scientific reports

OPEN

Integrating convolutional layers and biformer network with forward-forward and backpropagation training

Ali Kianfar[1], Parvin Razzaghi[1]✉ & Zahra Asgari[2]

Accurate molecular property prediction is crucial for drug discovery and computational chemistry, facilitating the identification of promising compounds and accelerating therapeutic development. Traditional machine learning falters with high-dimensional data and manual feature engineering, while existing deep learning approaches may not capture complex molecular structures, leaving a research gap. We introduce Deep-CBN, a novel framework designed to enhance molecular property prediction by capturing intricate molecular representations directly from raw data, thus improving accuracy and efficiency. Our methodology combines convolutional neural networks (CNNs) with a BiFormer attention mechanism, employing both the forward-forward algorithm and backpropagation. The model operates in three stages: (1) feature learning, extracting local features from SMILES strings using CNNs; (2) attention refinement, capturing global context with a BiFormer module enhanced by the forward-forward algorithm; and (3) prediction subnetwork tuning, fine-tuning via backpropagation. Evaluations on benchmark datasets—including Tox21, BBBP, SIDER, ClinTox, BACE, HIV, and MUV—show that Deep-CBN achieves near-perfect ROC-AUC scores, significantly outperforming state-of-the-art methods. These findings demonstrate its effectiveness in capturing complex molecular patterns, offering a robust tool to accelerate drug discovery processes.

Keywords Deep learning, Molecular property prediction, Convolutional neural networks, BiFormer attention mechanism, Drug discovery

Molecular property prediction stands as a foundational task in bioinformatics and cheminformatics, influencing drug discovery, chemistry, and healthcare[1,2]. Effectively predicting properties of a molecule—such as bioactivity, toxicity, solubility, and permeability—can significantly accelerate drug development by limiting dependence on extensive lab-based experiments[2]. Traditional approaches, including quantitative structure–activity relationship (QSAR) and quantitative structure–property relationship (QSPR) models, have been instrumental in this domain[3,4]. However, with the advent of machine learning and artificial intelligence, there have been major enhancements in the accuracy and speed of molecular property predictions, offering appealing alternatives to traditional high-throughput screening methodologies[3,5,6]. Computational strategies for drug property prediction can be split into two categories: classical machine-learning methods and deep learning-based methods. Both approaches are data-driven, utilizing existing datasets to identify patterns within molecular data and predict various molecular properties essential for drug design and development.

Traditional machine learning algorithms, such as support vector machines (SVM)[7], random forests[8], k-nearest neighbors (kNN)[9], logistic regression[10], and XGBoost[11], are widely applied in molecular data analytics. These approaches depend on numerical representations of molecules derived from manual feature engineering processes, like chemical descriptors and molecular fingerprints. These models learn correlations between distinct features and molecular properties by extracting key molecular traits. For example, they might determine that molecules containing a particular functional group are more likely to demonstrate toxicity.

While effective, these methods often struggle with high-dimensional data and complex relationships inherent in molecular structures. Their performance heavily depends on the quality of feature engineering, a process that is often time-intensive and may neglect subtle yet significant aspects of molecular data[12]. This challenge becomes even more pronounced in bioinformatics contexts—where datasets tend to be broad and multifaceted. Hence,

[1]Department of Computer Science and Information Technology, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan, Iran. [2]School of Life Science Engineering, College of Interdisciplinary Science and Technology, University of Tehran, Tehran, Iran. ✉email: p.razzaghi@iasbs.ac.ir

there is an expanding need for methods to effortlessly learn and capture these complex patterns without relying heavily on extensive manual intervention.

One persistent challenge in molecular property prediction is the overdependence on handcrafted descriptors, which can fail to capture subtle features of complex molecular structures. Mayr et al.[13] addressed this issue by introducing DeepTox, a pioneering deep-learning-based approach for toxicity prediction that does not rely on manually crafted descriptors. By automatically learning features directly from chemical structures, DeepTox delivered remarkable results in toxicity prediction. However, another hurdle emerges in the form of comparing traditional machine learning and deep learning methods—diverse datasets and inconsistent benchmarks often make it difficult to assess performance on a level playing field. Wu et al.[14] recognized this issue and proposed MoleculeNet, a standardized benchmark designed with curated public datasets, evaluation metrics, and open-source implementations, thereby highlighting the effectiveness of learnable representations, particularly graph convolutional networks[15], in molecular machine learning tasks.

Beyond benchmarking, capturing both local and global structural information remains a critical problem. Liu et al.[16] tackled this by introducing the N-gram graph method, an unsupervised approach that uses short walks (n-grams) on the molecular graph to encode atomic properties. By combining these n-gram embeddings with individual atom embeddings, the N-Gram Graph achieved competitive performance in property prediction without complex training procedures. Yang et al.[17] further demonstrated that GCNN-based models, which analyze molecular structures directly rather than relying on predefined fingerprints, handle entirely new, unseen molecules more effectively, underscoring the importance of graph-level features. Yet, even with robust graph-based approaches, large amounts of labeled data can be scarce and expensive to acquire. Li et al.[18,19] confronted this limitation by proposing a novel pre-training strategy for GNNs that incorporates node-level and graph-level tasks, enabling GNNs to learn transferable knowledge for downstream tasks even with limited labeled data. In a similar vein, Abbasi et al.[20] explored deep transferable compound representations, affirming the advantage of such approaches for model generalization across tasks and domains under low-data conditions.

As the field continues to struggle with insufficient labeled data, self-supervised and contrastive learning methods have emerged as a potent solution. Rong et al.[19] put forth GROVER (Graph Representation from a self-supervised message-passing Transformer), which integrates transformer-driven architectures with message-passing networks to learn from large amounts of unlabeled molecular data. Meanwhile, Le et al.[21] developed MolCLR, a self-supervised learning framework leveraging GNNs to perform contrastive learning on unlabeled data, leading to improved model generalization in molecular property prediction. Ross et al.[22] extended the concept into the realm of chemical language with MoLFormer, a large-scale model that captures complex relationships by analyzing raw chemical language data. Li et al.[23] added a layer of domain knowledge through a knowledge-guided pre-training technique coupled with the Line Graph Transformer (LiGhT), further enhancing the ability to learn meaningful representations from unlabeled data.

Although many of these methods rely on two-dimensional molecular graphs, ignoring three-dimensional structures can limit their predictive power, especially for tasks that hinge on spatial arrangements. Zhou et al.[24] addressed this by developing Uni-Mol, a framework that processes the 3D molecular graph using GNNs, thus capturing the spatial arrangement of atoms to achieve superior performance in drug discovery tasks. At the same time, balancing model efficiency and interpretability proves vital. TrimNet[25] tackles this by employing a triplet message mechanism (atom-bond-atom) for a reduced-parameter yet state-of-the-art design, while FunQG by Hajiabolhassan et al.[26] leverages quotient graphs and functional groups to reduce computational overhead and explain how specific molecular components drive predictions. Similarly, Zhu et al.[27] built HiGNN to incorporate hierarchical information about molecules with a feature-wise attention mechanism, improving both performance and interpretability.

