BİTLİS EREN ÜNİVERSİTESİ
FEN BİLİMLERİ DERGİSİ

ISSN: 2147-3129 / e-ISSN: 2147-3188

| | | | |
|---|---|---|---|
| **Article Type** | : Research Article | **Year** | : 2025 |
| **Received** | : January 12, 2025 | **Volume** | : 14 |
| **Revised** | : April 1, 2025 | **Issue** | : 2 |
| **Accepted** | : April 9, 2025 | **Pages** | : 907-929 |
| **DOI** | : 10.17798/bitlisfen.1618273 | | |


# CNS-DDI: AN INTEGRATED GRAPH NEURAL NETWORK FRAMEWORK FOR PREDICTING CENTRAL NERVOUS SYSTEM RELATED DRUG-DRUG INTERACTIONS

**Muhammed Ali PALA** [1, 2, *]

[1] *Sakarya University of Applied Sciences, Department of Electrical and Electronics Engineering, Sakarya, Türkiye*

[2] *Sakarya University of Applied Sciences, Biomedical Technologies Application and Research Center (BIYOTAM), Sakarya, Türkiye, pala@subu.edu.tr*

## ABSTRACT

The central nervous system (CNS) is one of the most complex and vital systems of the human body and is particularly interrelated with all other systems. Treatment modalities targeting the CNS as well as those targeting other systems may directly or indirectly affect the CNS. Especially in cases of polypharmacy, drug-drug interactions (DDIs) can lead to severe problems. The widespread use of drugs that have an effect on the CNS and the unpredictability of possible interactions between these drugs both complicate the treatment processes of patients and considerably increase health costs. In this study, a novel method based on Graph Convolutional Neural Networks (GCN) is proposed to predict CNS-related DDIs. The proposed approach utilizes a data fusion method by exploiting both graph structures and physical properties of drug molecules. This integrated approach enabled a more comprehensive and reliable prediction of drug interactions. The developed model achieved 98.67% accuracy and 0.994 AUC in the training process and 98.40% accuracy and 0.991 AUC in the validation process. A Graphical Interface (GUI) was designed to make the developed model easily usable by users. The integration of molecular structure and interaction network data sets a new benchmark for reliability and accuracy in DDIs prediction, addressing a critical need in modern healthcare systems. The developed methods and tools have significant potential for predicting drug interactions in the drug discovery process and in polypharmacy situations.

**Keywords:** Artificial intelligence in drug, Deep learning in drug discovery, Central Nervous System (CNS), Drug-Drug Interactions (DDIs), Computational chemistry, drug discovery.

# 1 INTRODUCTION

The central nervous system is one of the most complex and vital systems of the human system, regulating a wide range of physiological and psychological processes such as cognitive functions, emotional regulation, motor skills, sensory perception, homeostasis and behavioural responses [1], [2]. This system, composed of the brain and spinal cord, coordinates these processes through complex interactions of neurotransmitters, neuropeptides and other bioactive molecules [3]. The disruption of these processes and changes in their effectiveness due to various reasons cause serious problems. To eliminate these problems, modern medicine tries to find solutions by using drugs. CNS drugs used in the treatment of neurological and psychiatric disorders aim to have a therapeutic effect by targeting neurochemical instabilities in this system [4], [5]. These drugs regulate neurotransmission by modulating target receptors or enzymes, thereby alleviating or eliminating disease symptoms. However, the complex and dynamic nature of the CNS makes the effects of these drugs difficult to predict and increases the likelihood of off-target effects, i.e. unwanted side effects [6]. Furthermore, CNS drugs' pharmacokinetic and pharmacodynamic properties may show inter-individual and intra-individual variability, which may affect drug efficacy and safety [7]. Especially under polypharmacy conditions, the simultaneous use of multiple CNS-effective drugs may trigger pharmacologic interactions. These interactions may lead to decreased drug efficacy or increased toxic effects of drugs [8]. Therefore, a comprehensive understanding of the mechanisms of action, pharmacological properties and possible interactions of CNS drugs is critical for developing rational treatment strategies, implementing individualized treatment approaches and ensuring patient safety. Therefore, it is increasingly essential to understand these complex drug interactions in the CNS better to improve clinical practice.

