unimodal baselines are both 3-layer perceptrons. Each layer has 256 hidden units and is followed by an ELU activation (Clevert et al., 2015), a batch normalization operation (Ioffe & Szegedy, 2015) and dropout regularization (Srivastava et al., 2014) with $p = 0.25$ or $p = 0.5$ for the final layer. The streams of the integrative model have the same layout as the baselines described above (256-D layers for fingerprints, 64-D for gene expression levels—see Section 3 for details); the concatenation operation is regularized with dropout with $p = 0.25$. Finally, the tail of the model is formed of two fully-connected layers (256- and 32-D) with ELU activations and dropout layers with $p = 0.5$. Batch normalization is employed beyond the concatenation point.

Table 2: Results of 5-fold cross-validation for the unimodal and integrative architectures. All models were evaluated on both datasets (LINCS Phase II and subsequently Phase I data).

Model	F1 score (PII)	F1 score (PI)
Majority class	0.362	0.399
Gene expression	0.606 ± 0.049	0.478 ± 0.014
Fingerprints	0.803 ± 0.063	$\mathbf{0.910 \pm 0.010}$
Fingerprints and gene expression	$\mathbf{0.812 \pm 0.043}$	0.495 ± 0.011

4.3 Results

Table 2 shows that jointly learning from gene expression levels and molecular fingerprints results in a strong classification ability for the 4 Verhaar classes when using Phase II data. This result improves over both unimodal architectures, suggesting that adding information about the effects of chemicals can enhance the powerful discrimination capabilities of the fingerprint-only model.

Surprisingly, when evaluating the same models on the Phase I data, we observed that the joint model could not improve much beyond the performance of the gene-expression only network. This suggests that the transcriptional responses are heavily confounding the network, not allowing it to usefully merge this information with the chemical structure features, and might not even reflect the Verhaar scheme. The latter assumption invites further investigation into more informative modalities that can be combined with chemical structure.

This result is particularly important for classifying the unlabelled chemicals in LINCS. As a vital goal of this research direction, assessing the toxicity of *any chemical* could become a simple process by using the fingerprint-only model. The next step we aim to make is to validate the predictions on the unlabelled chemicals experimentally and using expert knowledge.

REFERENCES

Alexander Aliper, Sergey Plis, Artem Artemov, Alvaro Ulloa, Polina Mamoshina, and Alex Zhavoronkov. Deep learning applications for predicting pharmacological properties of drugs and drug repurposing using transcriptomic data. *Molecular pharmaceutics*, 13(7):2524–2530, 2016.

Adrià Cereto-Massagué, María José Ojeda, Cristina Valls, Miquel Mulero, Santiago Garcia-Vallvé, and Gerard Pujadas. Molecular fingerprint similarity search in virtual screening. *Methods*, 71: 58–63, 2015.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (ELUs). *arXiv preprint arXiv:1511.07289*, 2015.

SJ Enoch, M Hewitt, MTD Cronin, S Azam, and JC Madden. Classification of chemicals according to mechanism of aquatic toxicity: An evaluation of the implementation of the Verhaar scheme in Toxtree. *Chemosphere*, 73(3):243–248, 2008.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv preprint arXiv:1502.03167*, 2015.

Michio Iwata, Ryusuke Sawada, Hiroaki Iwata, Masaaki Kotera, and Yoshihiro Yamanishi. Elucidating the modes of action for bioactive compounds in a cell-specific manner by large-scale chemically-induced transcriptomics. *Scientific reports*, 7:40164, 2017.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Henk JM Verhaar, Cees J Van Leeuwen, and Joop LM Hermens. Classifying environmental pollutants. *Chemosphere*, 25(4):471–491, 1992.

Zichen Wang, Neil R Clark, and Avi Ma'ayan. Drug-induced adverse events prediction with the LINCS L1000 data. *Bioinformatics*, 32(15):2338–2345, 2016.