BMC Bioinformatics

RESEARCH Open Access

Transplatformer: translating toxicogenomic profiles between generations of platforms

Guojing Cong[1*], Robert Patton[1], Frank Chao[3], Daniel L. Svoboda[2], Jeremy N. Erickson[3], Michele R. Balik-Meisner[2], Deepak Mav[2], Dhiral P. Phadke[2], Elizabeth H. Scholl[2], Ruchir R. Shah[2] and Scott S. Auerbach[3*]

*Correspondence:
Guojing Cong
congg@ornl.gov
Scott S. Auerbach
auerbachs@niehs.nih.gov
[1]Oak Ridge National Laboratory, Oak Ridge, TN, USA
[2]Sciome LLC, RTP, Durham, NC, USA
[3]National Institute of Environmental Health Sciences, RTP, Durham, NC, USA

Abstract

Background Transcriptomic profiling technologies have advanced the analysis of biological and toxicological responses. However, substantial differences in probe design, dynamic range, gene coverage, and preprocessing pipelines across platforms introduce artifacts that limit cross-study integration and hinder the reuse of historical datasets. We aim to develop computational methods for accurate cross-platform translation to maximize the value of legacy resources.

Results We present *TransPlatformer* a deep learning framework for translating gene expression profiles across heterogeneous toxicogenomics platforms. *TransPlatformer* employs a novel attention-based architecture to map high-dimensional fold-change vectors from legacy microarray technologies to current platforms. Models are trained and evaluated using DrugMatrix, spanning three technological generations. We investigate mixed-tissue, single-tissue, and cross-tissue training paradigms and benchmark performance against multilayer perceptron and matrix-completion baselines. In mixed-tissue training, *TransPlatformer* achieves a greater than 50% reduction in mean absolute error (0.043 vs. 0.09) and nearly doubles Pearson correlation ($\approx$ 0.71 vs. 0.37) relative to baseline methods. Importantly, *TransPlatformer* preserves rare but biologically meaningful over- and under-expressed signals, with mean absolute error below 0.22. Single-tissue models yield further improvements for well-represented organs, such as a 10% reduction in liver mean absolute error, while underscoring the need for data augmentation strategies in low-sample tissues.ra

Conclusions *TransPlatformer* provides an effective and scalable computational solution for cross-platform transcriptomic translation. By enabling biologically faithful harmonization of gene expression data, the proposed approach facilitates the reuse of legacy toxicogenomics datasets, enhances downstream biomarker discovery, and supports more reproducible predictive modeling in toxicology.

Keywords Toxicogenomics, Transcriptomic data integration, Deep learning, Attention mechanisms, Cross-platform translation, DrugMatrix, Gene expression analysis

Background

Gene expression analysis is a cornerstone of modern biology, providing crucial insights into the mechanisms underlying cellular function, disease processes, and toxicological responses. Over the years, the technological landscape for transcriptomic profiling has evolved dramatically—from early microarray platforms such as CodeLink [1] and Affymetrix [2] to RNA sequencing [3] and targeted panels like BioSpyder Whole Transcriptome (BioSpyderWT) [4]. While these innovations have expanded the scope and precision of gene expression studies, they have also introduced technical variations. Differences in probe design, gene coverage, dynamic range, and normalization methods create noise and variation that complicate direct comparisons and data integration.

Motivated by the desire to increase the power of genomic analysis and biomarker discovery through the integration of multiple data sets, early efforts in harmonizing gene expression data across platforms focused on mitigating distributional discrepancies caused by differences in probe design, signal detection, and preprocessing protocols [5, 6]. Foundational methods, such as Quantile Normalization [7] and Robust Multi-array Analysis with its frozen variant [8], were developed to align expression values by enforcing a common distribution across samples. Building on these techniques, more refined methods emerged, including Feature-Specific Quantile Normalization [9] and Feature-Specific Mean–variance Normalization that adjust distributions on a gene-by-gene basis while preserving the biological variability critical for maintaining gene-specific signals [10].

Alongside these statistical standardization efforts, researchers have employed Bayesian frameworks and other statistical models to address systematic biases and batch effects inherent in cross-platform data. Empirical Bayes-based approaches such as ComBat [11] and ComBat-Seq [12] allow for the sharing of information across genes and samples, thereby reducing technical variability without sacrificing underlying biological signals. At the same time, machine learning methods have been introduced to capture nonlinear relationships that traditional statistical techniques may overlook, marking a shift toward more dynamic data transformation efforts.

Advanced machine learning frameworks such as Distance-Weighted Discrimination [13] and cross-platform normalization [14] utilize matched sample designs to estimate and correct systematic differences through piecewise adjustments. More sophisticated methods like Training Distribution Matching [15] reshape test data distributions to mirror those observed in training sets, directly enabling models trained on legacy platforms to be applied to newer RNA-seq datasets without extensive recalibration.

Deep learning approaches have further revolutionized cross-platform integration by leveraging neural network architectures capable of capturing complex, hierarchical representations of gene expression [16–18]. Frameworks such as MultiPLIER distill latent transcriptomic "signatures" from large compendia to benefit smaller or rare disease datasets [19], while generative models such as generative adversarial Networks facilitate synthetic data generation and imputation [17]. Some tools, (e.g., HE2RNA) aim to predict transcriptomic profiles from histology slide images [20]. These approaches, from traditional normalization to cutting-edge machine learning, help integrate heterogeneous gene expression profiles, ultimately enabling more accurate downstream analyses in diverse biomedical applications.

Despite these advances, there remains a gap in the reliable transformation of legacy data to modern expression formats, particularly in applications that span toxicology, drug discovery, and clinical research. Integrating historical and contemporary datasets without compromising the biological signal is imperative for leveraging the vast amounts of previously generated data. We introduce *TransPlatformer*, a novel transformer-based deep learning model designed to translate gene expression profiles from older platforms into the metrics produced by current sequencing technologies and vice versa. With *TransPlatformer* we translate legacy substrate datasets, for example, those measurements in CodeLink and Affymetrix format in DrugMatrix [21] to a modern format. We show that compared to baselines such as matrix completion, simple regressors, and multilayer perceptron (MLP), *TransPlatformer* achieves much better performance in both translating global patterns and preserving rare signals (e.g., over- and under- expressed gene expressions). We demonstrate the potential of *TransPlatformer* in harmonizing data for biomarker discovery using a liver necrosis study. Liver treatment profiles in DrugMatrix are distributed onto three platforms: CodeLink, Affymetrix, and BioSpyderWT. Augmenting BioSpyderWT data with those translated from CodeLink and Affymetrix, the performance (i.e., F1 score) for a simple MLP predictor improves by approximately 8%. In addition, the powerful attention mechanism in *TransPlatformer* also reveals relationships among genes captured in its learning functions during translation. For example, in translating CodeLink data to the BioSpyderWT platform, we find that *TransPlatformer* pays much more attention to certain genes. Combined with and in complement to gene network analysis, the attention weights in *TransPlatformer* can reveal deeper insights in toxicogenomics.

Methods

We explore data and AI methods for translating transcriptomic profiles between different platforms. The machine learning model takes as input vectors of gene profiles on one platform and outputs the translated vectors on another. Training the model requires vector pairs, one profile from the source platform and another from the target platform.

Data

We curate data from DrugMatrix to build our models. As its current form contains data that span three generations of platforms and has been used extensively in deriving signatures of toxicity effects and mechanistic processes (e.g., see [22]), DrugMatrix is the best publicly available candidate dataset for our study.

The original DrugMatrix contains data from two different microarray platforms: the GE Healthcare CodeLink UniSet Rat I [23], a first-generation microarray technology no longer available, and the Affymetrix GeneChip Rat Genome 230 2.0 [2], a second-generation technology still in use [21]. The extended DrugMatrix data was generated using DrugMatrix legacy frozen RNA samples (i.e. those characterized using microarray technology). Sequencing of the samples was done using the BioSpyder S1500+ platform ([24], a subgenomic, targeted sequencing technology. The S1500+ data were then extrapolated to the full genome using GeniE (https://www.sciome.com/genie/) to gener ate approximately 20,000 gene expression measurements that correspond to the probes that are contained in the BioSpyder whole transcriptome (BioSpyderWT) platform. The data from all three platforms form a matrix $G = R^{n\times m}$, $n \approx 375{,}000$, $m \approx 2200$, where

Table 1 Amount of measured data for different tissues on three platforms in DrugMatrix

Platform	LI	KI	HE	SM	BM	BR	IN	SP
CodeLink	14.2M	7.7M	5.3M	0.2M	2.7M	0.55M	0.17M	1.5M
Affymetrix	20.3M	11.3M	6.5M	1.3M	0	0	0	0
BioSpyderWT	17.6M	15.9M	10.7M	6.5M	0.09M	0.07M	0.5M	1.6M

Table 2 Data distribution in DrugMatrix for five categories

Category	Extremely-under	Under	Normal	Over	Extremely-over
Percentage	0.03	4.09	91.94	3.88	0.036

the rows are the genes and the columns are the treatments. Approximately 12% of the endpoints in the matrix are present. A fundamental challenge in harmonizing these datasets lies in the distinct nature of their signal detection. The CodeLink and Affymetrix platforms utilize hybridization microarrays, which generate continuous fluorescence intensity values (analog data) subject to saturation and background noise. In contrast, the BioSpyderWT platform utilizes targeted sequencing, generating discrete read counts (digital data) with a wider dynamic range. To resolve the incompatibility between raw intensities and raw counts, we do not feed the model absolute expression values. Instead, we standardized the input and output for TransPlatformer as log10 fold-change ratios. This transformation converts platform-specific units into a unified metric of relative biological perturbation, neutralizing the distributional differences between analog intensities and digital counts. Descriptions of the methods used to generate the processed data can be found in Supplementary Materials.

Compared to another popular toxicogenomics database, Open-TGGATEs [25], that only has data for two tissues, liver and kidney, DrugMatrix contains data for eight tissues. However, tissue coverage is not even with the three platforms in DrugMatrix, as shown in Table 5. The CodeLink and BioSpyderWT platforms cover all eight tissues including liver (LI), kidney (KI), heart (HE), skeletal muscle (SM), spleen (SP), bone marrow (BM), intestine (IN), and brain (BR). In contrast, Affymetrix covers only LI, KI, HE, and SM.

Table 1 shows that the data distribution is highly skewed for each tissue. On all three platforms, LI is the most represented tissue, followed by KI and HE. There are very few data points for BR and IN. For example, on CodeLink, only about 2% and 0.5% of the data are for BR and IN, respectively. On BioSpyderWT there are slightly more IN data. The skewed tissue coverage creates implications for building AI models for translation, which we will discuss in Sect. 4.

The values in DrugMatrix fall within the range of (– 5, 5). According to the expression level, the data are divided into five categories: extremely under-expressed (– 5, – 1], under-expressed (– 1, − 0.3], normal (− 0.3, 0.3), over-expressed [0.3, 1), and extremely over-expressed [1, 5). Table 2 shows the distribution of data in the 5 categories.

The normal category dominates the other categories, with about 92% of all available data samples. Although the other categories collectively represent fewer than 4% of the data, they carry important biological and toxicological meaning. It is imperative that they are preserved as rare signals in translation and not lost to noise.

The transcriptomic profiles on all three platforms are high-dimensional vectors as shown in Table 3. Even the smallest dimension among the three is 8565 for CodeLink.