Drug-drug interactions represent a complex and vital aspect of modern pharmacotherapy and have an increasing prevalence in today's clinical practice, especially in the management of CNS-affecting drugs and the prevalence of polypharmacy due to comorbidities. DDIs occur when two or more drugs administered simultaneously modify the pharmacokinetic and/or pharmacodynamic properties of two or more drugs [9]. Pharmacokinetic interactions can affect the drug's absorption, distribution, metabolism and excretion processes, altering the plasma concentration and, thus, the bioavailability of the drug [10]. Central nervous system drugs are susceptible to such interactions and can have serious clinical consequences as they target complex networks of neurotransmission and neurotransmitter systems [11]. DDIs may lead to a decrease in the expected therapeutic efficacy

of drugs, the occurrence of unintended adverse effects, variability in treatment response, prolonged hospitalisation and even increased treatment costs [12]. In clinical practice, prediction of DDIs and assessment of risk factors are vital for patient safety and optimal treatment outcomes. In this context, a deep understanding of the pathophysiological mechanisms of DDIs and the development of advanced methods that can predict these interactions are essential to support clinical decision-making.

Traditional experimental methods are time-consuming, costly and inadequate for DDIs determination, especially for large-scale monitoring. In recent years, these limitations have increased the need for highly efficient and fast DDIs prediction methods. Deep learning algorithms have emerged as a promising alternative in this field [13], [14], [15]. Deep learning approaches are a machine learning subfield based on multilayer artificial neural networks that can automatically learn and generalise patterns from complex data sets. By integrating a wide range of biological and pharmacological data, such as chemical structures of drugs, pharmacological properties, gene expression profiles, protein interaction networks and clinical data, deep learning algorithms can excel in predicting DDIs [16], [17]. In particular, deep learning models, such as Graph Neural Networks (GNNs), can model the molecular structure, sequential properties and relational structure of drugs to provide high accuracy and reliability in DDIs prediction [18], [19]. Deep learning-based approaches have the potential to accelerate the drug development process, assess the safety of potential drug combinations and contribute to the development of personalised treatment approaches by predicting DDIs that are difficult to determine experimentally. Moreover, deep learning-based DDIs prediction methods play an essential role in the design of clinical trials, drug safety monitoring and pharmacovigilance applications. Therefore, further exploration of the potential of deep learning in the field of DDIs prediction and integration of this technology into clinical applications is of great importance for patient safety and treatment efficacy.

Machine learning and deep learning-based approaches for predicting DDIs can utilise various feature vectors and data formats to represent drugs. Chemical structures, pharmacological classifications, biological activities, gene expression profiles, protein interaction networks and clinical information of drugs are frequently used data types to model and predict DDIs [20], [21]. Properly processing of this data and its input to deep learning models significantly impacts DDIs prediction performance. In particular, the Simplified Molecular Input Line System (SMILES) representation of the molecular structure of drugs is one of the best and most reliable data forms for deep learning algorithms. SMILES strings allow

a drug molecule's atomic structure, chemical bonds, branch points and cycles to be encoded as a one-dimensional text string [22]. In recent years, combining deep learning-based methods and SMILES representations has led to significant advances in DDIs prediction. On the other hand, although similarity matrices are used in DDIs prediction, these methods involve extra computational steps before application [23]. This brings with it several limitations for similarity-based DDIs estimation methods.

Similarity-based methods are based on the assumption that similar drugs will have similar interactions based on the similarities between the feature vectors of the drugs. The advantage of these methods is that they are straightforward and intuitive [24]. However, the choice of similarity measures in DDIs prediction can significantly affect the estimation performance. This leads to user-dependent analyses. On the other hand, Matrix Factorization Methods try to solve the missing data problem by decomposing the DDIs matrix into low-dimensional latent matrices [25]. The advantage of these approaches is that they can efficiently handle large and sparse DDIs matrices. However, these methods may have limitations in elucidating the underlying mechanisms of drug-drug interactions (DDIs), which hinders their ability to enhance performance when integrated with deep learning models. In contrast, graph-based approaches model drug interaction networks by representing drugs and their interactions as nodes and edges, offering a more structured and interpretable framework for understanding DDIs. GNNs give better results than other methods due to their ability to capture complex relational patterns in drug molecular structures or interaction networks.