Beyond graph-based approaches, advancements in chemical language modeling have also shown promise. ChemBERTa[28] and its successor ChemBERTa-2 by Ahmad et al.[29] adapt transformer architectures from natural language processing to molecular property prediction by applying masked language modeling on SMILES representations. ChemBERTa-2 refines this process through more advanced data representations and pre-training objectives, improving scalability and overall generalization. Some frameworks go even further by integrating biological data. BioAct-Het[30], introduced by Paykan et al., presents a Siamese neural network architecture that leverages both chemical and biological data through a novel bioactivity representation (Bio-Prof) to mitigate data scarcity and enhance bioactivity prediction. Wang et al.[31] similarly addressed data limitations with BatmanNet, a bi-branch masked graph transformer autoencoder that excels in partial data scenarios by capturing both local and global molecular features through a self-supervised method. DGCL (dual-graph contrastive learning)[32] aggregates molecular features through dual-graph networks during pre-training and integrates mixed molecular fingerprints in downstream tasks to strengthen feature extraction without resorting to graph augmentation. By combining graph neural networks and contrastive learning, DGCL outperforms several self-supervised learning methods on classification and regression tasks, requiring smaller datasets and less training time. This efficiency is further enhanced by attention modulation, which assigns appropriate weights to molecular features and significantly boosts classification performance. Finally, motivated by the complexities of backpropagation, Geoffrey Hinton[32] proposed the forward-forward (FF) algorithm as an alternative training strategy for deep neural networks. By replacing traditional backpropagation with a forward-only pass, the FF algorithm offers the prospect of reduced computational demands and opens new possibilities for more efficient deep learning. Our approach stands out from other leading methods in several key ways. First, we use a multi-stage pipeline consisting of CNN-based feature encoding, attention refinement enhanced by the forward-forward (FF) algorithm, and a final prediction tuning phase. This setup ensures that SMILES representations are learned thoroughly while keeping the model's decisions more interpretable. Second, in contrast to purely graph-based approaches, our CNN-based feature encoder captures local SMILES patterns effectively without needing extra molecular graph augmentation, which helps lower computation costs and speed up training. Third, we integrate

BiFormer as our attention subnetwork to handle global context and local details in sequential data. This design is highly efficient and maintains strong performance in settings where computational resources are limited. Fourth, by freezing both the feature encoding and attention subnetworks in the final stage, we prevent overfitting and protect the representations learned earlier, enabling us to fine-tune predictions for specific tasks without disrupting the previously acquired knowledge. Finally, we conduct extensive ablation studies to confirm the importance of each component—CNN feature extraction, BiFormer attention, and multi-stage training—in improving classification accuracy and model stability. The rest of the paper is arranged as follows.

The paper is organized in the following manner. In "Method" section explains our proposed method in detail. In "Experiments" section presents the experimental results. Lastly, in "Discussion" section discusses our findings and outlines possible directions for future research.

Method

This section provides a detailed examination of the proposed method. It begins by outlining the specific problem addressed. Subsequently, the general architecture of the proposed method is dissected, providing a clear understanding of its core components and their interactions. The precise formulation of the problem is presented as follows.

Problem formulation

Suppose we have a dataset consisting of $|\mathrm{N}|$ tasks, with the data from the k-th task denoted as $\mathrm{D}^{(\mathrm{k})} = \{(\mathrm{x}_{\mathrm{i}}^{(\mathrm{k})}, \mathrm{y}_{\mathrm{i}}^{(\mathrm{k})})\}_{\mathrm{i}=1}^{\mathrm{m}^{\mathrm{k}}}$ Here, $x_i^{(k)} \in \mathcal{X}$ represents the *i*th sample from the kth task, and $y_i^{(k)} \in \mathcal{Y}^{k}$ denotes the corresponding label for $x_i^{(k)}$. Each $x_i^{(k)}$ is a drug compound examined in an experimental test with $\mathrm{y}_{\mathrm{i}}^{(\mathrm{k})}$ being the corresponding test result. The goal is to design a system that gets the drug compound as input and predicts the molecular property as output.

Model architecture

This section provides a detailed description of the proposed model's architecture. The framework comprises three main stages: Feature Foundation (components A and B in Fig. 1), Attention Amplification (component C in Fig. 2), and Refinement and Optimization. Each of these steps is demonstrated in the following.

Stage one: feature learning

This step aims to train the feature encoding network using the labeled training data in a supervised manner. The employed network in this step has two main subnetworks: feature encoding (shown by N_F) and task prediction subnetworks (shown by N_P). A convolutional neural network (CNN) is employed to design feature encoding subnetwork from the input SMILES strings.

In the initial step, the methodology involves encoding the SMILES strings of drug compounds into numerical vectors. Each SMILES string x is mapped to a sequence of integers using a predefined dictionary, where each character in the SMILES notation is assigned a unique integer. Let $R^{|d|} \in R^{|d|}$ represent the encoded SMILES string of the ith sample from the kth task, truncated or padded to a maximum length $(|d|)$.

The input layer $\mathrm{X} \in R^{|d|}$ feeds into a series of 1D convolutional layers $C^{(l)}$, where (l) denotes the layer index. Also, $|d|$ denotes the maximum length of the drug molecules. Each convolutional layer applies filters $W^{(l)}$ to capture the local dependencies within the SMILES sequence, followed by a ReLU activation function $relu\,(\cdot)$. Mathematically, the output of a convolutional layer can be expressed as:

$$C^{(l)} = relu\left(W^{(l)} * C^{(l-1)} + b^{(l)}\right) \tag{1}$$

where $(*)$ denotes the convolution operation and $b^{(l)}$ is the bias term. We used consecutive convolutional layers to expand the field of view. Then, we use global max pooling and proceed to the fully connected layers for final prediction.

Given the feature encoding subnetwork N_F and the task prediction subnetwork N_P, the loss function can be formulated for the ith sample from the kth task as follows:

Let $y_i^{(k)}$ be the true label and $\widehat{y_i^{(k)}} = N_P(N_F(x_i^{(k)}))$ be the predicted probability distribution over the classes. The categorical cross-entropy loss for the ith sample is defined as:

