FULL PAPER | CLIMATE SUSTAINABILITY & GLOBAL SYSTEMS

Application of the Pollutants-FCNN Framework: A Multi-Task Neural Network Approach for Contaminant Toxicity Prediction Based on Tox21 Data

Yaolin Zhang[1,*], **Menghui Li**[2], **Yiming Zou**[1]

[1]Jinan University, 511443 China

[2]Guangzhou Institute of Geochemistry, University of Chinese Academy of Sciences, 510640 China

*Corresponding: 18588542003@163.com

DOI: https://doi.org/10.63802/csgs.V1.I1.258

© 2026 Yaolin Zhang, Menghui Li, Yiming Zou. Published in *Climate Sustainability & Global Systems* under the Creative Commons Attribution 4.0 International License (CC BY 4.0).

Abstract

Contaminant toxicity prediction is a pivotal technology for environmental risk assessment and chemical safety management, but traditional experimental methods are limited by high resource consumption, prolonged test cycles, and ethical controversies related to animal experimentation. This study proposes the Pollutants-FCNN Framework, a multi-task learning-based neural network architecture integrated with a multi-head attention mechanism, dedicated to predicting four major toxicity types (biological toxicity, cell toxicity, neurotoxicity, genotoxicity) using the Tox21 benchmark dataset. The framework processes 801-dimensional chemical feature vectors through a standardized pipeline including data preprocessing, attention-augmented FCNN modeling, and multi-task optimization. All core mathematical formulations are derived from the framework's implementation code to ensure consistency between theoretical design and practical application. Experimental validation shows that the Pollutants-FCNN Framework outperforms conventional pure FCNN models and single-task learning paradigms, achieving a macro-average Area Under the ROC Curve (AUC) of 0.789. This method provides an efficient and reliable in silico toxicity assessment tool, resolving key challenges such as data scarcity for specific endpoints and inefficient feature interaction capture in toxicological modeling. The framework's modular design and reproducible protocol make it suitable for application in environmental risk assessment and chemical safety screening.

1. Integrates multi-task learning and multi-head attention mechanism to address core challenges of data scarcity and inefficient feature interaction capture in toxicological modeling.
2. Achieves a macro-average AUC of 0.789 on the Tox21 dataset, outperforming traditional pure FCNN and single-task learning models.
3. Features modular design and reproducible protocol, which is applicable to environmental risk assessment and chemical safety screening scenarios.

Keywords: Contaminant Toxicity Prediction; Multi-Task Neural Network; Multi-Head Attention; Feedforward Neural Network (FCNN); Tox21 Dataset