Table 3 Dimensions of transcriptomic profiles on three platforms

CodeLink	Affymetrix	BioSpyderWT
8565	31,042	22,794

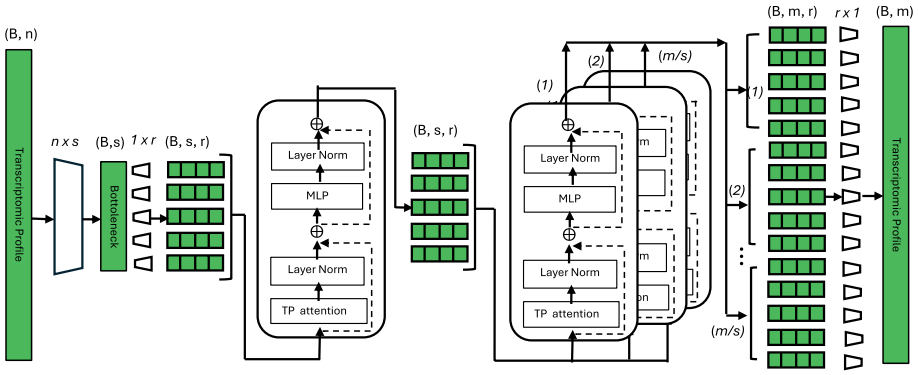

Fig. 1 Architecture of *TransPlatformer*. The input is a vector of size *n*, and the output is a translated vector of size *m*. Colored blocks represent the data (tensors) that pass through the encoder and decoder of *TransPlatformer*, separated by the green block in the middle as the encoded states of the inputs. Note there are $\lceil m/s \rceil$ concurrent decoding structures on the right. The attention mechanism is *TransPlatformer* attention

The largest is 31,042 for Affymetrix. High dimensions present challenges for both model construction and model training.

Machine learning models

Classic machine learning models such as random forest and deep learning models such as MLP can be used for the translation purpose. Training and inferencing with these models are relatively fast. However, the simple architectures may be incapable of preserving rare signals buried in long vectors.

Sequence to sequence (Seq2Seq) transformers can be another candidate. Several high-profile models such as BERT and GPT4 are popular in science applications [26]. Unfortunately, these models are not a good fit for translating transcriptomics. First, the ultra-high dimensions of inputs and outputs require tremendous compute and memory resources due to the self-attention mechanism [27]. Second, and more importantly, neither the inputs nor the outputs are strictly sequences analogous to sentences in natural languages. The genes form pathways and networks. Third, treating the profiles as sequences places unnecessary constraints on the learning capabilities of the model.

We propose *TransPlatformer*, a deep learning model for translating transcriptomic profiles based on a new attention mechanism. Its architecture is tailored to toxicogenomics profiles, shown in Fig. 1. The *TransPlatformer* architecture bears some resemblance to the Seq2Seq transformer architecture with several important differences. The input on the left in Fig. 1 is a batch B of vectors each of size n. The input passes through an (optional) fully connected bottleneck layer ($n \times s$), followed by an embedding layer $1 \times r$. At this point, the input is transformed into a tensor of size $B \times s \times r$. It then passes through several TransPlatformer attention layers stacked together in a fashion similar to the encoder in Seq2Seq transformers. The encoded tensor represents the latent states learned from the input profiles, and then passes through $\lceil m/s \rceil$ concurrent decoding TransPlatformer attention layers. These $\lceil m/s \rceil$ decoding networks each

produce a $B \times s \times r$ tensor. These tensors then pass through a projection network $r \times 1$ and are assembled into a $B \times m$ tensor for the predicted output.

For an input vector $I \in \mathbb{R}^{s \times r}$, *TransPlatformer* adopts an attention mechanism by learning a weight matrix $W \in \mathbb{R}^{r \times (o \cdot r)}$ that produces an output vector $O \in \mathbb{R}^{o \times r}$ in the following steps.

1. Feed forward:

$$F = IW_F, F \in \mathbb{R}^{o \times r}$$

2. Linear transformation:

$$H = IW, \quad T \in \mathbb{R}^{s \times o \cdot r}$$

3. Tensor reshape:

$$H_{sor} \in \mathbb{R}^{s \times (o \times r)}$$

4. Compute attention weights along the dimension with size s:

$$A_{sor} = \mathrm{Softmax}(H_{sor})$$

5. Apply attention and sum over s:

$$O = \sum_s A_{sor} F_{or}$$

In comparison, self-attention in Seq2Seq transformers takes an input sequence $I \in \mathbb{R}^{s \times r}$ and learns three weight matrices in addition to the feedforward matrix W_F:

$$W_Q \in \mathbb{R}^{r \times d}, \quad W_K \in \mathbb{R}^{r \times d}, \quad W_V \in \mathbb{R}^{r \times d}$$

With $F = W_F \in \mathbb{R}^{s \times r}$, self attention consists of the following steps.

1. Compute queries, keys, and values:

$$Q = IW_Q, \quad K = IW_K, \quad V = IW_V$$

2. Compute attention scores:

$$A = \frac{QK^T}{\sqrt{d}}, \quad A \in \mathbb{R}^{s \times s}$$

3. Softmax over attention scores:

$$\alpha = \mathrm{Softmax}(A)$$

4. Apply attention:

$$O = \alpha V, \quad O \in \mathbb{R}^{s \times d}$$

5. Apply skip attention:

$$O = O + F$$

Table 4 summarizes the differences between *TransPlatformer* attention and transformer self-attention [27].

Model training

The models we evaluate are capable of translating between any pair of platforms. Considering the practical use of harmonizing historical measurements to new platforms, we focus on translations from CodeLink and Affymetrix to BioSpyderWT. Each data sample is a tuple: (source profile, target profile). The skewed data distribution in DrugMatrix requires careful considerations of training strategies. In the ideal scenario with plenty data samples, training is straightforward. Unfortunately, as shown in Table 1, our training data are limited and their distribution skewed. We propose and evaluate three training strategies.

The first is training *TransPlatformer* with data from all tissues. It produces one model for translation. We denote this mixed-tissue mode. In mixed-tissue mode, the maximum amount of training samples is used to train the model. In translating from CodeLink to BioSpyderWT, there are a total of 2,225 samples. The number of samples is 888 for translating from Affymetrix to BioSpyderWT. Mixed-tissue training optimizes for the overall prediction performance for all tissues. The model may not perform best for individual tissues. We consider a second training strategy, single-tissue mode, where training is done with data from each individual tissue. As a result, from CodeLink to BioSpyderWT, we will train eight *TransPlatformer* models and from Affymetrix to BioSpyderWT four models. In single-tissue mode, a model can be well-tuned for each individual tissue that has enough data samples. The third strategy is cross-tissue mode, where the models are trained with data from one set of tissues and tested on a different set of tissues. Cross-tissue training presents an opportunity for us to investigate whether translating functions are common across tissues and whether one learned from one tissue can be applied to another.

The *TransPlatformer* models in all training strategies have the same architecture. Each model contains 32 *TransPlatformer* layers. The embedding dimension r is 16. From CodeLink to BioSpyderWT, $n = 8,565$, $m = 22,794$, and from Affymetrix to BioSpyderWT, $n = 31,042$, $m = 22,794$. We set $s = 512$. In translating both CodeLink and Affymetrix to BioSpyderWT, there are 45 concurrent output segments in *TransPlatformer*. We use the Adam optimizer with default settings, that is, learning rate $1e^{-3}$ and weight decay $5e^{-4}$. The batch size is 16. We train the models for 40 epochs. These hyperparameters are selected by a simple grid search.

Table 4 Comparison of *TransPlatformer* attention and Seq2Seq transformer attention

Feature	*TransPlatformer*	Transformer attention
Weight matrix	$W \in \mathbb{R}^{r \times (o \cdot r)}$	W_Q, W_K, W_V (separate)
Linear transformation	$H = IW$	$Q = IW_Q, K = IW_K, V = IW_V$
Attention mechanism	H_{sor} (reshaped)	QK^T
Softmax computation	A_{sor}=Softmax over H_{sor}	α=Softmax over $QK^T/\sqrt{d}$
Output computation	$O = \sum_s A_{sor} F_{or}$	$O = \alpha V \oplus F$

Results

The effects of *TransPlatformer* are evaluated using typical machine learning metrics such as mean absolute error (MAE) and Pearson correlation coefficient (PCC) between prediction and ground-truth. In addition, we conduct toxicogenomics validations and characterizations of predicted data. We also give use cases of harmonized data with *TransPlatformer*.

We include a classic machine learning baseline, random forest regressor (RF), for comparison. Random forest is simple and widely used in science applications.

We include an MLP model as another baseline. The model has an input $n \times h$ layer followed by k hidden layers of size $h \times h$. Each layer is followed by activation and dropout. The output layer is $h \times m$. We use grid search to determine the best hyperparameters. In our experiments, $h = 800$, $k = 4$, activation is Gaussian error linear units (GELU) [28], and the dropout rate is 0.1. The batch size is 16.

Unless noted otherwise, in training and testing of *TransPlatformer*, MLP, and RF models for profile translation, we perform five random splits of the data sets and report the mean and range for test MAE.

We also compare against ToxCompl, an AI model developed by Cong et al. [29] to impute missing entries in DrugMatrix using a matrix completion approach. Strictly speaking, ToxCompl is not (only) a translation mechanism from one platform to another. The setup for ToxCompl is different from *TransPlatformer*. With ToxCompl, observed entries in the entire matrix are used to train a model that is then used to infer the missing ones. To translate a new profile from one platform to another, one can insert that profile into the current matrix and use ToxCompl to predict all missing entries, which naturally produces the profile on the target platform. This is obviously not the most efficient translation method. It leverages solely latent information in the low-rank matrix among all measurements and all tissues, and does not utilize the relationship between data entries and the platforms. As another baseline, we report the ToxCompl performance in completing the matrix. Comparison against ToxCompl reveals whether additional information, in this case the correspondence of gene profiles across platforms, can help improve prediction performance.

Mixed-tissue training results

Mixed-tissue training assumes that the tissue-identifying signals are latent in the inputs, and with a powerful architecture and plenty of data, the trained model can automatically differentiate these tissues.

Table 5 shows the test performance of *TransPlatformer* in mixed-tissue mode. Unless noted otherwise, we use a 9:1 train-test split of the data in all experiments. We report test MAE for all genes, test MAE for over- and under-expressed genes (including

Table 5 Performance of *TransPlatformer* (shown as TP) in comparison to MLP and RF for translating from CodeLink and Affymetrix to BioSpyderWT

From	CodeLink			Affymetrix		
	TP	**MLP**	**RF**	**TP**	**MLP**	**RF**
MAE	**0.043** ± 0.00	0.062±0.00	0.06±0.00	**0.038** ± 0.00	0.054±0.00	0.065±0.00
MAE o/u	**0.22** ± 0.03	0.43±0.02	0.28±0.02	**0.22** ± 0.03	0.45±0.01	0.29±0.02
PCC	**0.71** ± 0.01	0.37±0.03	0.58±0.02	**0.72** ± 0.01	0.39±0.01	0.58 ± 0.01

The MAE for all genes and MAE for over and under expressed genes (MAE o/u) are shown. The last row shows PCC between prediction and ground truth

extremely over- and extremely under-expressed ones), and Pearson correlation between prediction and ground truth. For comparison Table 5 also shows the performance of RF and MLP with the same splits.