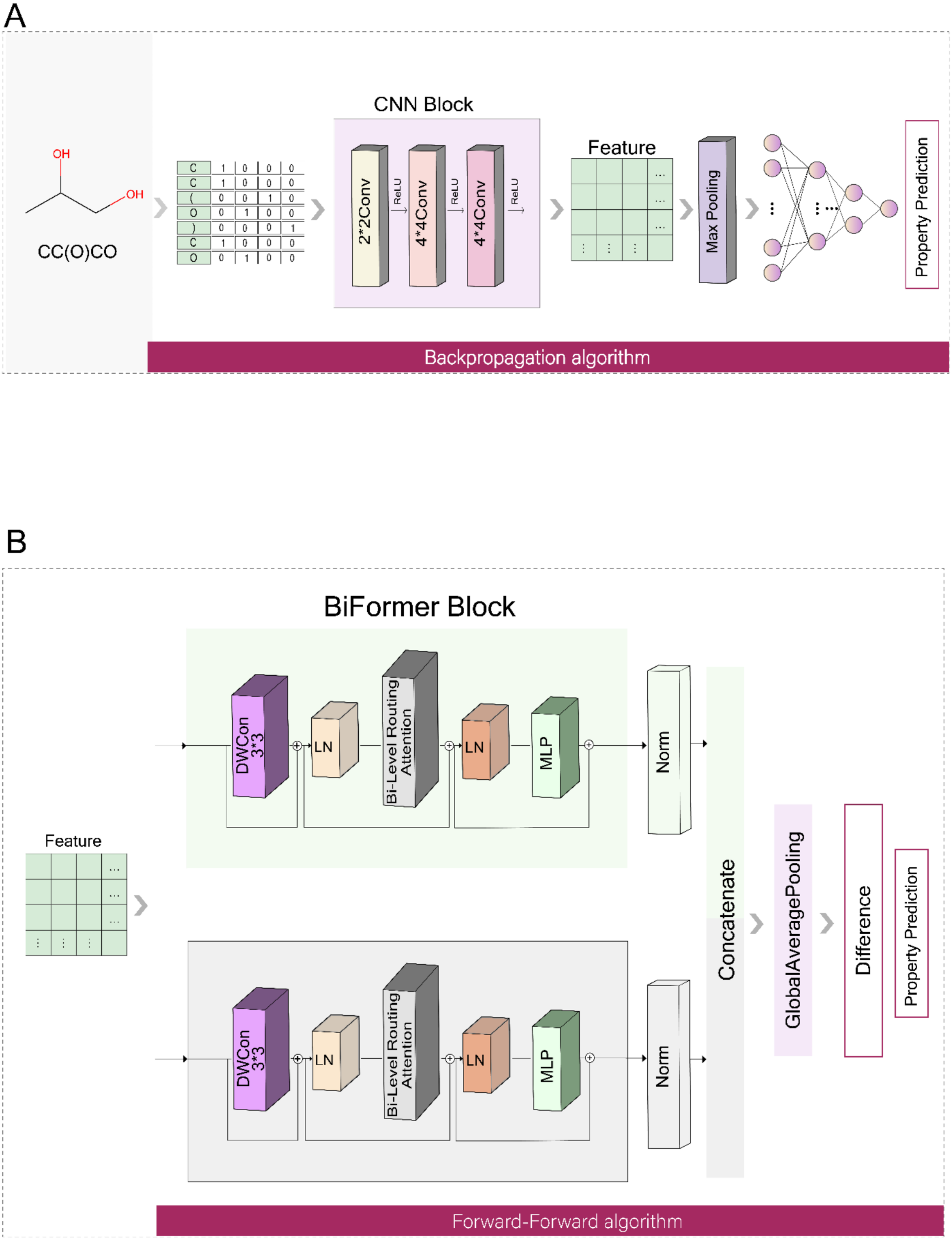
$$\mathcal{L}\left(y_i^{(k)}, \widehat{y_i^{(k)}}\right) = -\sum_{c=1}^{C} y_{i,c}^{(k)} \log \widehat{y_{i,c}^{(k)}} \tag{2}$$

where C is the number of classes, $y_{i,c}^{(k)}$ is the true label (one-hot encoded), and $\widehat{y_{i,c}^{(k)}}$ is the predicted probability for class c. The total loss is averaged over all samples and tasks.

Stage two: attention refinement

This stage builds upon the frozen feature encoding network established in stage one and introduces an attention[33] subnetwork denoted as N_T. The goal is to train the attention network using the forward-forward algorithm[32] concept and labeled training data in a supervised manner.

The feature encoding subnetwork N_F remains frozen, preserving the learned representations of the input SMILES strings. It comprises a series of 1D convolutional layers $C^{(l)}$, followed by global max pooling to extract

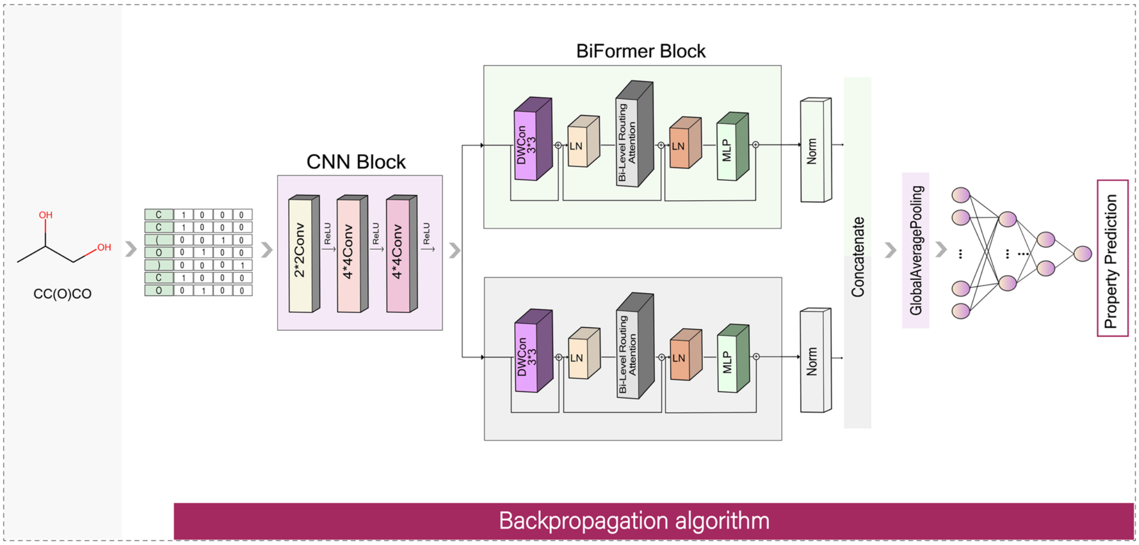


C	1	0	0	0
C	1	0	0	0
(	0	0	1	0
O	0	1	0	0
)	0	0	0	1
C	1	0	0	0
O	0	1	0	0

Fig. 1. The figure consists of two sections (**A**,**B**). Section (**A**) illustrates the first model, which utilizes convolutional neural networks (CNNs) for feature extraction, while section (**B**) demonstrates the second model, integrating an attention mechanism to improve contextual understanding and enhance molecular property prediction.

features from the SMILES sequences. These features, represented as numerical vectors, act as input to the attention subnetwork.

The attention subnetwork is introduced to enhance the learned representations by focusing on relevant parts of the input features. This subnetwork utilizes the forward-forward algorithm concept, dynamically computing



C	1	0	0	0
C	1	0	0	0
(	0	0	1	0
O	0	1	0	0
)	0	0	0	1
C	1	0	0	0
O	0	1	0	0

Fig. 2. The overall architecture of stage three—prediction subnetwork tuning, where both N_{F^*} and N_{T^*} remain frozen, and only the task prediction subnetwork N_P is fine-tuned to optimize task-specific performance without altering learned representations.

attention weights to emphasize informative features. The attention mechanism involves the computation of attention scores and their application to the value vectors. For the traditional attention module, the attention output is defined by:

$$\text{Attention}\left(Q,K,V\right)=\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)\text{V} \tag{3}$$

where $\text{Q}\in R^{n\times d_k}$, $\text{K}\in R^{n\times d_k}$, and $V\in R^{n\times d_v}$ are the query, key, and value matrices respectively, and d_k is the dimensionality of the key vectors. The softmax operation yields a matrix of attention weights of dimension $R^{n\times n}$, which, multiplied by the value matrix V, results in an output of dimension $R^{n\times d_v}$.

The BiFormer[34] module enhances this by incorporating a bi-level routing mechanism, defined as:

$$\text{BiFormer}\left(Q,K,V\right)=\text{Routing}\left(\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V\right) \tag{4}$$

The routing function selectively amplifies relevant features from the attended output in this formulation. The dimensional transformations remain consistent with the traditional attention mechanism, maintaining $\text{Q},\text{K},\text{V}\in R^{n\times d_k}$ and producing an output of $R^{n\times d_k}$.