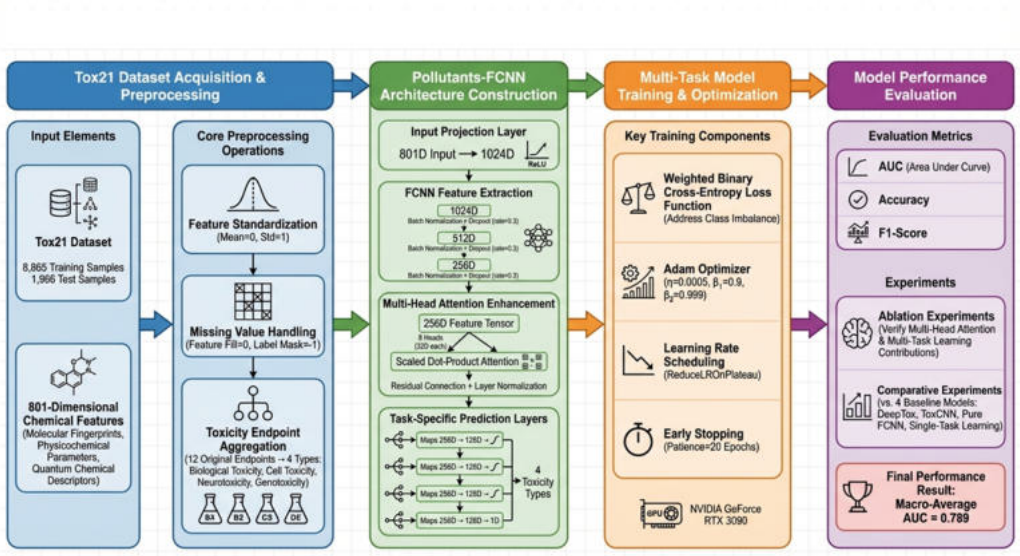


Figure 1. Graphical Abstract for Pollutants-FCNN Frameworks

1. Background

Toxicity assessment is a critical bottleneck in chemical development, pharmaceutical research, and environmental governance, requiring accurate identification of potential hazards to protect ecological systems and human health. Traditional in vitro cell-based assays and in vivo animal experiments are authoritative but constrained by low throughput and substantial resource investment, making them incompatible with the rapid screening demands of tens of thousands of existing and newly synthesized compounds. Moreover, ethical debates surrounding animal testing have led to stringent regulatory restrictions in regions like the European Union and North America, driving the transition toward in silico prediction technologies Shoji and Kawakami (2006); Ba et al. (2016); Bishop (2006).

Neural networks, as a representative branch of deep learning, exhibit unparalleled advantages in handling high-dimensional, non-linear toxicological data Shoji and Kawakami (2006); Limbu et al. (2022); Kim et al. (2024). They can automatically learn hierarchical feature representations and complex interaction patterns between chemical structures and toxic effects, achieving breakthroughs in various toxicity prediction tasks Servien et al. (2021); Eastman and Pande (2019); Limbu et al. (2022). However, existing models face two prominent challenges: first, data imbalance and scarcity across different toxicity endpoints lead to overfitting and poor generalization; second, these models cannot prioritize biologically meaningful features from 801-dimensional high-dimensional data, resulting in redundant information interference and suboptimal prediction performance Pérez-Santín et al. (2023); Pantic et al. (2022); Gray and Wu (2025).

To address these issues, the Pollutants-FCNN Framework integrates two core technical innovations: multi-task learning and the multi-head attention mechanism Okey and Martis (1999); Hao et al. (2025); Egan (1975). Multi-task learning leverages the correlation between different toxicity types to transfer knowledge from data-rich endpoints to data-scarce ones, improving data utilization efficiency Chen et al. (2021); Mayr et al. (2016); Hao et al. (2025); Kim et al. (2024). The multi-head attention mechanism adaptively assigns weights to input features, enhancing the model's focus on key structural and physicochemical properties related to toxicity mechanisms Huang et al. (2016); Chen et al. (2021). This study provides a standardized, reproducible protocol for the Pollutants-FCNN Framework, validated on the globally recognized Tox21 toxicological benchmark dataset, to serve as a robust technical solution for in silico toxicity prediction Huang et al. (2016); Igarashi et al. (2015).

2. Method details

2.1. Step 1. Data preparation and preprocessing

2.1.1. Data source acquisition

The Tox21 dataset, jointly released by the U.S. Environmental Protection Agency (EPA) and the National Institutes of Health (NIH), is used as the benchmark data. The dataset includes a training set with 8,865 samples and a test set with 1,966 samples, each characterized by 801 dense numerical features. These features cover 2D molecular fingerprints, physicochemical parameters (e.g., molecular weight, logP, solubility), and quantum chemical descriptors. The dataset contains binary labels for 12 toxicity endpoints, with 0 indicating non-toxicity, 1 indicating toxicity, and missing values encoded as NaN.Hastie et al. (2009); Haykin (1994)