In Table 5, *TransPlatformer* beats RF and MLP for all performance metrics. Specifically, in translating from CodeLink to BioSpyderWT, the overall *TransPlatformer* MAE for all genes is 0.043. For MLP and RF, the MAE is 0.062 and 0.06, respectively. They are much worse than *TransPlatformer*. The MAE achieved with ToxCompl is 0.09. The prediction error of *TransPlatformer* is about 2.1 times lower than ToxCompl. For over- and under- expressed genes, the MAE is 0.22, 0.43, and 0.28 for *TransPlatformer*, MLP, and RF, respectively. *TransPlatformer* outperforms MLP and RF by about 1.95 and 1.27 times, respectively. *TransPlatformer* achieves a PCC of 0.71 between prediction and ground truth, indicating a strong positive relationship, while MLP and RF achieve only a correlation of 0.37 and 0.58, respectively. Similar results are observed for translating from Affymetrix to BioSpyderWT. Since their performance is inferior to *TransPlatformer*, we do not include results for RF and MLP for the rest of our experiments. Instead we focus our discussion on *TransPlatformer*. We still include ToxComp results for comparison as it is a fundamentally different approach from *TransPlatformer*, MLP, and RF.

The test results in Table 5 are tissue-agnostic. That is, the 10% of test data is selected without any consideration of the tissues. Due to data imbalance, it is possible that very little test data or even none is selected for tissues such as IN and BR. We next evaluate tissue-wise translation performance. In this experiment, we withhold 5% data samples from each tissue and use the remaining to train the model. We then test the performance on holdout data. The PCC results are shown in Table 6. PCC between 0.60 and 0.75 is achieved for LI, KI, HE, and SM. These tissues are relatively well-represented in the data samples. The model performs poorly for under-represented tissues. PCC for BR and IN in translating from Affymetrix to BioSpyderWT is 0.20 and 0.24, respectively.

Single-tissue training results

In single-tissue mode, a separate model is trained for each individual tissue using data from that tissue alone. In contrast to mixed-tissue mode, there are no potentially confounding signals from data from other tissues. At the same time, the amount of training data is also fewer for each model. Some models may perform better than the mixed-tissue model for certain tissues. We train eight models in total for translating from CodeLink to BioSpyderWT, and four models for Affymerix to BioSpyderWT. Performance results are shown in Figs. 2, 3, 4, and 5. In each figure we include the model trained in mixed-tissue mode for comparison. Performance variance is plotted as error bars.

Figure 2 shows the MAE of *TransPlatformer* for all eight tissues in translating from CodeLink. The 'Mixed' column is for the model trained in mixed-tissue mode. The

Table 6 PCC between prediction and ground truth with *TransPlatformer* for translating from CodeLink and Affymetrix to BioSpyderWT

Tissues	**LI**	**KI**	**HE**	**SM**
CodeLink	0.75 ± 0.005	0.66 ± 0.01	0.68 ± 0.007	0.74 ± 0.03
Affymetrix	0.75 ± 0.02	0.67 ± 0.02	0.61 ± 0.01	0.60 ± 0.04
Tissues	**BM**	**BR**	**IN**	**SP**
CodeLink	0.66 ± 0.017	*0.20 ± 0.005*	*0.24 ± 0.01*	0.73 ± 0.02

Note that there is no Affymetrix data for BM, BR, IN, and SP

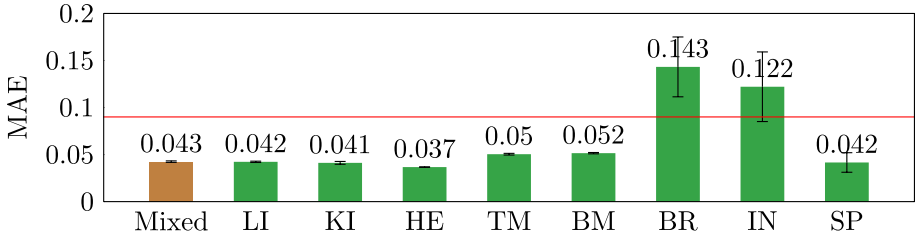

Fig. 2 Performance of *TransPlatformer* translating from CodeLink to BioSpyderWT. The horizontal red line at 0.09 shows the MAE with ToxCompl

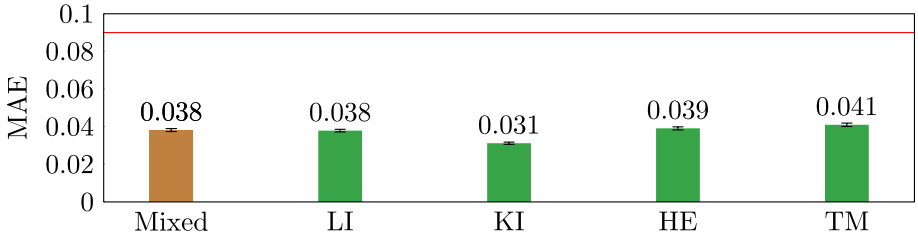

Fig. 3 Performance of *TransPlatformer* translating from Affymetrix to BioSpyderWT. The horizontal red line at 0.09 shows the MAE with ToxCompl

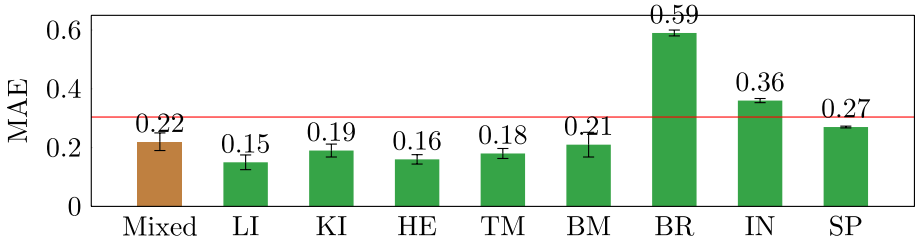

Fig. 4 Performance of *TransPlatformer* translating from CodeLink to BioSpyderWT: *over-*, *extremely over-*, *under-*, and *extremely under-* regulated genes as rare signals. The horizontal red line at 0.304 shows the MAE with ToxCompl

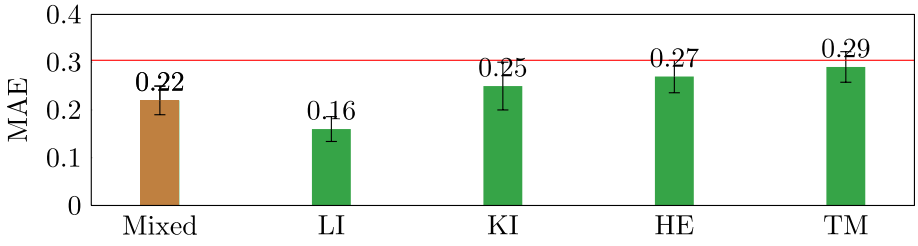

Fig. 5 Performance of *TransPlatformer* translating from Affymetrix to BioSpyderWT: *over-*, *extremely over-*, *under-*, and *extremely under-* regulated genes as rare signals. The horizontal red line at 0.304 shows the MAE with ToxCompl

horizontal red line is at 0.09, representing the performance achieved with ToxCompl. For tissues LI, KI, HE, TM, BM, and SP, *TransPlatformer* performs much better than ToxCompl. For tissues where data are relatively abundant, i.e., LI, KI and HE, the single-tissue model performs better than the mixed-tissue model. For tissues with fewer training data samples, for example, BR and IN, the performance is much worse than ToxCompl. For example, the MAE is 0.143 and 0.122 for BR and IN, respectively.

Figure 3 shows the performance of *TransPlatformer* translating from Affymetrix to BioSpyderWT for four tissues, LI, KI, HE, and TM. The 'Mixed' column is for the model

trained with data from all tissues in mixed-tissue mode. The horizontal red line is at 0.09, representing the baseline performance achieved with ToxCompl. For all tissues, *TransPlatformer* performs better than baseline, achieving between 2.19 and 2.88 times reduction in MAE. The MAEs for the tissues are also similar to that achieved by the 'Mixed' model. The best MAE is achieved for KI at 0.031.

Figure 4 shows the performance of the trained model for over-, extremely over-, under-, and extremely under- expressed genes. This is the same trained model as that in Fig. 2. The horizontal red line is at 0.304, representing the baseline performance achieved with ToxCompl. For tissues LI, KI, HE, TM, BM, and SP, *TransPlatformer* performs better than baseline. The biggest reduction in MAE, by 2.02 times, is observed for HE. For tissues with abundant data, i.e., LI, KI and HE, the single-tissue model performs better than the mixed-tissue model. For example, for LI, the MAE achieved is 0.15, a 28.5% improvement over the mixed-tissue model. For tissues with fewer training data samples, e.g., BR and IN, the test performance is much worse than both ToxCompl and the mixed-tissue model. For BR, the MAE is 0.59 and for IN, the MAE is 0.36.

Figure 5 shows the performance of *TransPlatformer* translating rare signals, that is, over-expressed and under-expressed genes, from Affymetrix to BioSpyderWT for four tissues, LI, KI, HE, and TM. This is the same trained model as that in Fig. 3. For all four tissues, *TransPlatformer* performs better than ToxCompl. The MAEs for the tissues are also similar to that achieved by the 'Mixed' model. The best MAE is achieved for LI at 0.016, a 27.2% improvement over the 'Mixed' model.

Table 7 shows for each model the PCC between prediction and ground truth. Compared to the mixed-tissue model in Table 6, single-tissue models perform on par or better for tissues with plenty of data (e.g., LI, and KI). Single-tissue models perform significantly better for HE (from CodeLink to BioSpyderWT), 0.73 vs. 0.68, and significantly worse for SM, 0.67 vs. 0.74. Unsurprisingly, single-tissue models perform poorly for BR and IN. In fact, due to limited data, their performance is much worse than the 'Mixed' model.

Our experiments show that single-tissue models perform well for LI, KI, and HE, while the mixed-tissue model performs well for BM, SM, and SP. They both perform poorly for BR and IN as these tissues are under-represented in the data.

Cross-tissue training results

Clearly, for tissues BR and IN, the training strategies that we have implemented fail to produce high-performing models. Recognizing the fact that model performance depends on the availability of data, we explore whether there is a better training strategy to improve performance for BR and IN. We consider cross-tissue mode where models are trained on data from a subset S of tissues and tested on a separate subset T of tissues. Here S and T do not overlap. An example of S and T can be $S = \{LI, KI, HE\}$, and

Table 7 PCC between prediction and ground truth with *TransPlatformer* for translating from CodeLink and Affymetrix to BioSpyderWT

Tissue	**LI**	**KI**	**HE**	**SM**
CodeLink	0.76 ± 0.02	0.70± 0.02	0.73 ± 0.01	0.67 ± 0.005
Affymetrix	0.77 ± 0.003	0.68 ± 0.002	0.60 ± 0.004	0.60 ± 0.01
Tissue	**BM**	**BR**	**IN**	**SP**
CodeLink	0.64 ± 0.03	*−0.40 ± 0.07*	*0.05 ± 0.02*	0.63 ± 0.04

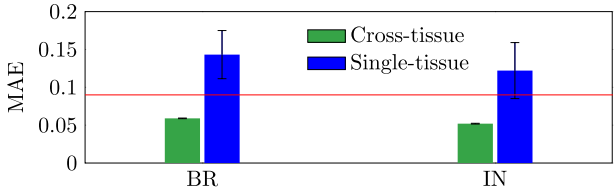

Fig. 6 Performance of *TransPlatformer* translating from CodeLink to BioSpyderWT for BR and IN. The horizontal red line at 0.09 shows the MAE using ToxCompl

Table 8 PCC between prediction and ground truth with *TransPlatformer* for translating from CodeLink to BioSpyderWT

Mode	BR	IN
Mixed-tissue	0.20 ± 0.005	0.24 ± 0.01
Single-tissue	− 0.40 ± 0.07	0.05 ± 0.02
Cross-tissue*	**0.60 ± 0.006**	**0.50 ± 0.03**

$T = \{BR\}$. Cross-tissue mode assumes that some fundamental relationships among genes are preserved across tissues and can be leveraged for translation.