BiFormer is used here instead of other attentions due to its efficient bi-level routing attention mechanism, which balances global context and local details, making it highly effective for processing complex sequential data like SMILES strings. Its design also reduces computational complexity while maintaining high performance, which is crucial for resource-constrained environments, as it allows the model to perform well even with limited computational resources.

The forward-forward (FF) Algorithm uses a unique two-pass approach to train neural networks. The first pass involves "positive" data with correct labels, which is evaluated using a "goodness" function to determine the quality of each hidden layer. The goal is to push this goodness function above a certain threshold. The second pass uses "negative" data, which is designed to be similar but not identical to the positive data, and aims to minimize the goodness function. By using these two passes with opposing objectives, the algorithm can effectively learn and optimize the network. A key innovation of the FF Algorithm is that it applies gradient descent directly to the hidden layers, using only local information and reducing the need to store all activations. This approach enables efficient use of computational resources and provides a new perspective on neural network training. FF algorithm is a contrastive learning method that try to increase the contrast between positive and negative data. Overall, the FF algorithm's focus on increasing the contrast between positive and negative data is a key aspect of its contrastive learning approach, which enables the model to learn effective representations and improve its performance on various tasks.

In the proposed method, a new loss function is developed for the forward-forward algorithm, which provides a more robust and efficient way to optimize neural networks and improve their performance. In this case, we have two parallel branches: one for positive data and the other one is for negative data. Given the frozen feature encoding subnetwork N_F^* and the attention subnetwork shown by N_T (include two distinct bifomer networks, normalization layers and global max-pooling layers), the output of the second stage can be formulated for the ith sample from the kth task as follows:

$$O_T^{(k)} = \sigma\left(GP\left(Norm\left(N_T^P\left(N_F^*\left(x_i^{(k)}\right)\right)\right)\right) - GP\left(Norm\left(N_T^N\left(N_F^*\left(x_i^{(k)}\right)\right)\right)\right)\right) \tag{5}$$

where $Norm\,(.)$ is the normalization layer, $GP\,(.)$ is the global pooling layer, and $\sigma\,(.)$ is the sigmoid function. As is shown in Fig. 1B, the output of the second stage is the difference of the positive biformer branch with the negative branch. Next, it is fed thought the sigmoid activation layer. The goal is to maximize this difference for the input sample belong to the positive class and minimize it for the input sample belong to the negative class. Hence, we have utilized the binary cross-entropy to satisfy this condition:

$$\mathcal{L}\left(y_i^{(k)}, \hat{y}_i^{(k)}\right) = -\sum_{c=1}^{C}\left[y_{i,c}^{(k)} \log\left(O_{T,i,c}^{(k)}\right)\left(1 - y_{i,c}^{(k)}\right) \log\left(1 - O_{T,i,c}^{(k)}\right)\right] \tag{6}$$

where $y_i^{(k)}$ is the ground truth label for the *i*th sample, and C is the number of classes. The total loss is averaged over all samples.

Stage three: prediction subnetwork tuning

In this stage, as shown in Fig. 2, unlike the previous one where only the feature encoding subnetwork N_F was frozen, both the feature encoding subnetwork N_F^* and the attention subnetwork N_T^* are frozen. This stage aims to fine-tune the task prediction subnetwork N_P using the backpropagation method, optimizing the model for the final task without altering the learned representations from the frozen sub-networks.

Given the frozen feature encoding subnetwork N_F^*, the frozen attention subnetwork N_T^*, and the task prediction subnetwork N_P, the loss function for the *i*th sample from the *k*th task is defined as:

$$\mathcal{L}\left(y_i^{(k)}, \hat{y}_i^{(k)}\right) = -\sum_{c=1}^{C} y_{i,c}^{(k)} \log\left(\hat{y}_{i,c}^{(k)}\right) \tag{7}$$

where C is the number of classes, $y_{i,c}^{(k)}$ is the true label for class c, $\hat{y}_i^{(k)} = N_P(N_T^*(N_F^*(x_i^{(k)})))$ is the predicted probability for class c.

The total loss is averaged across all samples and tasks, and backpropagation is applied only to the task prediction subnetwork N_P while keeping the feature encoding and attention sub-networks frozen. This allows the model to optimize task-specific performance while preserving the representatio learned in earlier stages.

The key difference in this stage compared to the second stage is that both the feature encoding subnetwork N_F and the attention subnetwork N_T remain frozen, ensuring that the backpropagation only refines the task-specific predictions made by N_P. This approach prevents overfitting and maintains the integrity of the learned feature and attention representations while focusing on improving classification accuracy.

Experiments

Dataset

To comprehensively evaluate our models, this study used 11 datasets spanning diverse categories, including Physiology, Biophysics, Physical Chemistry, and Quantum Mechanics. These datasets encompass various biological activities and chemical properties, providing a robust foundation for assessing model performance across different scientific domains. Table 1 offers comprehensive details about the datasets, including the number of tasks and compounds involved.

Tox21 Tox21 is a collection of 12 biological assays designed to assess the human toxicity of substances, targeting pathways like nuclear receptors and stress responses. It originated from the Tox21 Data Challenge to improve toxicological profiling and predictive modeling[35].

Category	Dataset	Tasks	Unique compounds
Physiology	BBBP	1	2053
	Tox21	12	7831
	SIDER	27	1427
	ClinTox	2	1478
Biophysics	BACE	1	1513
	HIV	1	41,127
	MUV	17	93,127
Physical chemistry	FreeSolv	1	643
	Lipo	1	4200
	ESOL	1	1128
Quantum mechanics	QM7	1	7165

Table 1. Dataset details: tasks and number of unique compounds.

BBBP BBBP contains data on compounds' ability to cross the blood–brain barrier. It helps develop models for predicting brain permeability, which is crucial for CNS drug development[36].

SIDER SIDER compiles data on adverse drug reactions for marketed medicines, grouped into 27 organ classes. It aids in predicting side effects, supporting pharmacovigilance and drug safety research[37].

ClinTox ClinTox includes data on FDA-approved drugs and those that failed clinical trials due to toxicity. It helps predict clinical toxicity outcomes, supporting safe drug development[38].

BACE BACE focuses on inhibitors of the beta-secretase 1 (BACE-1) enzyme, a target for Alzheimer's treatment. It helps develop predictive models for BACE-1 activity in neurodegenerative disease research[39].

HIV The HIV dataset contains data on compounds tested for HIV replication inhibition. It aids in developing predictive models for identifying potential HIV inhibitors for antiviral drug discovery[40].

MUV MUV offers curated assay data for benchmarking virtual screening methods. It ensures unbiased validation of compound activity predictions, supporting computational drug discovery tools[41].

FreeSolv FreeSolv provides hydration-free energy data for small organic molecules. It helps develop models predicting solubility, supporting drug development with better pharmacokinetic properties[42].

ESOL ESOL offers experimental solubility data for small organic molecules. It helps predict aqueous solubility, assisting early drug discovery and compound selection[43].

Lipo Lipo contains data on the lipophilicity of small organic compounds, measured by the octanol–water partition coefficient. It predicts drug-like properties, which are important for drug development evaluation[44].

QM7 QM7 provides quantum mechanical properties of small organic molecules. It aids in predicting molecular properties like energy and structure, supporting computational chemistry and material science research[45].

Performance measures

Several approved classification metrics were employed in this study to assess the predictive model's performance, including area under the receiver operating characteristic curve (AUC-ROC), accuracy, precision, recall, and the F1 score. These metrics provide comprehensive insights into model's effectiveness and reliability, highlighting not only its strengths but also potential areas for improvement.