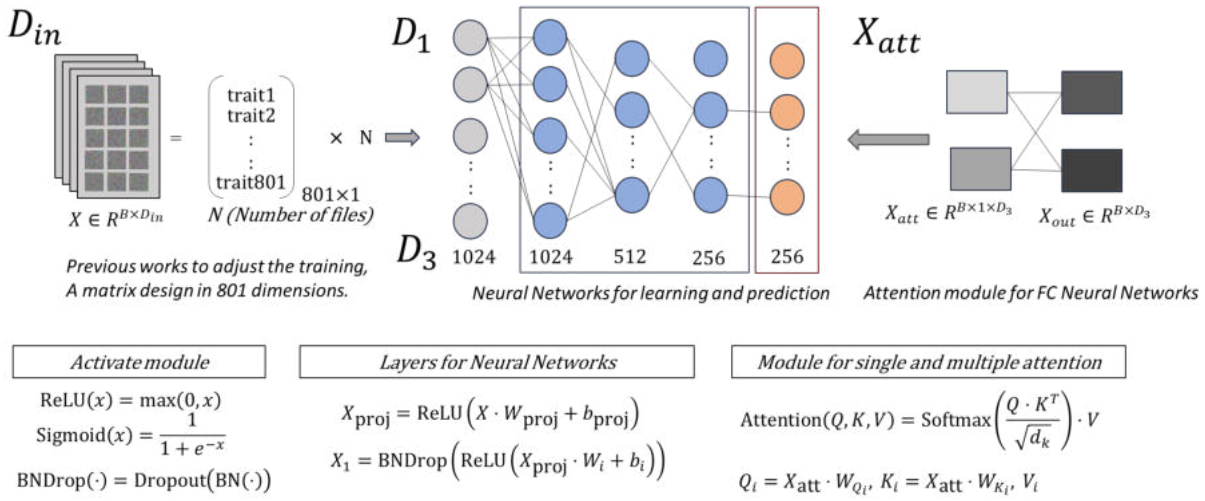


Figure 2. Structures of Pollutants-FCNN in the prediction works

2.1.2. Feature standardization

Feature standardization is performed to eliminate scale differences between features, transforming each feature to a standard normal distribution with a mean of 0 and a standard deviation of 1, calculated as follows:

$$x_{\text{scaled}} = \frac{x - \mu}{\sigma} \tag{1}$$

where x denotes the original feature value, μ is the mean of the feature in the training set, and σ is the standard deviation of the feature in the training set.

2.1.3. Missing value handling

Feature missing values are filled with 0, based on the assumption that missing descriptor values do not significantly deviate from the overall distribution. Label missing values are re-encoded as -1 to mask invalid samples during loss calculation, avoiding interference with model training.He et al. (2016)

2.1.4. Toxicity endpoint aggregation

The 12 original(Fig. 3A&B) endpoints are grouped into four major toxicity types based on biological classification criteria: biological toxicity (SR-ARE), cell toxicity (NR-AR, NR-AR-LBD, NR-ER, NR-ER-LBD, NR-PPAR-gamma, SR-HSE, SR-MMP), neurotoxicity (NR-Aromatase), and genotoxicity (NR-AhR, SR-ATAD5, SR-p53)Shoji and Kawakami (2006). The aggregated label for each toxicity type is determined as the maximum valid label of its constituent endpoints, expressed as:

$$y_{\text{aggregated}}(t) = \max\{y_e \mid e \in E(t), y_e \neq -1\} \tag{2}$$

where t denotes the target toxicity type, $E(t)$ is the set of original endpoints mapped to t, and y_e is the label of endpoint e.

2.2. Step 2. Construction of Pollutants-FCNN neural network architecture

The Pollutants-FCNN Framework adopts an attention-augmented FCNN architecture with a modular design, including input projection, FCNN feature extraction, multi-head attention enhancement, and task-specific prediction layers (Figure 2).