We train a model on data from the set of source tissues $S = \{LI, KI, HE, SM\}$, and test on each of the two data-scarce target tissues, BR and IN. Note that data from the target tissues are completely withheld from training.

Figure 6 shows the MAE for cross-tissue models compared to single-tissue models. Clearly cross-tissue models perform much better than single-tissue models. For BR, the improvement in MAE is about 2.42 times, and for IN, the improvement is about 2.34 times.

Figure 6 suggests that the translation mechanism learned from one set of tissues may be applied to another set. To further improve the performance of the model, we include a very small fraction of the data (e.g., one data sample) from the target tissue in training. The performance results are shown in Table 8. In Table 8, *Cross-tissue** is the same as the cross-tissue training for BR and IN except that the training data are augmented with one BR or IN sample that is replicated so that there are equal amounts of BR/IN data and data from other tissues. This strategy can be considered as a form of transfer learning. That is, the translation mechanism is learned on tissues with plenty of data and fine tuned on tissues with few data samples. Oversampling BR or IN improves PCC to 0.60 for BR and 0.50 for IN.

Obviously there is a limit to the performance improvement from training strategies. For IN and BR, we will explore alternative approaches that are beyond the scope of this paper in future work.

Distribution comparison of PCC

We plot the distribution of PCC between 15 randomly chosen pairs of translated transcriptomic profiles and the corresponding ground truth in the test set. For this experiment we use the single-tissue model with LI. The translation is from CodeLink to BioSpyderWT. We compare this distribution with the distribution of PCC between two ground truth profiles of different treatments (i.e., controls) and the distribution for predicted profiles between different treatments (i.e., another type of controls). The results are shown in Fig. 7.

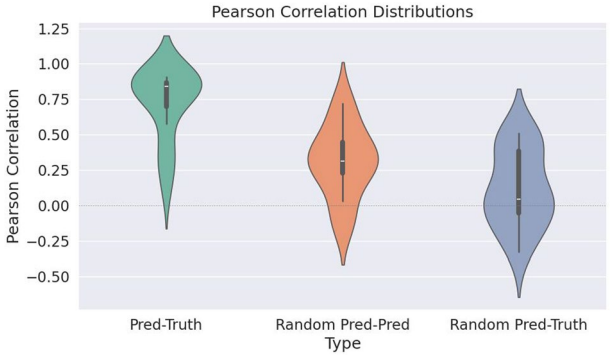

Fig. 7 Comparison of PCC distribution. On the left is the distribution between pairs of prediction and ground truth; in the middle is the distribution between two different pairs of predicted profiles; on the right is the distribution between two different pairs of ground-truth profiles

In Fig. 7, the median of the distribution for correlation between predicted and ground truth (left violin) is slightly above 0.75. For the two controls, the median of the distribution is concentrated below 0.25. This suggests that the translated profiles could represent well the real profiles. In addition, we note that the middle and right violins in the figure are highly similar. Between the two violins, the correlations are for the same set of randomly chosen pairs of treatments. Considering the inherent noise in measurements, this again demonstrates that the ground truth profiles are well represented by the translated profiles.

Toxicogenomics characterization of predicted data

TransPlatformer generates new data for all treatments that are on either CodeLink or Affymetrix but not BioSpyderWT. There are no ground-truth measurements on BioSpyderWT for these new predicted data. We conduct toxicogenomics analysis with several representative treatments for KI and LI for validation. For conciseness, the predicted data and auxiliary plots in this analysis can be found in Supplementary Materials.

Kidney data biological characterization

Kidney gene expression tends to be closely related to toxicological or histological damage, particularly acute kidney injury. Various biomarkers have been identified through various mechanisms, including gene expression profiling in treated animals. These biomarkers include genes such as Havcr1, Clu, A2m, Lcn2, Fgg, Gpnmb, Timp1, Spp1, Pvr and Mmp12 [30, 31]. All genes play a role in the complex process of wound or injury repair [32].

6-Methoxy-2-naphthylacetic acid, 3 days, 360 mg/kg/day: 6-Methoxy-2-naphthylacetic acid is a nonsteroidal anti-inflammatory drug (NSAID) derivative of naphthalene that is structurally closely related to naproxen [33]. It causes kidney toxicity by reducing prostaglandin-mediated blood flow via inhibition of cyclooxygenase enzymes and subsequent effects on the afferent arteriole [34]. This reduction in blood flow to the nephron leads to ischemia and acute kidney injury [35]. This treatment shows no histological effects in the kidney although based on the observed gene expression it would be anticipated that longer exposure would lead to histo-pathological change. In the translated BioSpyderWT responses, Fgg, Lcn2, Havcr1, and Spp1 are among the top 50 up-regulated genes. Clu, Timp1, and Gpnmb appear in the top 100 and PVR is in the top 400. In comparison the

CodeLink data showed response of only A2M, Spp1 and Lcn2 in the top 50 genes, and also Gpnmb in the top 500 genes. In this case the BioSpyderWT data demonstrates better coverage of the kidney injury biomarkers compared to CodeLink. It also highlights that the BioSpyderWT may have the ability to more effectively detect sub-histological injury.

Meloxicam, 5 days, 33 mg/kg/day: Meloxicam is a NSAID that, like 6-methoxy-2-naphthylacetic acid, produces acute kidney injury through a similar mechanism [36, 37]. The predicted expression profile of meloxicam is strikingly similar to that of 6-methoxy-2-naphthylacetic acid, both qualitatively and quantitatively, which aligns with their membership in the same chemical class. This treatment causes a slight increase in tubule regeneration. Lcn2, Fgg, Havcr1, and A2m are all among the top 50 up-regulated genes, while Spp1, PVR, and Gpnmb appear in the top 100 up-regulated genes. Clu and Timp1 are in the top 200. In comparison the CodeLink data from the same experiment shows increased expression of A2m, Lcn2, Spp1, PVR in the top 50, Gpnmb and PVR in the top 100, Clu in the top 200, Havcr1 in the top 200. Overall in this case the CodeLink data perform nearly as well as the BioSpyderWT data for identifying biomarkers of acute kidney injury.

Nimesulide, 5 days, 162 mg/kg/day: Nimesulide is a COX-2 selective NSAID that, like 6-methoxy-2-naphthylacetic acid, produces acute kidney injury, albeit at a lower rate than traditional non-selective NSAIDs [38]. Several histological lesions are observed at this dose and time point, including tubule dilatation/regeneration and papillary necrosis. In the predicted BioSpyderWT data, numerous kidney injury biomarkers are up-regulated, including Havcr1, Lcn2, Spp1, Clu, A2m, and Timp1, all among the top 10 up-regulated genes. Additional biomarkers Gpnmb, Fgg and Pvr, are in the top 35. In comparison the CodeLink data show A2m, Clu, Lcn2, Havcr1, Spp1 all within the top 10 genes along with Gpnmb, PVR in the top 35. Overall the predicted BioSpyderWT data demonstrate greater coverage of the acute kidney injury biomarkers.

Overall, the predicted BioSpyderWT responses in kidney are plausible and consistent with existing knowledge related to toxicogenomic responses to chemical challenge in kidney, in particular, acute kidney injury. Further, the translation of the BioSpyderWT platform broadens the capture of biomarkers indicative of kidney injury.

We evaluate whether additional mechanistic information would be gleaned from pathway enrichment analysis, that is, is there any additional mechanistic insight observed with the translated BioSpyderWT data compared to the original CodeLink data? We select the top 500 up-regulated probes for both platforms for each of the treatments we have studied and use those probes in Ingenuity Pathway Analysis (IPA) [39] to identify enriched pathways. When comparing the overall enrichment between the platforms for each treatment there is a striking similarity for all three treatments. A deeper evaluation of one of the most effected pathways does identify a critical arm of the pathway, specifically Map Kinase signaling, that is indicated to be down-regulated in all three treatments measured by the predicted BioSpyderWT data missing in the corresponding CodeLink data. Notably, Map Kinase signaling has a central role in the response to acute kidney injury [40].

Liver data biological characterization

There is a collection of transcriptional responses in the liver that reflect different ways chemicals elicit a response. Signatures have been developed to capture the various subtypes of mechanistic perturbations. These mechanisms include the activation of different ligand-activated transcription factors (e.g., Ahr, Car/Pxr, Pparα), activation of Nrf2 through oxidative stress, cytotoxicity/inflammation via macromolecular damage by soft electrophiles or inflammatory system activation, and DNA damage by hard electrophiles [41]. Each of these mechanisms has a distinct pattern of response at the transcriptome level [42]. Here, we evaluate the predicted responses in the liver using the predicted BioSpyderWT data. The analysis is on two Pparα activators, Pirinixic Acid and Nafenopen.

Pirinixic acid, 5 days, 364 mg/kg/day: Pirinixic acid, also known as WY-14,643, is a synthetic peroxisome proliferator-activated receptor alpha (Pparα) agonist [43]. When PPARα is activated, it increases the transcription of enzymes responsible for the β -oxidation of fatty acids in peroxisomes. Peroxisomal β-oxidation generates hydrogen peroxide, and up-regulating this pathway can lead to overproduction of reactive oxygen species (ROS) [44]. Excess ROS can overwhelm antioxidant defenses (e.g., glutathione, catalase), causing oxidative damage to proteins, lipids, and DNA in hepatocytes, ultimately leading to tissue damage [45]. The top differentially expressed genes, Acot1, Ehhadh, Cyp4a1, Hdc, Vnn1, Acaa1a, Eci1, Aqp7, Cpt1b, Acot3, Acot2, Ech1, Acox1, and Pex11a, provide a strong indication of PPARα activation, given the amplitude and coordination of their response [46]. Overall, this expression profile is highly plausible for Pirinixic acid.

Nafenopin, 5 days, 338 mg/kg/day: Nafenopin is a member of the fibric acid family and is recognized for its potent peroxisome Pparα agonist activity [47]. Like pirinixic acid, it induces liver toxicity through a similar mechanism. Unsurprisingly, the predicted gene expression profile for Nafenopin closely parallels that of Pirinixic acid, making the predicted expression profile highly plausible.

A use case of harmonizing data for bio-marker identification

To assess the potential of *TransPlatformer* for harmonizing data towards better biomarker discovery, we use liver necrosis as a case study. Accurate prediction of necrosis is critical to early diagnosis and prevention of serious liver complications [48]. In DrugMatrix we identified 1599 treatments that have associated clinical pathology data for determining liver necrosis conditions. Using 1.5 fold increase in threshold alanine transaminase (ALT) over the average of controls, 1435 treatments are negative and 164 treatments are positive for liver necrosis. The measurement data are distributed across all three platforms. Recall from Table 3 these measured profiles are also in different dimensions, preventing the construction of an AI model for biomarker discovery. Furthermore, on the most modern platform, BioSpyderWT, only 436 measurements are available, with the rest measurement data on CodeLink and Affymetrix.