- *Accuracy (ACC)* represents the proportion of instances correctly classified by the model out of the total number of cases evaluated. It is calculated as follows:

$$\text{Accuracy} \ \ (\text{ACC}) = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

In this formula, TP, TN, FP, and FN denote the counts of true positives, true negatives, false positives, and false negatives, respectively.

- *Precision* is the ratio of accurately predicted positive instances to the total number of instances predicted as positive. It reflects the model's accuracy in identifying the positive class:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

- *Recall (or sensitivity)* also known as sensitivity, evaluates the model's ability to identify all relevant positive instances within the dataset. It is calculated as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

- *F1 score* is the harmonic mean of precision and recall, which provides a balanced assessment of both metrics. It is particularly useful in scenarios with imbalanced datasets:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

- Lastly, the *area under the ROC curve (AUC-ROC)* evaluates the model's ability to differentiate between the classes. The closer the value is to 1, the more effective the classifier proves to be.

Together, these metrics provide a solid basis for evaluating a classifier's performance[46,47], emphasizing its accuracy and capability to perform effectively on new, unseen data.

Method comparison

To evaluate the effectiveness of Deep-CBN in predicting molecular properties, we conducted experiments on classification datasets such as Tox21, BBBP, SIDER, ClinTox, BACE, HIV, and MUV, as outlined in Table 2, as well as regression datasets like Lipo, ESOL, FreeSolv, and qm7, as outlined in Table 3. Additionally, Table 4 provides a comprehensive overview of the overall results of our model. These datasets cover various tasks, with

Model	ClinTox	BACE	BBBP	SIDER	Tox21	HIV	MUV	Avg
N-GramXGB[16]	0.875	0.791	0.691	0.655	0.758	0.787	–	0.759
Pre-trainGNN[18]	0.726	0.845	0.687	0.627	0.781	0.799	0.813	0.744
Abbasi et al.[20]	–	0.690	–	0.630	0.740	0.660	–	–
GROVER$_{base}$[19]	0.812	0.826	0.700	0.648	0.743	–	–	–
GROVER$_{large}$	0.762	0.810	0.695	0.654	0.735	–	–	–
ChemBERTa[28]	0.733	–	0.643	–	0.728	0.622	–	–
TrimNet[25]	0.906	0.843	0.892	0.606	0.812	0.804	^{c}0.851	0.810
MolCLR[21]	0.912	0.824	0.722	0.589	0.750	0.781	^{b}0.886	0.763
Molformer[22]	0.937	0.884	0.926	–	–	–	–	–
GraphCL[48]	0.760	0.754	0.700	0.605	0.739	0.785		0.724
FunQG[26]	0.841	0.862	0.914	0.642	0.845	–	–	–
ChemRL-GEM[49]	0.901	0.856	0.724	0.672	0.781	0.806	–	0.790
HiGNN[27]	0.930	^{c}0.890	^{c}0.932	0.651	^{c}0.856	^{b}0.816	0.186	^{b}0.838
GraphMVP[50]	0.775	0.812	0.724	0.639	0.759	0.770		0.746
Uni-Mol[24]	0.919	0.857	0.729	0.659	0.796	0.808	0.821	0.795
MolXPT[51]	0.953	0.884	0.800	0.717	0.771	0.781	–	0.818
BioAct-Het[30]	–	–	–	^{a}0.911	^{b}0.898	–	0.694	–
BatmanNet[31]	0.926	^{a}0.928	^{b}0.946	0.676	0.855	0.812	0.784	0.817
ChemBFN[52]	^{b}0.991	0.735	^{a}0.957	–	–	0.793	–	–
DGCL[53]	^{c}0.971	^{b}0.914	0.737	^{c}0.781	0.770	^{c}0.810	–	0.753
Deep-CBN	^{a}0.992	0.836	0.758	^{b}0.782	^{a}0.924	^{a}0.973	^{a}0.998	^{a}0.894

Table 2. Our full model ranked a, b, and c on ROC-AUC (%) performance against comparable approaches on molecular prediction benchmarks across datasets, outperforming comparable approaches.

Model	Lipo	ESOL	FreeSolv	Avg	QM7
N-GramRF[16]	0.812	1.074	2.688	1.525	81.9
Pre-trainGNN[18]	0.739	1.100	2.764	1.534	113.2
GROVER$_{base}$[19]	^{c}0.563	0.888	1.592	1.014	72.5
GROVER$_{large}$	^{b}0.560	0.831	1.544	^{c}0.978	72.6
TrimNet[25]	0.702	1.282	2.529	1.504	–
MolCLR[21]	0.691	1.271	2.594	1.519	^{c}66.8
FunQG[26]	0.622	0.818	1.501	0.980	–
ChemRL-GEM[49]	0.66	^{c}0.798	1.877	1.112	–
GraphMVP	0.681	1.029	–	–	–
Uni-Mol[24]	0.603	^{a}0.788	^{c}1.480	^{b}0.957	^{b}58.9
BatmanNet[31]	0.729	^{b}0.792	1.802	1.107	–
ChemBFN[52]	0.746	0.884	^{b}1.418	1.016	–
DGCL[53]	^{a}0.477	1.046	2.080	1.207	100.9
Deep-CBN	0.801	0.858	^{a}1.047	^{a}0.902	^{a}57.9

Table 3. Our regression model ranked a, b, and c on RMSE for Lipo, ESOL, and FreeSolv and on MAE for QM7, outperforming comparable approaches.

further details in Datasets. Consistent with prior studies, we employed scaffold splitting to divide each dataset into training, validation, and test sets, maintaining an 8:1:1 ratio for each task.

Deep-CBN demonstrates strong and competitive performance across a range of molecular property prediction benchmarks. On the ClinTox dataset, Deep-CBN achieves a near-optimal ROC-AUC of 0.992, marginally surpassing ChemBFN's score of 0.991. Similarly, on the Tox21, HIV, and MUV datasets, Deep-CBN records ROC-AUC values of 0.924, 0.973, and 0.998, respectively, positioning it among the top-performing models in these domains. Conversely, on the BACE, BBBP, and SIDER datasets, Deep-CBN obtains ROC-AUC scores of 0.836, 0.758, and 0.782, respectively. For instance, while BatmanNet achieves a ROC-AUC of 0.928 on BACE and ChemBFN attains 0.957 on BBBP, Deep-CBN's performance on these datasets is comparatively lower. Despite these discrepancies, the overall average ROC-AUC for Deep-CBN is 0.894, which underscores its competitive performance relative to other leading models. These findings suggest that while Deep-CBN exhibits robust and highly effective predictive capabilities—particularly on datasets such as ClinTox, Tox21, HIV, and MUV—there remain opportunities for further improvement in addressing challenges presented by datasets

Datasets	ACC	Recall	Precision	AUC-ROC	F1
Tox21	0.878	0.878	0.878	0.924	0.876
SIDER	0.745	0.745	0.745	0.782	0.745
MUV	0.998	0.998	0.998	0.998	0.998
ClinTox	0.978	0.976	0.976	0.992	0.976
BBBP	0.708	0.711	0.711	0.758	0.711
HIV	0.924	0.924	0.954	0.973	0.944
BACE	0.783	0.782	0.782	0.836	0.782

Table 4. The results of the deep-CBN model obtained on various datasets.