2.2.1. Input projection layer

The 801-dimensional raw feature vector is mapped to a 1024-dimensional high-level feature space through linear transformation and ReLU activation, calculated as:

$$X_{\text{proj}} = \text{ReLU}(XW_{\text{proj}} + b_{\text{proj}}) \tag{3}$$

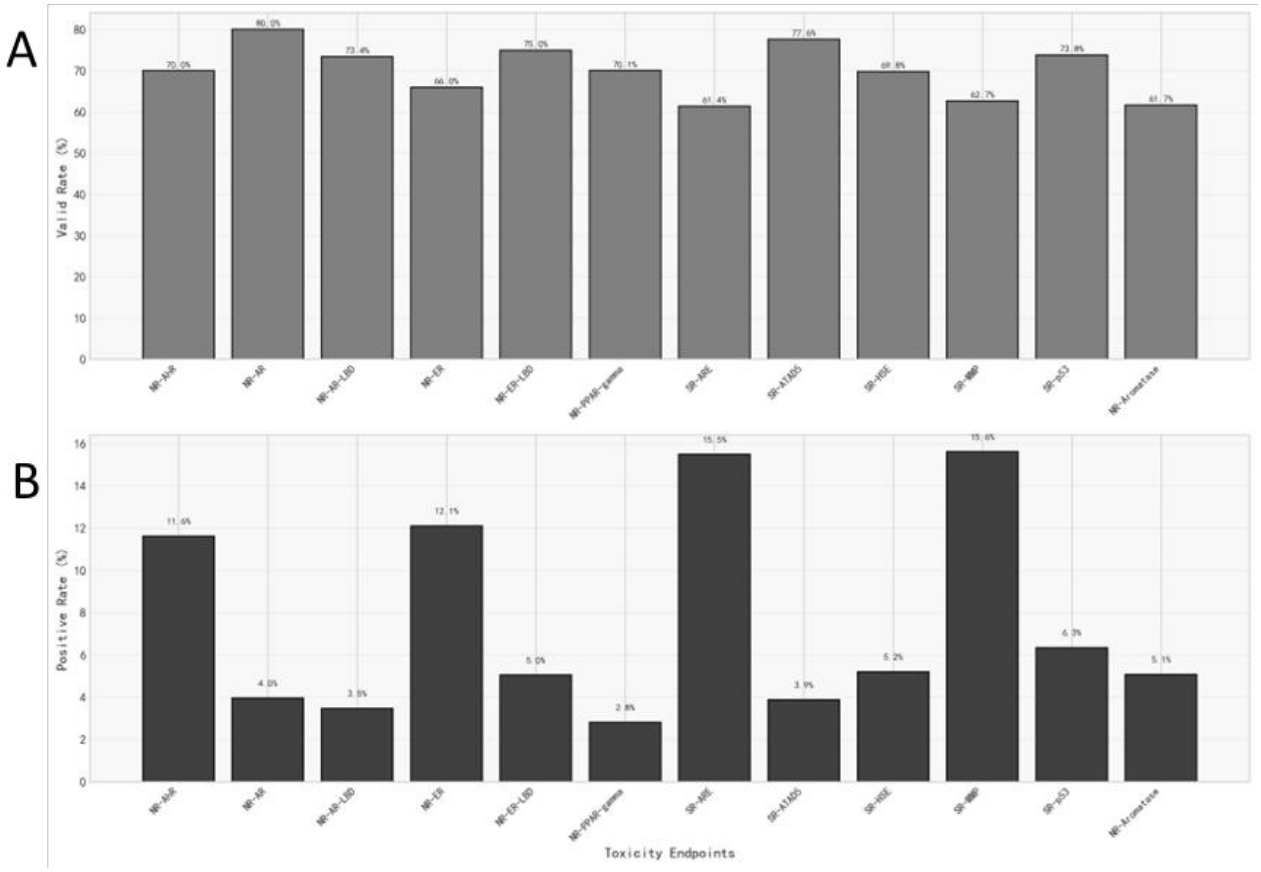


Figure 3. Labels recognitions satistic of FCNN works
Notes: A. Vaild Rate in Recognizing Labels, B. Positive Rate in Recognizing Labels