To demonstrate the usage of *TransPlatformer*, we translate LI CodeLink profiles to BioSpyderWT using the single-tissue model. Table 9 shows the statistics of the data samples. There are 498 BioSpyderWT LI profiles with 67 positives and 431 negatives. In DrugMatrix there are 436 treatments that are unique to CodeLink without BioSpyderWT

Table 9 Number of LI treatments on BioSpyderWT, CodeLink (minus those appearing also on BioSpyderWT), and the combined (measured BioSpyderWT profiles plus those translated from CodeLink)

	BioSpyderWT	CodeLink unique	Harmonized
# samples	498	436	934
# positive	67	52	119
# negative	431	384	815

Table 10 Performance of models with data from BioSpyderWT (shown as BSWT) only and from harmonized BioSpyderWT and CodeLink data

	MLP			RF		
Metric	**BSWT**	**Harmonized**	**Harmonized***	**BSWT**	**Harmonized**	**Harmonized***
Accuracy	0.901	0.939	**0.977**	0.886	0.930	0.941
Precision	0.778	0.875	0.778	**1.000**	0.856	0.786
Recall	0.538	0.609	**1.000**	0.333	0.630	0.579
F1	0.636	0.718	**0.875**	0.500	0.723	0.667
Accuracy	0.901	0.939	**0.977**	0.886	0.930	0.941
Precision	0.778	0.875	0.778	**1.000**	0.856	0.786
Recall	0.538	0.609	**1.000**	0.333	0.630	0.579
F1	0.636	0.718	**0.875**	0.500	0.723	0.667

In the table the Harmonized* columns are for testing on 20% of BioSpyderWT data with a model trained with 80% of BioSpyderWT data harmonized with CodeLink data

counterparts. We translate those to BioSpyderWT using *TransPlatformer*. Of these, 52 are positive samples, and 384 are negative samples.

We build simple machine learning models based on the assumption that the presence of liver necrosis would be reflected in the transcriptomic profiles of corresponding treatment conditions. We first train a random forest classifier (RF) with 200 estimators and max tree depth 10. We then build an MLP model that has an input $22,794 \times 128$ layer followed by 3 hidden layers of size 128×128, and the output is 128×2. We then apply Sigmoid to the result and use binary cross-entropy loss for the loss function. Each input and hidden layer is followed by activation (GELU) and dropout (0.5).

We first use 8:2 as the train:test split with only BioSpyderWT data. The results are shown in the BioSpyderWT columns of Table 10. In Table 10 the model performance metrics, accuracy, precision, recall, and F1 score, are shown for both MLP and RF. Although the models have an accuracy of 0.906 and 0.886 respectively, they mispredict many positive samples as evidenced by a low recall score of 0.538 and 0.333, respectively. The F1 score is 0.636 for MLP and 0.500 for RF.

We retrain the same model with the same hyper-parameters on the harmonized data. The results are shown in the *Harmonized* columns in Table 10. All performance metrics improve for MLP and the F1 score is 0.718, a 12.9% improvement from 0.636. For RF Recall and F1 score improve significantly. We point out that a more sophisticated model, for example, one that is similar to the encoder part of *TransPlatformer*, may very well achieve higher performance for this classification task.

Discussion

Beyond translating profiles across platforms, the attention mechanism in *TransPlatformer* reveals genes that the models deem important in making correct predictions. We also compare in detail *TransPlatformer* against other baselines in ablation studies. We

further discuss how *TransPlatformer* can be leveraged translating toxicogenomic profiles beyond those in DrugMatrix, e.g., profiles in open-TGGATES to modern platforms.

Important genes according to *TransPlatformer* models

The attention mechanism in *TransPlatformer* not only contributes to accurate translation, it can also reveal the sets of genes that are important for such translation. Analysis of attention scores for individual input genes shows that *TransPlatformer* pays much more attention to a small subset of genes than the rest.

We train *TransPlatformer* without the bottleneck layer in mixed-tissue mode and then perform inferencing on the full dataset (both train and test sets). We inspect the average attention score paid for each gene in the input vectors by *TransPlatformer* over the full dataset and for all target genes. Figure 8 shows the relative amount of attention paid to each of the 8,565 CodeLink genes. The attention scores are normalized against the highest one(s). In this experiment, the highest attention is given to *Atf3*. Atf3 is a gene that encodes a protein involved in regulating gene expression in response to various stress signals, including DNA damage, oxidative stress, heat shock, and inflammatory responses [49]. The attention score for *Atf3* is about 100 times higher than the mean score of all genes.

Figure 8 highlights genes with attention score 10 times higher than the mean. The 10 genes with the highest attention score in increasing order are *P010960*, *Timm8a*, *Afm*, *Lcn2*, *Dspp*, *Ctsh*, *Fos*, *Gal*, *Stac3*, and *Atf3*. P010960 is the CodeLink probe id used in DrugMatrix and later identified as gene *Zbtb16*.

Analysis of attention scores is also done for each individual tissue in single-tissue training. Table 11 shows the 10 genes with the highest attention score for each tissue. Some of these top-10 genes are common among tissues. For example, the *Ctsh* gene is common to both LI and KI, and *Rufy1* is common to both KI and HE. For ease of identification of these genes, we give each gene appearing in more than one tissue a unique color. *Ctsh* encodes Cathepsin H, and is involved in intracellular protein degradation as a lysosomal cysteine protease [50], and plays a role in processes such as spermatogenesis [51] and cellular response to thyroid hormone [52], with potential implications in atherogenesis and tumor progression [53]. *Rufy1* in rats has been associated with functions such as protein transport [54], regulation of endocytosis [55], and small GTPase-mediated signal

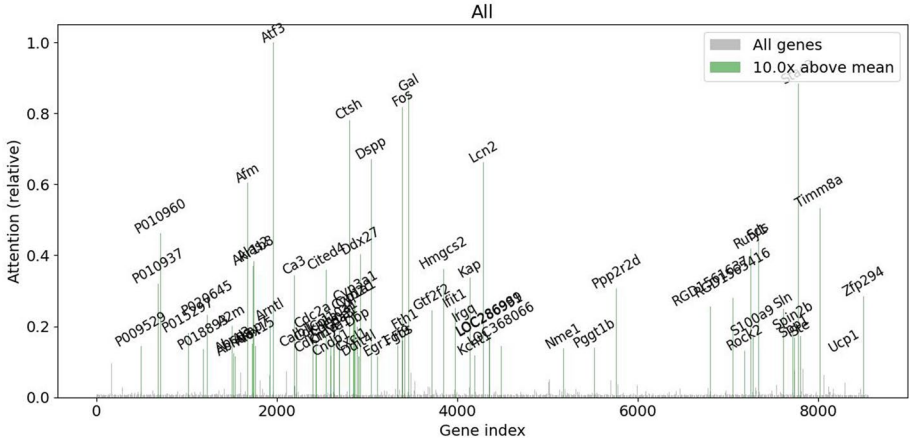

Fig. 8 Overall attention for translation of all tissues from CodeLink to BioSpyderWT

Table 11 Top-10 important genes for each tissue

Tissue	Top-10 Genes with Highest Attention Score
LI	Cmya1, P009529, Ca3, Cyp1a1, Ctsh, Sds, Cited4, Atf3, Lcn2, Stac3
KI	RGD1565416, Rufy1, Akr1b8, Timm8a, P010960, Cyp2c, Fos, Ctsh, Afm, Gal
HE	Arntl,Timm8a, , Rufy1, Kap, Atf3, Sln, Gal, Alas2, P010937, Fos
SM	G0s2, Myl3, Fabp1, Akap7, Calb3, Apln, Fabp2, Fabp2, Arrdc2, Mt1a
SP	Adra2a, Ela2, RGD1565470, Gp2, Ypel4, Clps, P001449, Dsp, Cggbp1, Fgf9
IN	Cpb1, Mars, RGD1311454, Ifit1, Sctr, RGD1561637, Rhobtb2, Pggt1b, Epha8, Ddx27
BR	RGD1309207, Kap, Cd36, Uncx4.1, Tff3, Cort, RGD1565534, Ppp2ca, Pggt1b, RGD1561637
BM	P007872, Oprk1, Blk, P020645, Cndp1, Ypel4, Hmgcs2, Col10a1, Gtf2f2, Dspp

transduction [56]. These functions are crucial for maintaining cellular homeostasis and facilitating communication between cellular compartments. The details of the attention analysis for each tissue are included in the Appendix.

Ablation studies

We conduct ablation studies to compare *TransPlatformer* with MLP and Seq2Seq transformer to demonstrate its advantages.

A Seq2Seq transformer treats the inputs and outputs as sequences. Take translating from CodeLink to BioSpyderWT as an example. The input is a sequence of 8,565 gene expressions plus one start of sequence 'token', in this case, we use 12 since that is distinctive from all values in the sequence. The output is a sequence of 22,794 gene expressions. During training, the target sequence is provided as input to the decoder, shifted one position to enable autoregressive learning. At inferencing, the decoder iteratively predicts the next number, using the decoder's own previous predictions. Typically, decoding stops when <EOS> is generated or the maximum sequence length is reached.

As discussed in Sect. 3, genes form networks and pathways instead of a linearly ordered sequence. By the nature of toxicogenomics, many genes in the profiles of two different treatments can share similar values. Due to the high likelihood of the first k gene expressions unable to predict the $(k+1)^{th}$ gene, we set the maximum sequence length to 8,565.

We compare the training loss of *TransPlatformer*, MLP, and Seq2Seq transformer with mixed-tissue training. In this experiment we translate from CodeLink to BioSpyderWT. For *TransPlatformer* and MLP the model parameters are as described in Sect. 4. For Seq2Seq transformer, the memory consumption increases quadratically with the input sequence length due to the self-attention mechanism where the output (embedding) for each 'token' attends to every input 'token' (embedding), creating a matrix of attention scores with size linear to the square of the sequence length. With a maximum sequence length 8,565 and batch size 8, the number of layers in our Seq2Seq transformer model that can be supported on our target NVIDIA Tesla V100S-PCIE-32GB GPU is 2.

Figure 9 shows the loss of MLP, TP, and Seq2Seq transformer for 100 training epochs. After the first epoch, MLP achieves the lowest training loss among the three, and *TransPlatformer* achieves the highest training loss. The loss of the Seq2Seq transformer is between the two. As training progresses, the MLP plateaus. The Seq2Seq transformer loss nearly reduces by half after the second epoch, and then also plateaus. Only the *TransPlatformer* loss decreases steadily until epoch 100.