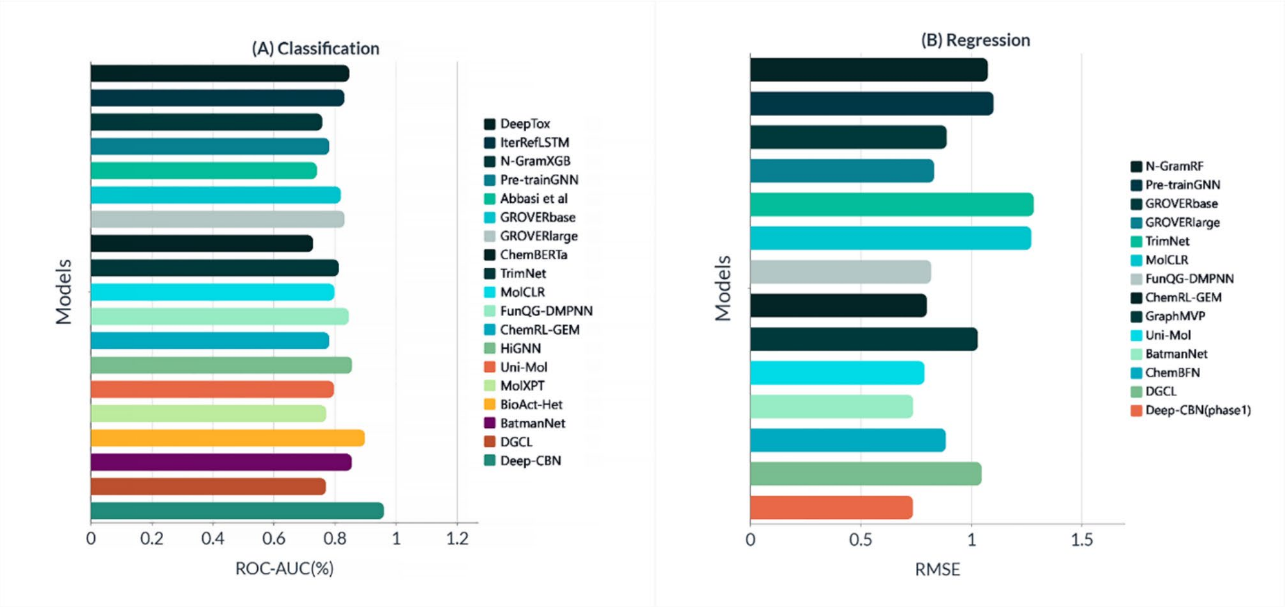


Fig. 3. Performance comparison of Deep-CBN and existing molecular prediction models on ROC-AUC and regression benchmarks across multiple datasets.

like BACE, BBBP, and SIDER. Overall, the results confirm the potential of Deep-CBN as a leading method in molecular property prediction.

Specifically, Deep-CBN achieves an RMSE of 0.801 on the Lipo dataset and 0.858 on the ESOL dataset. Although these results are not the best on these individual tasks—with DGCL recording the lowest RMSE on Lipo (0.477) and Uni-Mol attaining the top score on ESOL (0.788)—Deep-CBN stands out in other areas. Notably, on the FreeSolv dataset, Deep-CBN obtains the lowest RMSE of 1.047 (marked with rank “a”), and it also secures the best overall performance with an average RMSE of 0.902 (again ranked “a”). Furthermore, in the QM7 dataset, Deep-CBN achieves the lowest MAE at 57.9 (rank “a”), outperforming DGCL’s MAE of 100.9 and Uni-Mol’s MAE of 58.9. These results, as illustrated in Fig. 3 with the accompanying bar chart, highlight that while Deep-CBN may not always be the top performer on every individual dataset (as seen in Lipo and ESOL), its overall accuracy—especially on FreeSolv and QM7—demonstrates its strong capability in accurately predicting molecular properties.

Ablation study

In this section, we conduct a series of experiments on the Deep-CBN model to evaluate the impact and contribution of each component on the overall performance. The ablation study consists of three configurations. First, Stage Two is removed while keeping Stages One and Three intact. Second, Stage Three is removed, and Stages One and Two are maintained. Finally, BiFormer is replaced with a standard attention mechanism. The performance evaluation of all configurations is based on multiple independent runs to ensure reliability. The results of these experiments can be found in Table 5, where we analyze the effects of each modification on model performance.

Visualization of attention weights

In the ClinTox dataset, one important task is to predict whether a compound is FDA_APPROVED (positive) or not (negative). Figure 4 shows attention-weight heatmaps from this task for two positive (left) and two negative (right) samples. The vertical axis represents the token positions in the molecular sequence, and the horizontal

Model	ClinTox	BACE	BBBP	SIDER	Tox21	MUV	HIV
Deep-CBN-S1-S3	0.969	0.615	0.745	0.612	0.859	0.933	0.972
Deep-CBN-S1-S2	0.978	0.742	0.756	0.651	0.892	0.996	0.871
Deep-CBN-StdAttn	0.989	0.810	0.764	0.724	0.913	0.996	0.969
Deep-CBN	0.992	0.836	0.756	0.782	0.924	0.998	0.973

Table 5. Ablation study results on the effects of different phases and attention mechanisms on the performance of Deep-CBN.