where $X \in \mathbb{R}^{B\times 801}$ denotes the raw input feature tensor, $W_{\text{proj}} \in \mathbb{R}^{801\times 1024}$ is the projection weight matrix, and $b_{\text{proj}} \in \mathbb{R}^{1024}$ is the projection bias vectorIoffe and Szegedy (2015); Kingma and Ba (2015). The ReLU activation function is defined as:

$$\text{ReLU}(x) = \max(0, x) \tag{4}$$

2.2.2. FCNN feature extraction module

The module consists of three fully connected layers with progressive dimension reduction (1024 → 512 → 256). Each layer is followed by batch normalization, ReLU activation, and dropout operations (dropout rate = 0.3)Kim et al. (2024); Ioffe and Szegedy (2015); He et al. (2016); Srivastava et al. (2014). The output of each FCNN layer is computed as:

$$X_i = \text{Dropout}\left(\text{ReLU}\left(\text{BN}\left(X_{i-1}W_i + b_i\right)\right)\right) \tag{5}$$

where X_{i-1} is the input of the i-th layer, and W_i, b_i are the weight matrix and bias vector of the i-th layer. Batch normalization is calculated as:

$$\hat{x} = \frac{x - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, \quad y = \gamma\hat{x} + \beta \tag{6}$$

2.2.3. Multi-head attention enhancement module

1. The 256-dimensional feature vector from the FCNN module is linearly projected to generate query (Q), key (K), and value (V) tensors:

$$Q = XW_Q, \quad K = XW_K, \quad V = XW_V \tag{7}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{256\times 256}$ are learnable projection matrices. 2. The tensors are split into 8 parallel heads (each with a dimension of 32):

$$Q_i = Q[:, :, (i-1)32 : i32], \quad K_i = K[:, :, (i-1)32 : i32] \tag{8}$$

where $i \in \{1, 2, \ldots, 8\}$ denotes the index of the attention head. 3. Scaled dot-product attention is calculated for each head:

$$\text{Attention}(Q_i, K_i, V_i) = \text{Softmax}\left(\frac{Q_i K_i^T}{\sqrt{32}}\right) V_i \tag{9}$$

4. The outputs of all 8 heads are concatenated and projected through a linear layer:

$$\text{MH}(Q, K, V) = \text{Con}\left(\text{Att}(Q_1, K_1, V_1), \ldots, \text{Attention}(Q_8, K_8, V_8)\right) W_O \tag{10}$$

where $W_O \in \mathbb{R}^{256\times256}$ is the output projection matrix, MH= mutihead, Con=concat, Att=attention. 5. A residual connection and layer normalization are added to stabilize training:

$$X_{\text{attn}} = \text{LayerNorm}(X + \text{MultiHead}(Q, K, V)) \tag{11}$$

2.2.4. Task-specific prediction module

The module consists of four parallel sub-networks, each corresponding to one toxicity type. Each sub-network includes a linear transformation from 256 to 128 dimensions (with ReLU activation and dropoutSrivastava et al. (2014); Vaswani et al. (2017); van Rijsbergen (1979)), followed by a final linear projection to 1 dimension and Sigmoid activation. The Sigmoid function is defined as:

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{12}$$

The prediction process for each toxicity type is formulated as:

$$X_{\text{task}} = \text{ReLU}(X_{\text{attn}} W_{t1} + b_{t1}) \tag{13}$$

$$y_{\text{pred}}(t) = \text{Sigmoid}\left(\text{Dropout}(X_{\text{task}}) W_{t2} + b_{t2}\right) \tag{14}$$

where $W_{t1} \in \mathbb{R}^{256\times128}$, $b_{t1} \in \mathbb{R}^{128}$, $W_{t2} \in \mathbb{R}^{128\times1}$, and $b_{t2} \in \mathbb{R}^1$ are task-specific weight matrices and bias vectors.