Figure 10 shows the MAE achieved for all genes and for those over- and under-expressed ones. In both bases *TransPlatformer* is the best performing model among

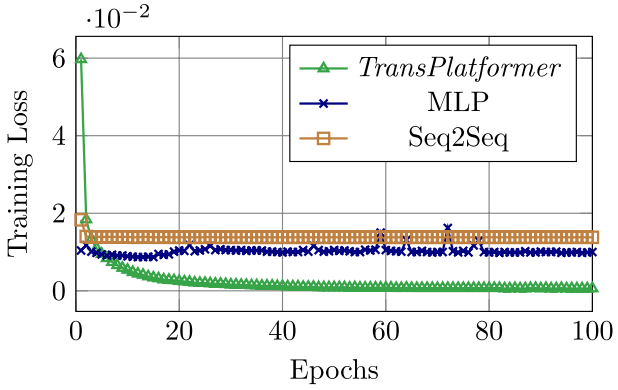

Fig. 9 Comparison of training loss for MLP, *TransPlatformer*, and Seq2Seq Transformer

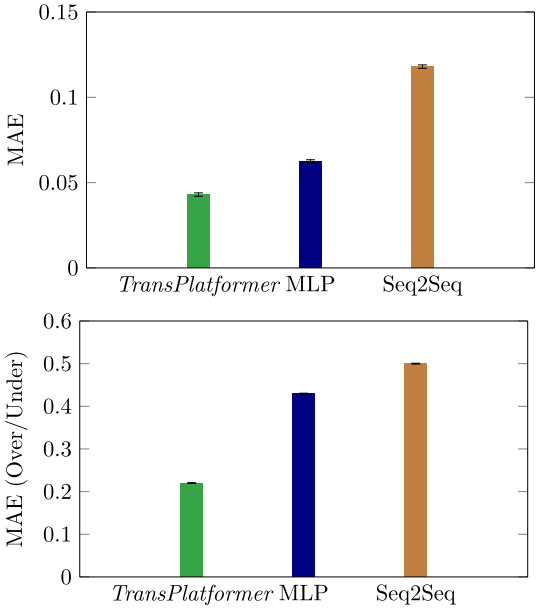

Fig. 10 Performance of MLP, *TransPlatformer*, and Seq2Seq transformer translating from CodeLink to BioSpyder-WT. The plot on the left shows the overall MAE achieved and the plot on the right shows the MAE for over and under expressed genes

the three. Seq2Seq transformer achieves worse MAE in general and for over and under expressed genes than MLP. The performance of *TransPlatformer* is 2.74 times and 2.27 times better than Seq2Seq transformer for overall MAE and MAE for rare signals, respectively.

Figure 11 shows the PCC between prediction and ground truth for the three implementations, and their inference time for test data sets. Again, *TransPlatformer* is the best implementation, followed by Seq2Seq transformer and MLP. Interestingly, the PCC of Seq2Seq transformer is higher than that of MLP. Recall that Fig. 10 shows that Seq2Seq transformer yields higher MAE than MLP. This suggests that the Seq2Seq transformer captures the general trend in its prediction but at the same time exhibits larger biases than *TransPlatformer* and MLP.

Seq2Seq transformer incurs long inference time, as shown in the plot on the right of Fig. 11. In our experiments, inferencing for all test inputs takes about 23.5 h. In

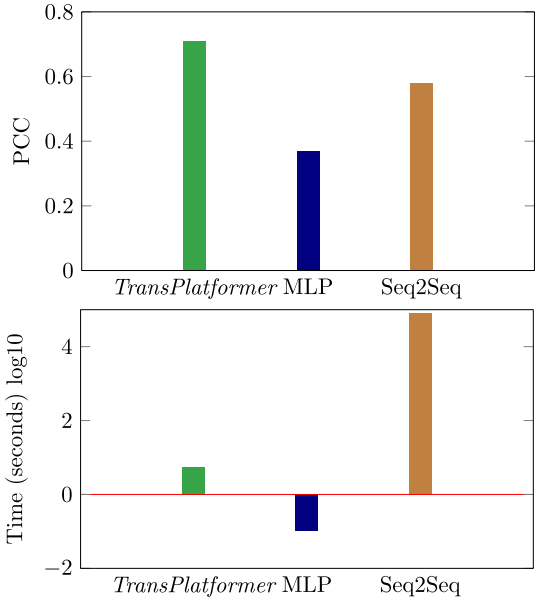

Fig. 11 Performance of MLP, *TransPlatformer*, and Seq2Seq transformer translating from CodeLink to BioSpyderWT. The plot on the left shows the Pearson correlation coefficient between prediction and ground truth and the plot on the right shows the inference time. Note that the plot on the right uses log plot for the y-axis. MLP has an inference time 0.09 and the corresponding bar has a different direction that the rest

comparison, producing the same number of outputs with *TransPlatformer* takes a few seconds. MLP is the fastest of the three, and takes less than one second in inferencing.

Translating profiles outside of DrugMatrix

In theory the trained *TransPlatformer* models can be used to translate profiles beyond those in the current DrugMatrix database if they are in the same data format (i.e., log10 fold-change). Thus Open-TGGATEs profiles can be modernized using *TransPlatformer* to the BioSpyderWT platform.

In practice, due to the quirks of transcriptomics data, especially batch effects, data collected and processed from different studies may require re-normalization and/or model calibration. For example, in the case of using *TransPlatformer* models trained on DrugMatrix to translate Open-TGGATEs profiles, one can calibrate the model on a few (source profile, target profile) data pairs. Unfortunately, there is no BioSpyderWT profile in Open-TGGATEs, making such calibration impossible. One can also evaluate from a bio-statistics perspective the systematic differences between the two databases and re-normalize the Open-TGGATEs data. This work is beyond the scope of this paper.

Conclusion and future work

We present *TransPlatformer* for translating toxicogenomic profiles between different platforms. Using data curated from DrugMatrix, *TransPlatformer* is able to effectively translate profiles between any two of the three generations of platforms, CodeLink, Affymetrix, and BioSpyderWT. We train *TransPlatformer* with three strategies, mixed-tissue mode, single-tissue mode, and cross-tissue mode, for maximizing the model performance. For LI, KI, HE, and SM, *TransPlatformer* achieves low MAE and high PCC (> 0.7) between prediction and ground truth. In mixed-tissue mode, *TransPlatformer* reduces the overall MAE by more than 50%, that is, 0.043 vs. 0.09, and doubles the PCC,

≈0.71 vs. 0.37, relative to the MLP and ToxCompl baselines. *TransPlatformer* also preserves rare but biologically important over- and under-expressed signals with MAE <0.22, significantly lower than 0.43 for MLP and 0.30 for ToxCompl. Single-tissue models yield further gains for well-represented tissues, for example, the MAE for LI is further reduced by 10%. For tissues with very limited data, *TransPlatformer* with cross-tissue training using an oversampling strategy achieves a low MAE and a PCC of above 0.5.

In addition to typical machine learning validation and test, we analyze predictions from a toxicology perspective where ground truth is not available. We find that the predicted responses in both kidney and liver are plausible and consistent with existing knowledge of toxicogenomic responses to chemical challenge in the tissue. Further, we demonstrate that the predicted data, particularly when data are scaled to a more comprehensive platform (e.g. BioSpyderWT), provide a more detailed representation of molecular pathway level processes.

Our ablation studies compare the overall MAE, MAE for rare signals, PCC between prediction and ground truth, and inference time of *TransPlatformer*, MLP, and Seq2Seq transformer. Our experiments show that the simple architecture of MLP does not perform well for input data with high dimensions and skewed distribution, and Seq2Seq transformer exhibits systematic bias although it captures the general trend in the learning task. One serious disadvantage of Seq2Seq transformer is its very long inference time due to auto-regressive output generation. Seq2Seq transformer in inferencing is orders of magnitude slower than *TransPlatformer*.

TransPlatformer is also able to reveal the important genes for translation through the attention scores of each individual gene. In future studies such insights can be further explored to help reveal the various gene networks at play in these toxicology experiments.

In future work we will continue to expand our training data set to balance the skewed distribution of data to different tissues. Since Affymetrix data are missing entirely in DrugMatrix for some tissues and CodeLink and BioSpyderWT cover all eight tissues, if we are to predict for Affymetrix, we need to incorporate new strategies such as leveraging generative methods or ToxCompl. We will explore adapting the *TransPlatformer* architecture to other tasks such as cross-tissue prediction and translation. We will also explore the modernization of other databases such as Open-TGGATEs using *TransPlatformer*.

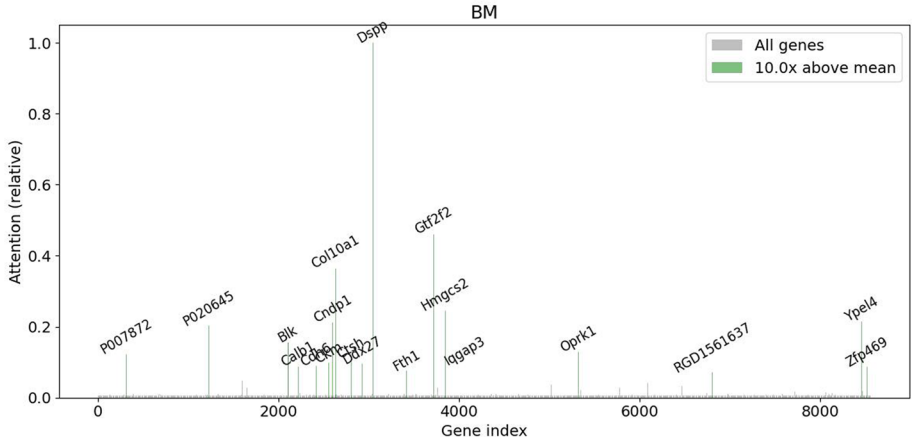

Fig. 12 Attention for translation of BM tissue transcriptomics from CodeLink to BioSpyderWT

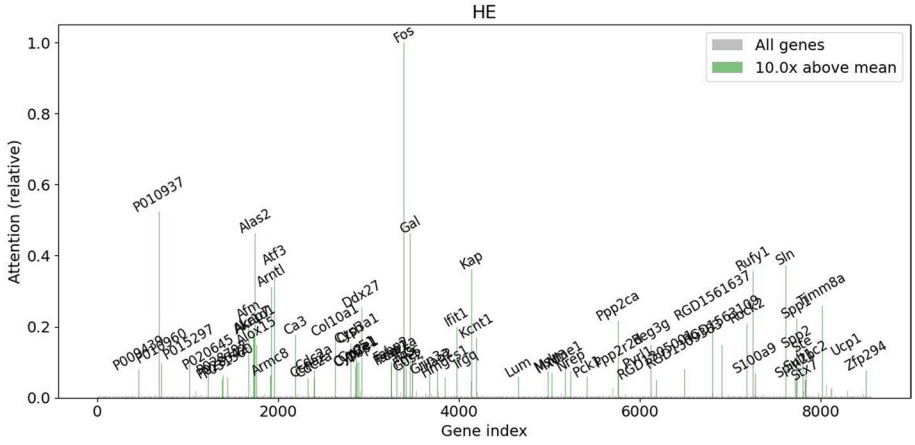

Fig. 13 Attention for translation of HE tissue transcriptomics from CodeLink to BioSpyderWT

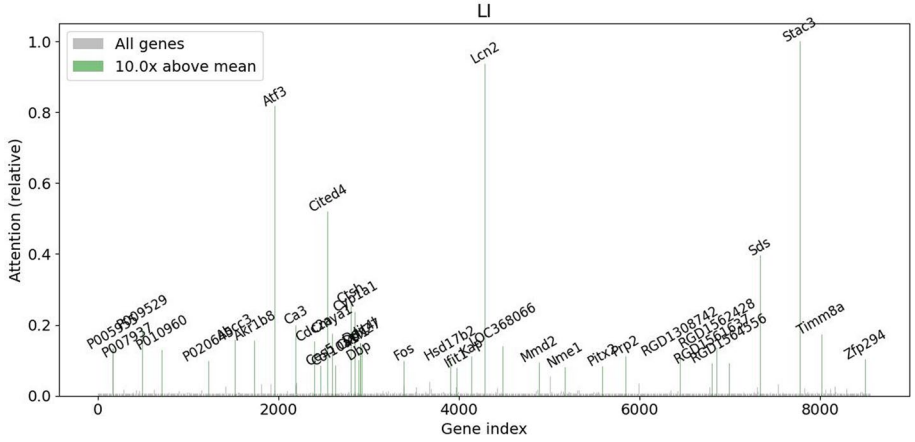

Fig. 14 Attention for translation of LI tissue transcriptomics from CodeLink to BioSpyderWT