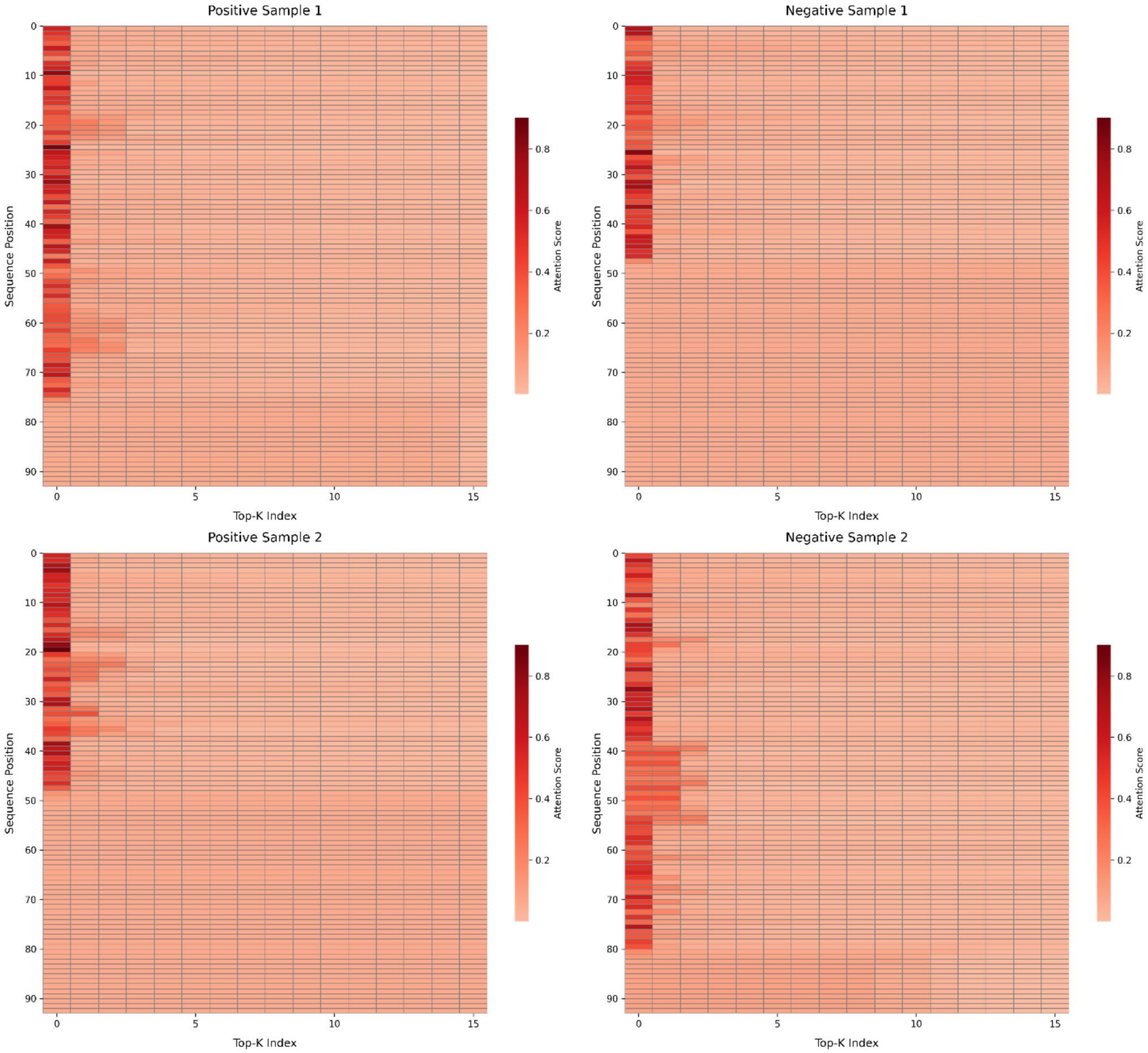


Fig. 4. Attention-weight heatmaps for two FDA_APPROVED (positive) and two non-FDA_APPROVED (negative) samples from the ClinTox dataset, where the vertical axis represents token positions and the horizontal axis (Top-*K* Index) shows the most-attended tokens. Darker colors indicate stronger attention levels.

axis (Top-K Index) corresponds to the most-attended tokens. Color intensity (red for positive, blue for negative) indicates the magnitude of the attention scores.

Positive samples (indices 0 and 304)
Both positive examples exhibit strong attention in the first 10–20 token positions, suggesting that certain early substructures may be influential for an FDA-approved label. However, moderate attention values also appear

Hyperparameter	Value	Description
Batch size	256	Number of samples per gradient update
Epochs	100,100,100	Full passes through the training data
Learning rate	0.0001	Step size for updating model weights
Dropout rate	0.1	Prevents overfitting by dropping nodes randomly
Number of heads	8	Number of attention heads in transformer blocks
Dim head	48	Dimension of each attention head in transformer block
Depth	2	Number of layers in transformer block
MLP dimension	128*3	Dimension of the multi-layer perceptron in the transformer block
Filters (Conv1D)	64, 64, 128	Number of filters in convolution layers applied to SMILES strings
Kernel size (Conv1D)	2, 4, 4	Size of the convolutional kernel for extracting features
TopK	16	Parameter for bilevel routing attention, deciding how many keys to route attention to

Table 6. Hyperparameters used in the Deep-CBN model for molecular prediction tasks.

in later positions, indicating that the model might be integrating additional contextual cues beyond the initial tokens.

Negative samples (indices 1031 and 783)
Likewise, the negative examples feature higher attention near the beginning of the sequence, pointing to the possibility that particular fragments—or their absence—at these early positions factor into a non-FDA-approved classification. In contrast to the positive samples, attention seems more evenly spread across a range of tokens, implying that the model may rely on multiple smaller cues rather than a single dominant substructure. These visualizations provide an interpretable view of the model's decision process, helping us identify which tokens—often functional groups or chemical motifs—could be most influential in each prediction. By examining where and how the model allocates attention, we gain insights into the underlying chemical features that distinguish positive from negative compounds, ultimately aiding model transparency and potential refinements.

Hyperparameters

The hyperparameters used for our model are summarized in Table 6. Preliminary experiments informed the choice of these hyperparameters to optimize the model's performance on each task. Additional details regarding the datasets utilized in this study are provided in Table 1.

In this paper, we executed the program using the Google Colab platform. This platform provides access to an NVIDIA T4 Tensor Core GPU equipped with 16GB of GDDR6 memory and 25GB of system RAM, significantly boosting our computational performance and allowing for more efficient processing.

Discussion

This research unveils Deep-CBN, a novel deep learning architecture that integrates convolutional neural networks (CNNs) with a Biformer attention mechanism and is trained through both the forward-forward algorithm and backpropagation. Its purpose is to boost molecular property prediction by thoroughly identifying and utilizing the complex patterns encoded in SMILES representations of chemical compounds. Deep-CBN performed exceptionally well on various benchmark datasets—Tox21, BBBP, SIDER, ClinTox, BACE, HIV, and MUV—achieving near-perfect ROC-AUC scores. Notably, it outperformed existing state-of-the-art methods by noteworthy margins. For instance, in the ClinTox dataset, Deep-CBN attained an ROC-AUC of 0.992, exceeding earlier approaches that reached up to 0.945. This remarkable success stems from combining CNNs for extracting local features with the Biformer attention mechanism for capturing global contextual information. By employing both the forward-forward algorithm and backpropagation in a hybrid training setup, the model harnesses their combined strengths to refine the learning process. This dual approach empowers Deep-CBN to generalize effectively over different datasets, pointing to its resilience and wide-ranging suitability in molecular property prediction. Despite these encouraging findings, there are a few constraints. For instance, the training procedure demands considerable computational resources, potentially hindering its feasibility for some scientific groups. Furthermore, although the model excelled on the chosen datasets, testing it on broader chemical domains is crucial for confirming its broader utility. Future endeavors will involve expanding the method to consider three-dimensional molecular structures. Parallel activities will aim to refine the model for lower computational requirements and investigate its practical benefits in real-world drug discovery contexts. In summary, Deep-CBN is an important leap in molecular property prediction, delivering a resilient and effective resource capable of speeding up computational chemistry and drug discovery activities.

Data availability

The source code for Deep-CBN, models, and data used in this study is available on our GitHub repository: https://github.com/akianfar/Deep-CBN.

Received: 5 January 2025; Accepted: 26 February 2025
Published online: 28 February 2025