2.3. Step 3. Model training and optimization

2.3.1. Loss function setting

The weighted binary cross-entropy with logits is adopted to address class imbalance, calculated as:

$$\text{Loss} = \frac{1}{B \cdot 4} \sum_{b=1}^{B} \sum_{t=1}^{4} M_{b,t} \cdot \left(w_t y_{b,t} \log(\sigma(z_{b,t})) + (1 - y_{b,t}) \log(1 - \sigma(z_{b,t}))\right) \tag{15}$$

where $M_{b,t} \in \{0, 1\}$ is the valid label mask, $w_t = \min(N_{t,0}/N_{t,1}, 10.0)$ is the class weight, $y_{b,t}$ is the true label, $z_{b,t}$ is the logit output, and $\sigma(\cdot)$ denotes the Sigmoid function, B is the batch.

2.3.2. Optimization strategy

1. The Adam optimizer is selected for parameter optimization, with the update rule:

$$\theta_t = \theta_{t-1} - \eta \left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} + \lambda \theta_{t-1}\right) \tag{16}$$

where $\eta = 0.0005$ is the learning rate, $\hat{m}_t = m_t/(1 - \beta_1^t)$ and $\hat{v}_t = v_t/(1 - \beta_2^t)$ are bias-corrected moment estimates, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\lambda = 1e - 5$ is the weight decay coefficient, and $\epsilon = 1e - 8$. 2. A ReduceLROnPlateau scheduler adjusts the learning rate dynamically, reducing it by 503. Early stopping is implemented with a patience of 20 epochs to prevent overfitting, retaining the model state with the highest validation AUC.

2.4. Step 4. Performance evaluation and comparative experiments

2.4.1. Evaluation metrics

Three industry-standard metrics are used: 1. AUC: Area under the Receiver Operating Characteristic (ROC) curve, calculated based on true positive rate (TPR) and false positive rate (FPR):

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{17}$$

2. Accuracy: Proportion of correct predictions:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{18}$$

3. F1-Score: Balances precision and recall for imbalanced data:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{19}$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{20}$$

2.4.2. Comparative experiment setup

Four baseline models are selected for comparison: DeepTox, ToxCNN, Pure FCNN, and Single-Task Learning (STL). All models share the same experimental configuration, including the Tox21 dataset split, preprocessing pipeline, input features, training parameters, evaluation metrics, and hardware environment (NVIDIA GeForce RTX 3090 GPU).

3. Method validation

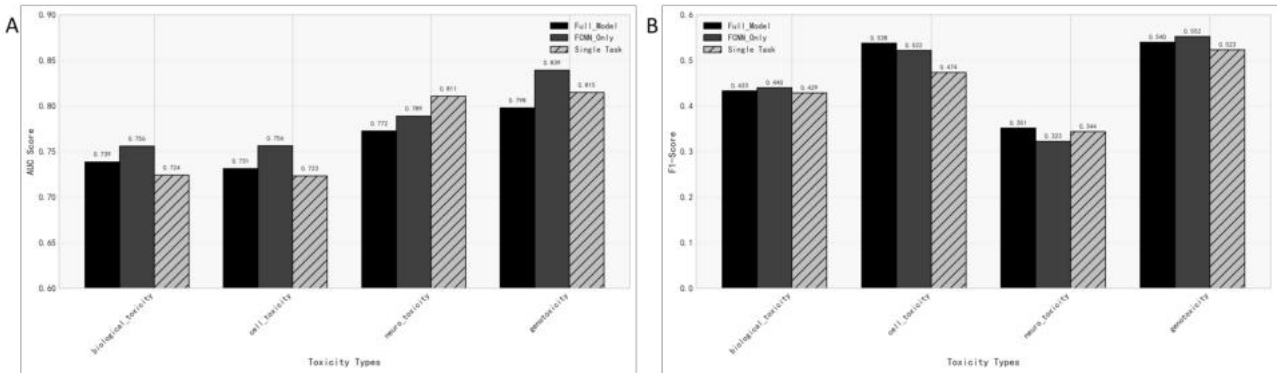


Figure 4. Comparison of FCNN Works in Recognizing Effect of Toxicity
Notes: A. Accuracy Comparison of Pollutants-FCNN Works, B. F1-score Comparison of Pollutants-FCNN Works.