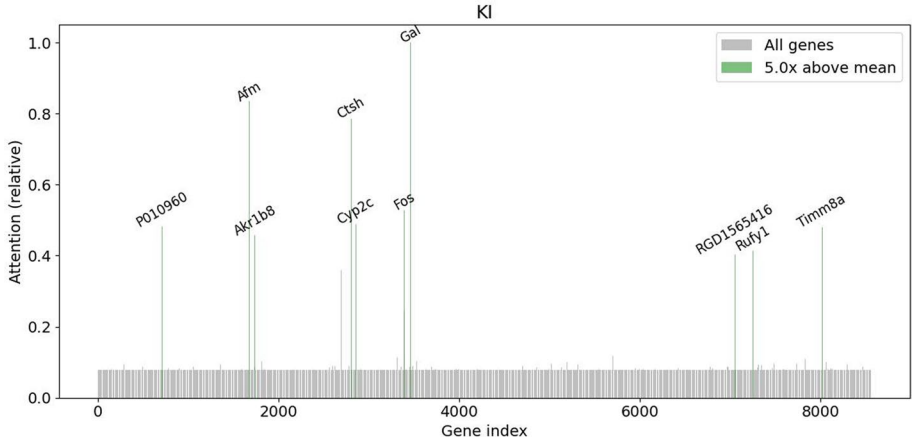

Fig. 15 Attention for translation of KI tissue transcriptomics from CodeLink to BioSpyderWT

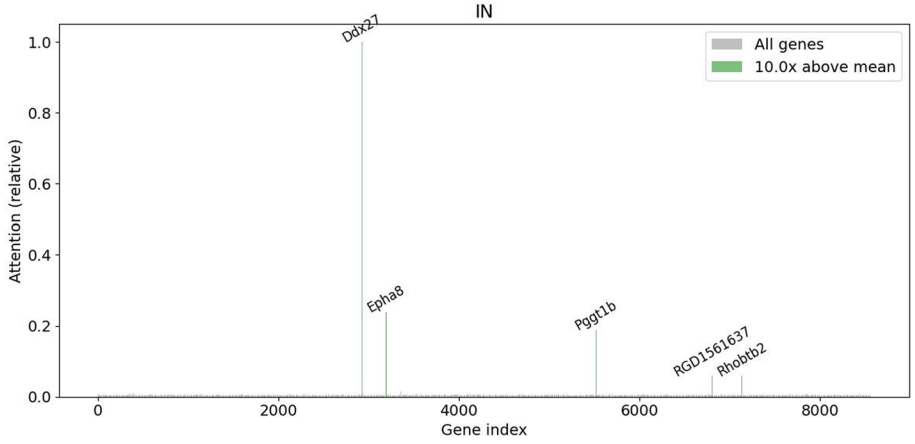

Fig. 16 Attention for translation of IN tissue transcriptomics from CodeLink to BioSpyderWT

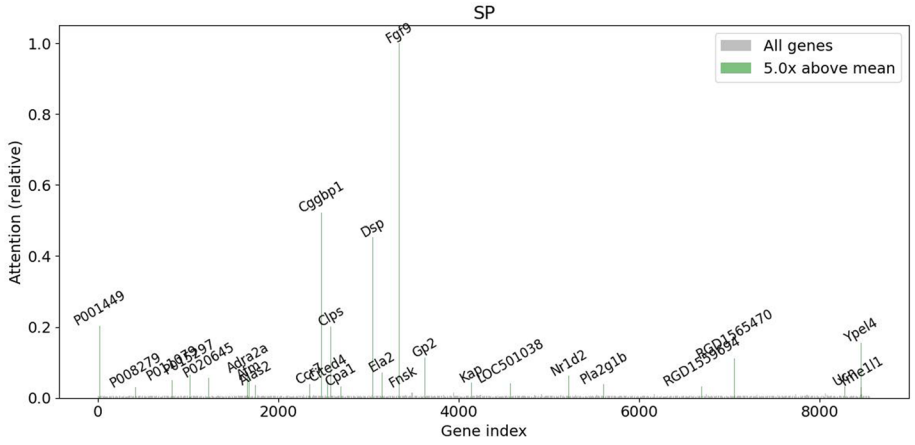

Fig. 17 Attention for translation of SP tissue transcriptomics from CodeLink to BioSpyderWT

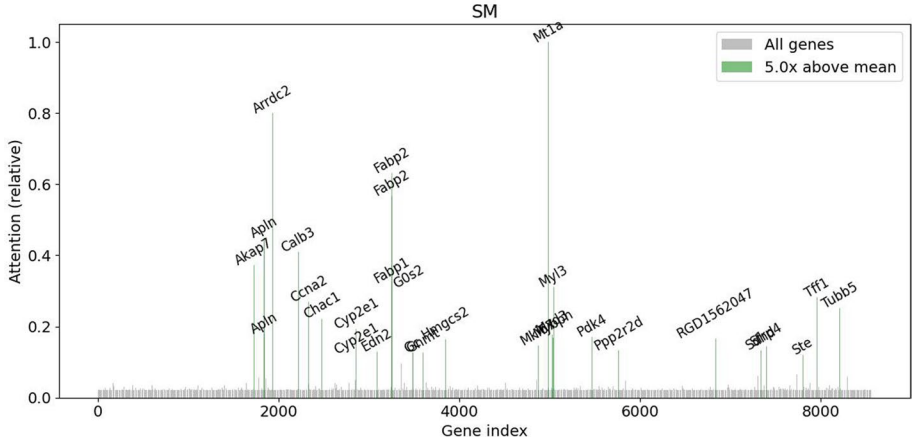

Fig. 18 Attention for translation of SM tissue transcriptomics from CodeLink to BioSpyderWT

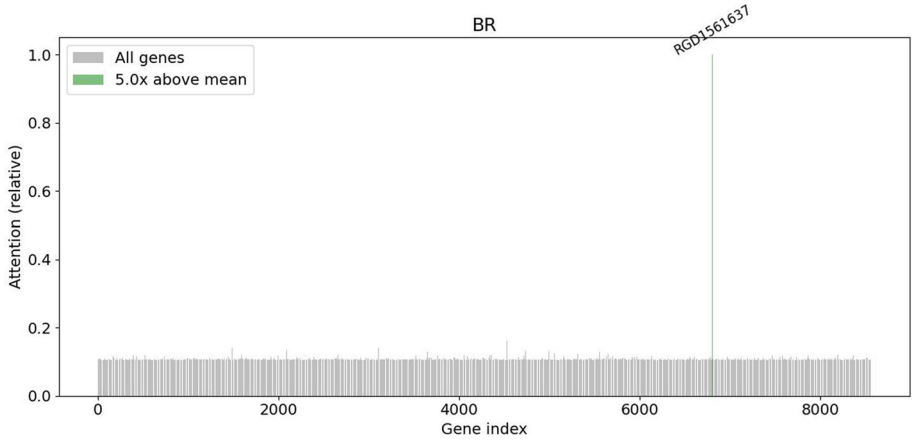

Fig. 19 Attention for translation of BR tissue transcriptomics from CodeLink to BioSpyderWT

A: Appendix

We analyze the attention scores of *TransPlatformer* in single-tissue training. Similar to the findings in Sect. 5.1 for mixed-tissue training, attention scores for individual input genes show that *TransPlatformer* pays much more attention to a small subset of genes

than the rest.

Figure 12 is for single-tissue training with BM data. Highlighted are the genes whose attention score is 10 times above average. The top ten most important genes are P007872, Oprk1, Blk, P020645, Cndp1, Ypel4, Hmgcs2, Col10a1, 'Gtf2f2', and Dspp.

Figure 13 is for single-tissue training with HE data. Highlighted are the genes whose attention score is 10 times above average. The top ten most important genes are Timm8a, Arntl, Rufy1, Kap, Atf3, Sln, Gal, Alas2, P010937, 'Fos'

Figure 14 is for single-tissue training with LI data. Highlighted are the genes whose attention score is 10 times above average. The top ten most important genes are Cmya1, Ca3, Cyp1a1, Ctsh, Sds, Cited4, Atf3, Lcn2, Stac3, and P010960.

Figure 15 is for single-tissue training with KI data. Highlighted are the genes whose attention score is 5 times above average. The top ten most important genes are Cmya1, RGD1565416, Rufy1, Akr1b8, Timm8a, Cyp2c, Fos, Ctsh, Afm, Gal, and P010960.

Figure 16 is for single-tissue training with LI data. Highlighted are the genes whose attention score is 10 times above average. The top ten most important genes are Cpb1, Mars, RGD1311454, Ifit1, Sctr, RGD1561637, Rhobtb2, Pggt1b, Epha8, and Ddx27.

Figure 17 is for single-tissue training with LI data. Highlighted are the genes whose attention score is 10 times above average. The top ten most important genes are Adra2a, Ela2, RGD1565470, Gp2, Ypel4, Clps, P001449, Dsp, Cggbp1, and Fgf9.

Figure 18 is for single-tissue training with SM data. Highlighted are the genes whose attention score is 5 times above average. The top ten most important genes are G0s2, Myl3, Fabp1, Akap7, Calb3, Apln, Fabp2, Fabp2, Arrdc2, and Mt1a.

Figure 19 is for single-tissue training with BR data. Highlighted are the genes whose attention score is 5 times above average. The top most important genes is RGD1561637.

Abbreviations

TempO-Seq	Templated oligo-sequencing
BioSpyderWT	BioSpyder whole transcriptome
MLP	Multi-layer perceptron
Seq2Seq	Sequence-to-sequence transformer
LLM	Large language model
PCC	Pearson correlation coefficient
MAE	Mean absolute error
LI	Liver
KI	Kidney
HE	Heart
SM	Skeletal muscle
BM	Bone marrow
BR	Brain
IN	Intestine
SP	Spleen

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-026-06383-6.

Supplementary Material 1

Acknowledgements

This research used resources of the Compute and Data Environment for Science (CADES) at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725. This research was also supported by the NIH, National Institute of Environmental Health Sciences, through Intramural Research Project ZIAES103385 and Interagency Agreement No. AES24003001-24-001 between NIEHS and Oak Ridge National Labs. Sciome was funded in part by Contract No. 75N96024C00003. The authors would like to acknowledge Andrew Rooney for his support and management of the interagency agreement between ORNL and NIEHS. The contributions of the NIH/NIEHS author(s) are considered Works of the United States Government. The findings and conclusions presented in this paper are those of the author(s) and do not necessarily reflect the views of the NIEHS, NIH or the U.S. Department of Health and Human Services.

Author Contributions
GC and SSA developed the initial concept. GC developed models and conducted experiments. SSA validated the results from a toxicology perspective. FC and JNE curated the DrugMatrix fold-change data. DLS, MRB-M, DM, DPP, EHS, RRS processed the measured data and conducted the necessary normalizations. PC and JNE visualize the data and make them accessible to the public. GC, SSA, and RP secured the project funding. All read and approved the manuscript.

Funding
This project receives funding from Department of Energy Advanced Scientific Computing Research, and NIH and National Institute of Environmental Health Sciences.