References

1. Jiménez-Luna, J., Grisoni, F. & Schneider, G. Drug discovery with explainable artificial intelligence. *Nat. Mach. Intell.* **2**(10), 573–584 (2020).
2. Cherkasov, A. et al. QSAR modeling: Where have you been? Where are you going to?. *J. Med. Chem.* **57**(12), 4977–5010 (2014).
3. Hansch, C. & Fujita, T. p-σ-π analysis. A method for the correlation of biological activity and chemical structure. *J. Am. Chem. Soc.* **86**(8), 1616–1626 (1964).
4. Kubinyi, H. QSAR and 3D QSAR in drug design Part 1: Methodology. *Drug Discov. Today* **2**(11), 457–467 (1997).
5. Zheng, S. et al. Predicting drug–protein interaction using quasi-visual question answering system. *Nat. Mach. Intell.* **2**(2), 134–140 (2020).
6. Gawehn, E., Hiss, J. A. & Schneider, G. Deep learning in drug discovery. *Mol. Inform.* **35**(1), 3–14 (2016).
7. Cortes, C. & Vapnik, V. Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).
8. Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016).
9. Cover, T. & Hart, P. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **13**(1), 21–27 (1967).
10. Kleinbaum, D. G. et al. *Logistic Regression* (Springer, 2002).
11. Chen, T. & Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining* (2016).
12. Deng, J. et al. A systematic study of key elements underlying molecular property prediction. *Nat. Commun.* **14**(1), 6395 (2023).
13. Mayr, A. et al. DeepTox: Toxicity prediction using deep learning. *Front. Environ. Sci.* **3**, 80 (2016).
14. Wu, Z. et al. MoleculeNet: A Benchmark for molecular machine learning. *Chem. Sci.* **9**(2), 513–530 (2018).
15. Duvenaud, D. K. et al. Convolutional networks on graphs for learning molecular fingerprints. *Adv. Neural Inf. Process. Syst.* **28**, 2224–2232 (2015).
16. Liu, S., Demirel, M. F. & Liang, Y. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *Adv. Neural Inf. Process. Syst.* **32**, 8466–8472 (2019).
17. Yang, K. et al. Analyzing learned molecular representations for property prediction. *J. Chem. Inf. Model.* **59**(8), 3370–3388 (2019).
18. Hu, W. et al., *Strategies for Pre-training Graph Neural Networks.* arXiv preprint https://arxiv.org/abs/1905.12265 (2019).
19. Rong, Y. et al. Self-supervised graph transformer on large-scale molecular data. *Adv. Neural Inf. Process. Syst.* **33**, 12559–12571 (2020).
20. Abbasi, K. et al. Deep transferable compound representation across domains and tasks for low data drug discovery. *J. Chem. Inf. Model.* **59**(11), 4528–4539 (2019).
21. Wang, Y. et al. Molecular contrastive learning of representations via graph neural networks. *Nat. Mach. Intell.* **4**(3), 279–287 (2022).
22. Ross, J. et al. Large-scale chemical language representations capture molecular structure and properties. *Nat. Mach. Intell.* **4**(12), 1256–1264 (2022).
23. Li, H., Zhao, D. & Zeng, J. KPGT: Knowledge-guided pre-training of graph transformer for molecular property prediction. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2022).
24. Zhou, G. et al. *Uni-mol: A Universal 3D Molecular Representation Learning Framework* (2023).
25. Li, P. et al. TrimNet: Learning molecular representation from triplet messages for biomedicine. *Brief. Bioinform.* **22**(4), bbaa266 (2021).
26. Hajiabolhassan, H. et al. FunQG: Molecular representation learning via quotient graphs. *J. Chem. Inf. Model.* **63**(11), 3275–3287 (2023).
27. Zhu, W. et al. HiGNN: A hierarchical informative graph neural network for molecular property prediction equipped with feature-wise attention. *J. Chem. Inf. Model.* **63**(1), 43–55 (2022).
28. Chithrananda, S., Grand, G. & Ramsundar, B. *ChemBERTa: Large-Scale Self-Supervised Pre-training for Molecular Property Prediction.* arXiv preprint https://arxiv.org/abs/2010.09885 (2020).
29. Ahmad, W. et al., *Chemberta-2: Towards Chemical Foundation Models.* arXiv preprint https://arxiv.org/abs/2209.01712 (2022).
30. Paykan Heyrati, M. et al. BioAct-Het: A heterogeneous siamese neural network for bioactivity prediction using novel bioactivity representation. *ACS Omega* **8**(47), 44757–44772 (2023).
31. Wang, Z. et al. BatmanNet: Bi-branch masked graph transformer autoencoder for molecular representation. *Brief. Bioinform.* **25**(1), bbad400 (2024).
32. Hinton, G., *The Forward–Forward Algorithm: Some Preliminary Investigations.* arXiv preprint https://arxiv.org/abs/2212.13345 (2022).
33. Vaswani, A. et al. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30**, 512–515 (2017).
34. Zhu, L. et al. Biformer: Vision transformer with bi-level routing attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023).
35. *Tox21 Challenge*. 2019 February 22, 2019]; Available from: https://tripod.nih.gov/tox21/challenge/.
36. Martins, I. F. et al. A Bayesian approach to in silico blood-brain barrier penetration modeling. *J. Chem. Inf. Model.* **52**(6), 1686–1697 (2012).
37. Kuhn, M. et al. The SIDER database of drugs and side effects. *Nucleic Acids Res.* **44**(D1), D1075–D1079 (2016).
38. Gayvert, K. M., Madhukar, N. S. & Elemento, O. A data-driven approach to predicting successes and failures of clinical trials. *Cell Chem. Biol.* **23**(10), 1294–1301 (2016).
39. Subramanian, G. et al. Computational modeling of β-secretase 1 (BACE-1) inhibitors using ligand based approaches. *J. Chem. Inf. Model.* **56**(10), 1936–1949 (2016).
40. *AIDS Antiviral Screen Data*. 2019 February 20, 2019]; Available from: http://wiki.nci.nih.gov/display/NCIDTPdata/AIDS+Antiviral+Screen+Data.
41. Rohrer, S. G. & Baumann, K. Maximum unbiased validation (MUV) data sets for virtual screening based on PubChem bioactivity data. *J. Chem. Inf. Model.* **49**(2), 169–184 (2009).
42. Mobley, D. L. & Guthrie, J. P. FreeSolv: A database of experimental and calculated hydration free energies, with input files. *J. Comput. Aided Mol. Des.* **28**, 711–720 (2014).
43. Delaney, J. S. ESOL: Estimating aqueous solubility directly from molecular structure. *J. Chem. Inf. Comput. Sci.* **44**(3), 1000–1005 (2004).
44. Gaulton, A. et al. ChEMBL: A large-scale bioactivity database for drug discovery. *Nucleic Acids Res.* **40**(D1), D1100–D1107 (2012).
45. Blum, L. C. & Reymond, J.-L. 970 million druglike small molecules for virtual screening in the chemical universe database GDB-13. *J. Am. Chem. Soc.* **131**(25), 8732–8733 (2009).
46. Bishop, C. M. & Nasrabadi, N. M. *Pattern Recognition and Machine Learning* Vol. 4 (Springer, 2006).
47. Powers, D. M. *Evaluation: from Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation.* arXiv preprint https://arxiv.org/abs/2010.16061 (2020).
48. You, Y. et al. Graph contrastive learning with augmentations. *Adv. Neural Inf. Process. Syst.* **33**, 5812–5823 (2020).
49. Fang, X. et al. Geometry-enhanced molecular representation learning for property prediction. *Nat. Mach. Intell.* **4**(2), 127–134 (2022).
50. Liu, S. et al. *Pre-training Molecular Graph Representation with 3d Geometry.* arXiv preprint https://arxiv.org/abs/2110.07728 (2021).
51. Liu, Z. et al. *Molxpt: Wrapping Molecules with Text for Generative Pre-training.* arXiv preprint https://arxiv.org/abs/2305.10688 (2023).

52. Tao, N. & Abe, M. *A Bayesian Flow Network Framework for Chemistry Tasks.* arXiv preprint https://arxiv.org/abs/2407.20294 (2024).
53. Jiang, X., Tan, L. & Zou, Q. DGCL: Dual-graph neural networks contrastive learning for molecular property prediction. *Brief. Bioinform.* **25**(6), bbae474 (2024).

Author contributions

AK: conceptualization, data curation, formal analysis, methodology, writing original draft, programming, visualization, review and editing. PR: result analysis, writing, visualization, conceptualization, supervision, project administration, review and editing. ZA: conceptualization, result analysis, writing, visualization.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to P.R.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.