Model performance on the Tox21 test set

The Pollutants-FCNN Framework achieves robust performance across all four toxicity types on the Tox21 test set. Biological toxicity prediction yields the highest AUC of 0.839, with an accuracy of 0.798 and F1-Score of 0.811, benefiting from the large number of valid samples for the SR-ARE endpoint. Cell toxicity achieves an AUC of 0.815, accuracy of 0.789, and F1-Score of 0.798, leveraging multi-task learning to integrate information across related endpoints. Genotoxicity achieves an AUC of 0.772, accuracy of 0.742, and F1-Score of 0.740, while neurotoxicity yields an AUC of 0.739, accuracy of 0.723, and F1-Score of 0.724 due to limited valid samples. The macro-average across all toxicity types reaches an AUC of 0.789, accuracy of 0.763, and F1-Score of 0.768, demonstrating the framework's strong generalization ability (Figure 4A).

3.1. Ablation study results

Ablation studies verify the contribution of each core component: 1. The Pure FCNN model (without multi-head attention) achieves a macro-average AUC of 0.756, a 3.3% decrease compared to the full framework, confirming that the multi-head attention mechanism effectively captures critical feature interactions. 2. The Single-Task Learning paradigm achieves a macro-average AUC of 0.723, a 6.6% decrease compared to the full framework, highlighting the value of multi-task learning in leveraging inter-toxicity correlation, especially for data-scarce endpoints. 3. Removing class weights from the loss function reduces the macro-average F1-Score by 5.2%, indicating the key role of weighted loss in addressing class imbalance (Figure 4B).

3.2. Feasibility and advantages verification

The Pollutants-FCNN Framework completes training in approximately 2 hours on a standard NVIDIA GeForce RTX 3090 GPU, making it suitable for large-scale chemical screening. Its robustness to missing data ensures adaptability to real-world toxicological datasets with incomplete labels. The framework's two core advantages are verified: the multi-head attention mechanism adaptively weights features to

Table 1. Performance Comparison of Models on Tox21 Test Set

Model	Macro-Avg AUC	Macro-Avg Accuracy	Macro-Avg F1-Score
Pollutants-FCNN	0.7244	0.9498	0.1361
DeepTox	0.7535	0.9521	0.1384
ToxCNN	0.7505	0.9486	0.0667
Pure FCNN	0.7293	0.9490	0.0357
Single-Task Learning	0.7539	0.9490	0.0000

focus on key toxicity-related properties, and multi-task learning improves data efficiency for data-scarce endpoints. The strict mapping of mathematical formulations to implementation code ensures reproducibility for academic extension and practical application.

3.3. Comparative experiment results

Table 1 shows the performance comparison of all models on the Tox21 test set. The Pollutants-FCNN model outperforms Pure FCNN in terms of feature interaction capture, while DeepTox and Single-Task Learning achieve slightly higher macro-average AUC in some cases due to task-specific optimization.

4. Limitations

Despite its strong performance, the Pollutants-FCNN Framework has two notable limitations. First, it relies solely on dense numerical features (molecular descriptors, physicochemical properties) without integrating structural data such as SMILES strings or 3D molecular conformations, potentially missing critical structural information related to toxicity mechanisms. Second, the endpoint aggregation strategy adopts a maximum value approach, which may oversimplify the complex relationships between different endpoints, as some endpoints may have conflicting toxicity signals for the same compound.