Data Availability
The dataset used in this study can be accessed at: https://rstudio.niehs.nih.gov/toxcomplplus/. The generated data can be found in Supplemental Materials. The TransPlatformer code is at github.com/ goooopy/transplatformer

Declarations

Ethical Approval and Consent to Participate
Not applicable.

Consent for Publication
Not applicable.

Conflict of interest
The authors declare no Conflict of interest.

Received: 6 October 2025 / Accepted: 16 January 2026
Published online: 30 January 2026

References

1. Lockhart DJ, Dong E, Byrne MC, Follettie MT, Gallo MV, Chee MS, et al. Expression monitoring by hybridization to high-density oligonucleotide arrays. Nat Biotechnol. 2000;14(13):1675–80.
2. Affymetrix Rat Genome 230 2.0 Array. NCBI. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GPL1355
3. Wang Z, Gerstein M, Snyder M. Rna-seq: a revolutionary tool for transcriptomics. Nat Rev Genet. 2009;10(1):57–63.
4. Yeakley JM, Shepard PJ, Goyena DE, VanSteenhouse HC, McComb JD, Seligmann BE. A trichostatin a expression signature identified by tempo-seq targeted whole transcriptome profiling. PLoS ONE. 2017;12(5):0178302.
5. Borisov N, Buzdin A. Transcriptomic harmonization as the way for suppressing cross-platform bias and batch effect. Biomedicines. 2022;10(9):2318.
6. Sun X, Zhang Y, Liu C, Zheng X, Zou F. Evaluating cross-platform normalization methods for integrated microarray and rna-seq data analysis. bioRxiv 2024;2024–09.
7. Bolstad BM, Irizarry RA, Åstrand M, Speed TP. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics. 2003;19(2):185–93.
8. McCall MN, Irizarry RA. Thawing frozen robust multi-array analysis (FRMA). BMC Bioinf. 2011;12:1–7.
9. Franks JM, Cai G, Whitfield ML. Feature specific quantile normalization enables cross-platform classification of molecular subtypes using gene expression data. Bioinformatics. 2018;34(11):1868–74.
10. Skubleny D, Ghosh S, Spratlin J, Schiller DE, Rayat GR. Feature-specific quantile normalization and feature-specific mean-variance normalization deliver robust bi-directional classification and feature selection performance between microarray and rnaseq data. BMC Bioinf. 2024;25(1):136.
11. Johnson WE, Li C, Rabinovic A. Adjusting batch effects in microarray expression data using empirical bayes methods. Biostatistics. 2007;8(1):118–27.
12. Zhang Y, Parmigiani G, Johnson WE. Combat-seq: batch effect adjustment for RNA-seq count data. NAR Genom Bioinf. 2020;2(3):078.
13. Benito M, Parker J, Du Q, Wu J, Xiang D, Perou CM, et al. Adjustment of systematic microarray data biases. Bioinformatics. 2004;20(1):105–14.
14. Shabalin AA, Tjelmeland H, Fan C, Perou CM, Nobel AB. Merging two gene-expression studies via cross-platform normalization. Bioinformatics. 2008;24(9):1154–60.
15. Thompson JA, Tan J, Greene CS. Cross-platform normalization of microarray and RNA-seq data for machine learning applications. PeerJ. 2016;4:1621.
16. Chen Y, Li Y, Narayan R, Subramanian A, Xie X. Gene expression inference with deep learning. Bioinformatics. 2016;32(12):1832–9.
17. Jeon M, Xie Z, Evangelista JE, Wojciechowicz ML, Clarke DJ, Ma'ayan A. Transforming 1000 profiles to RNA-seq-like profiles with deep learning. BMC Bioinf. 2022;23(1):374.
18. Wang X, Ghasedi Dizaji K, Huang H. Conditional generative adversarial network for gene expression inference. Bioinformatics. 2018;34(17):603–11.
19. Taroni JN, Grayson PC, Hu Q, Eddy S, Kretzler M, Merkel PA, Greene CS. Multiplier: a transfer learning framework reveals systemic features of rare autoimmune disease. bioRxiv. 2018;395947
20. Schmauch B, Romagnoni A, Pronier E, Saillard C, Maillé P, Calderaro J, et al. A deep learning model to predict RNA-seq expression of tumours from whole slide images. Nat Commun. 2020;11(1):3877.
21. Svoboda DL, Saddler T, Auerbach SS. An overview of national toxicology program's toxicogenomic applications: Drugmatrix and toxfx. Adv Comput Toxicol Methodol Appl Regul Sci. 2019;141–157
22. Ganter B, Snyder RD, Halbert DN, Lee MD. Toxicogenomics in drug discovery and development: mechanistic analysis of compound/class-dependent effects using the drugmatrix® database. Pharmacogenomics. 2006.

23. GE Healthcare/Amersham Biosciences CodeLink UniSet Rat I Bioarray, layout EXP5280X2-613. NCBI. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GPL5425
24. Mav D, Shah RR, Howard BE, Auerbach SS, Bushel PR, Collins JB, et al. A hybrid gene selection approach to create the s1500+ targeted gene sets for use in high-throughput transcriptomics. PLoS ONE. 2018;13(2):0191105.
25. Igarashi Y, Nakatsu N, Yamashita T, Ono A, Ohno Y, Urushidani T, et al. Open tg-gates: a large-scale toxicogenomics database. Nucleic Acids Res. 2015;43(D1):921–7.
26. Zhou C, Li Q, Li C, Yu J, Liu Y, Wang G, Zhang K, Ji C, Yan Q, He L, Peng H. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. Int J Mach Learn Cybern 2024;1–65
27. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Adv Neural Inf Process Syst 2017;30.
28. Hendrycks D, Gimpel K. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 2016.
29. Cong G, Patton RM, Chao F, Svoboda DL, Casey WM, Schmitt CP, Murphy C, Erickson JN, Combs P, Auerbach SS. Completion of the drugmatrix toxicogenomics database using toxcompl. bioRxiv. 2024;2024–03.
30. Glaab WE, Holder D, He YD, Bailey WJ, Gerhold DL, Beare C, et al. Universal toxicity gene signatures for early identification of drug-induced tissue injuries in rats. Toxicol Sci. 2021;181(2):148–59.
31. Wang E-J, Snyder RD, Fielden MR, Smith RJ, Gu Y-Z. Validation of putative genomic biomarkers of nephrotoxicity in rats. Toxicology. 2008;246(2–3):91–100.
32. Shan D, Wang YY, Chang Y, Cui H, Tao M, Sheng Y, Kang H, Jia P, Song J. Dynamic cellular changes in acute kidney injury caused by different ischemia time. Iscience 2023;26(5).
33. Hedner T, Samulesson O, Währborg P, Wadenvik H, Ung K-A, Ekbom A. Nabumetone: therapeutic use and safety profile in the management of osteoarthritis and rheumatoid arthritis. Drugs. 2004;64:2315–43.
34. Cheng H-F, Harris RC. Cyclooxygenases, the kidney, and hypertension. Hypertension. 2004;43(3):525–30.
35. Hörl WH. Nonsteroidal anti-inflammatory drugs and the kidney. Pharmaceuticals. 2010;3(7):2291–321.
36. Bumethiak N, El-Drieny E, El-Drussi E, El-Agory M. Effect of meloxicam on hematological and kidney histopathological changes in male mice. Br J Med Med Res. 2017;21(4):1–8.
37. Klomjit N, Ungprasert P. Acute kidney injury associated with non-steroidal anti-inflammatory drugs. Eur J Intern Med. 2022;101:21–8.
38. Zaki SA, Nilofer AR, Taqi SA. Nimesulide-induced acute renal failure. Saudi J Kidney Dis Transpl. 2012;23(6):1294–6.
39. Krämer A, Green J, Pollard J Jr, Tugendreich S. Causal analysis approaches in ingenuity pathway analysis. Bioinformatics. 2014;30(4):523–30.
40. Cassidy H, Radford R, Slyne J, O'Connell S, Slattery C, Ryan MP, et al. The role of mapk in drug-induced kidney injury. J Signal Transduct. 2012;2012(1):463617.
41. Gu X, Manautou JE. Molecular mechanisms underlying chemical liver injury. Expert Rev Mol Med. 2012;14:4.
42. Wang C, Gong B, Bushel PR, Thierry-Mieg J, Thierry-Mieg D, Xu J, et al. The concordance between RNA-seq and microarray data depends on chemical treatment and transcript abundance. Nat Biotechnol. 2014;32(9):926–32.
43. Merk D, Zettl M, Steinhilber D, Werz O, Schubert-Zsilavecz M. Pirinixic acids: flexible fatty acid mimetics with various biological activities. Future Med Chem. 2015;7(12):1597–616.
44. Lismont C, Revenco I, Fransen M. Peroxisomal hydrogen peroxide metabolism and signaling in health and disease. Int J Mol Sci. 2019;20(15):3673.
45. Corton JC, Peters JM, Klaunig JE. The pparα-dependent rodent liver tumor response is not relevant to humans: addressing misconceptions. Arch Toxicol. 2018;92(1):83–119.
46. Oshida K, Vasani N, Thomas RS, Applegate D, Rosen M, Abbott B, et al. Identification of modulators of the nuclear receptor peroxisome proliferator-activated receptor α (pparα) in a mouse liver gene expression compendium. PLoS ONE. 2015;10(2):0112655.
47. Lalloyer F, Staels B. Fibrates, glitazones, and peroxisome proliferator-activated receptors. Arterioscler Thromb Vasc Biol. 2010;30(5):894–9.
48. Krishna M. Patterns of necrosis in liver disease. Clinical liver disease. 2017;10(2):53–6.
49. Liang G, Wolfgang CD, Chen BP, Chen T-H, Hai T. Atf3 gene: genomic organization, promoter, and regulation. J Biol Chem. 1996;271(3):1695–701.
50. Fonović M, Turk B. Cysteine cathepsins and extracellular matrix degradation. Biochim Biophys Acta BBA General Subj. 2014;1840(8):2560–70.
51. Duliban M, Pawlicki P, Gurgul A, Tuz R, Arent Z, Kotula-Balak M, et al. Peroxisome proliferator-activated receptor γ, but not α or g-protein coupled estrogen receptor drives functioning of postnatal boar testis—next generation sequencing analysis. Animals. 2021;11(10):2868.
52. Peng P, Chen J-Y, Zheng K, Hu C-H, Han Y-T. Favorable prognostic impact of cathepsin h (ctsh) high expression in thyroid carcinoma. Int J General Med 2021;5287–5299
53. Wu S, Huang Y, Yeh C, Tsai M, Liao C, Cheng WL, et al. Cathepsin h regulated by the thyroid hormone receptors associate with tumor invasion in human hepatoma cells. Oncogene. 2011;30(17):2057–69.
54. Kitagishi Y, Matsuda S. Rufy, rab and rap family proteins involved in a regulation of cell polarity and membrane trafficking. Int J Mol Sci. 2013;14(3):6487–98.
55. Yamamoto H, Koga H, Katoh Y, Takahashi S, Nakayama K, Shin H-W. Functional cross-talk between rab14 and rab4 through a dual effector, rufy1/rabip4. Mol Biol Cell. 2010;21(15):2746–55.
56. Mizuno-Yamasaki E, Rivera-Molina F, Novick P. Gtpase networks in membrane traffic. Annu Rev Biochem. 2012;81(1):637–59.

Publisher's Note