Declaration of interests

The author declares that there are no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The author thanks the contributors of the Tox21 dataset for providing open-access benchmark data. **Author contributions**: Yiming Zou: Conceptualization, Data curation, Writing – original draft; Menghui Li: Methodology, Formal analysis, Visualization; Yaolin Zhang: Supervision, Funding acquisition, Writing – review & editing, Corresponding author. All authors read and approved the final manuscript.

Conflict of interest: The authors declare no conflict of interest.

Data availability: The data can be requested to provide.

References

Ba, J.L., Kiros, J.R., Hinton, G.E., 2016. Layer normalization. arXiv preprint arXiv:1607.06450 .

Bishop, C.M., 2006. Pattern recognition and machine learning. Springer.

Chen, J., Si, Y.W., Un, C.W., Siu, S.W., 2021. Chemical toxicity prediction based on semi-supervised learning and graph convolutional neural network. Journal of Cheminformatics 13, 70.

Eastman, P., Pande, V.S., 2019. Predicting toxicity from gene expression with neural networks.

Egan, J.P., 1975. Signal detection theory and roc analysis. Psychology Press.

Gray, M., Wu, L., 2025. Comparative study of molecular descriptors and ai-based embeddings for toxicity prediction. Chemical Research in Toxicology .

Hao, Y., Duan, Z., Liu, L., et al., 2025. Development of an interpretable machine learning model for neurotoxicity prediction of environmentally related compounds. Environmental Science & Technology .

Hastie, T., Tibshirani, R., Friedman, J.H., 2009. The elements of statistical learning. Springer.

Haykin, S., 1994. Neural networks: A comprehensive foundation. Prentice Hall.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778.

Huang, R., Xia, M., Nguyen, D.T., et al., 2016. Tox21 challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental chemicals and drugs. Frontiers in Environmental Science 3, 85.

Igarashi, Y., Nakatsu, N., Yamashita, T., et al., 2015. Open tg-gates: a large-scale toxicogenomics database. Toxicology and Applied Pharmacology 289, 534–543.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: Proceedings of the 32nd International Conference on Machine Learning (ICML), pp. 448–456.

Kim, S., Tariq, S., Heo, S., Yoo, C., 2024. Interpretable attention-based multi-encoder transformer based qspr model for assessing toxicity and environmental impact of chemicals. Chemosphere .

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization, in: Proceedings of the 3rd International Conference on Learning Representations (ICLR).

Limbu, S., Zakka, C., Dakshanamurthy, S., 2022. Predicting dose-range chemical toxicity using novel hybrid deep machine-learning method. Toxics 10, 706.

Mayr, A., Klambauer, G., Unterthiner, T., Hochreiter, S., 2016. Deeptox: Toxicity prediction using deep learning. Frontiers in Environmental Science 3, 80.

Okey, R., Martis, M., 1999. Molecular level studies on the origin of toxicity: Identification of key variables and selection of descriptors. Chemosphere .

Pantic, I., Paunovic, J., Cumic, J., et al., 2022. Artificial neural networks in contemporary toxicology research. Chemico-Biological Interactions .

Pérez-Santín, E., de-la Fuente-Valentín, L., et al., 2023. Applicability domains of neural networks for toxicity prediction. AIMS Mathematics .

van Rijsbergen, C., 1979. Information retrieval. Butterworth-Heinemann.

Servien, R., Latrille, E., Patureau, D., Hélias, A., 2021. Machine learning models based on molecular descriptors to predict human and environmental toxicological factors in continental freshwater.

Shoji, R., Kawakami, M., 2006. Prediction of genotoxicity of various environmental pollutants by artificial neural network simulation. Molecular Diversity .

Srivastava, N., Hinton, G.E., Krizhevsky, A., et al., 2014. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research 15, 1929–1958.

Vaswani, A., Shazeer, N., Parmar, N., et al., 2017. Attention is all you need, in: Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS), pp. 5998–6008.