Many studies are using GNN methods for DDI prediction. The HetDDI model uses a Heterogeneous Graph Neural Network (HGNN) to predict DDI by integrating the molecular structures of drugs with external biomedical information [26]. This model improves its ability to effectively predict unobserved interactions by collecting information from various sources. In contrast, the generalization capabilities are improved through pre-training methods that transform drug SMILES into molecular graphs and initialize node embeddings. Another approach, the Knowledge Graph Neural Network (KGNN) framework, addresses the limitations of existing models by capturing high-order structures and semantic relationships in knowledge graphs [27]. By expanding the receptive field to include information multiple steps away from each entity, KGNN more effectively models long-distance correlations between drugs and their potential interactions and outperforms traditional models. On the other hand, the EmerGNN model is a GNN-based method that focuses on predicting interactions for emerging drugs for which comprehensive DDIs data is often unavailable [28]. By extracting

pathways between drug pairs and incorporating relevant biomedical concepts, EmerGNN delivers DDI predictions with higher accuracy than existing methods, making it particularly useful for novel therapeutic agents. The AutoDDI model is a method that automates the design of GNN architectures specifically for DDIs prediction [29]. This approach uses reinforcement learning to optimise the architecture based on various datasets. It helps to achieve high performance on real-world datasets while significantly reducing the time and expertise required for manual design. Another notable approach in the literature is the HGNN-DDI model [30]. This model utilised attentional mechanisms within GNNs to improve DDIs prediction accuracy. It further enhanced its prediction capabilities by integrating attention layers to focus on essential features in drug interaction data. In conclusion, GNNs show promising results in DDIs prediction.

This study proposes a novel GNN-based deep learning method to predict interactions between drugs that are effective in the CNS. Aiming to overcome the limitations of traditional approaches, this deep learning model addresses key challenges such as the need for extensive data preprocessing, the complexity of handling multi-parametric models, and the reliance on similarity-based methods. Traditional approaches often struggle with these issues, limiting their predictive accuracy and generalization capabilities. In contrast, the proposed model integrates SMILES strings representing the molecular structures and physicochemical properties of drugs with graph-based neural networks, offering a more efficient and accurate alternative for drug interaction prediction. This integration allows for more accurate modelling of interactions by considering both the atomic-level structure of drugs and their relational structure in interaction networks. While SMILES represent and efficiently encode the chemical structure of drug molecules, GNNs offer the ability to learn the complex relationships and interactions between these structures. The main contributions and novelties of the proposed method to DDIs prediction are (1) its ability to model the molecular structure and interaction networks of drugs simultaneously, (2) the ability to provide higher performance using fewer data pre-processing parameters compared to other deep learning models, such as similarity matrix-based methods, and (3) its ability to perform a more comprehensive DDIs prediction by integrating both structural and relational information. In this context, the proposed method significantly improves existing DDIs prediction approaches and contributes to more accurate and reliable detection of interactions between CNS drugs. The findings of this study are expected to improve drug selection and safety in clinical applications and lead to new approaches in drug discovery and development processes.

## 2 PROPOSED METHOD

To predict CNS drug interactions, a novel deep learning model has been developed that model the molecular structure of drugs with a graph-based approach and integrates this structural information with associative information in interaction networks. The proposed method consists of finding drug pairs that affect CNS activity among drug interaction pairs, preprocessing these interaction pairs and classifying the interaction processes. In the first stage, interaction data between CNS drugs and SMILES strings and properties representing the chemical structures of these drugs were obtained from DrugBank, a comprehensive drug database. A series of complementary and holistic data preprocessing procedures were applied to the identified data. In the DrugBank dataset, drug pairs without SMILES strings or physicochemical properties were removed from the selected data after text mining. SMILES strings expressing the molecular structure of drugs were converted into graphical form. During this conversion process, the strings were standardised using tokenisation and padding techniques, thus making them suitable for the model's input format. A graph convolutional neural network-based method was developed to predict drug interactions. This model aims to learn the interaction patterns by considering the molecular structure of drugs as a graph structure, considering the atomic level properties of drugs and their neighbourhood relationships. This approach is designed better to understand molecular structures' complex patterns and interactions. The developed deep learning model was trained with the information obtained from the dataset, and the parameters of the model were adjusted to minimise the prediction error using appropriate optimisation algorithms. Finally, the model's performance was evaluated using various metrics such as accuracy, precision and area under the ROC curve. The proposed method provides a more comprehensive analysis by modelling the molecular structure of drugs and their relational structure in interaction networks simultaneously. It aims to make more reliable DDIs predictions compared to existing methods. In addition, a user graphical interface has been designed to enable experts in the field to use the developed method.

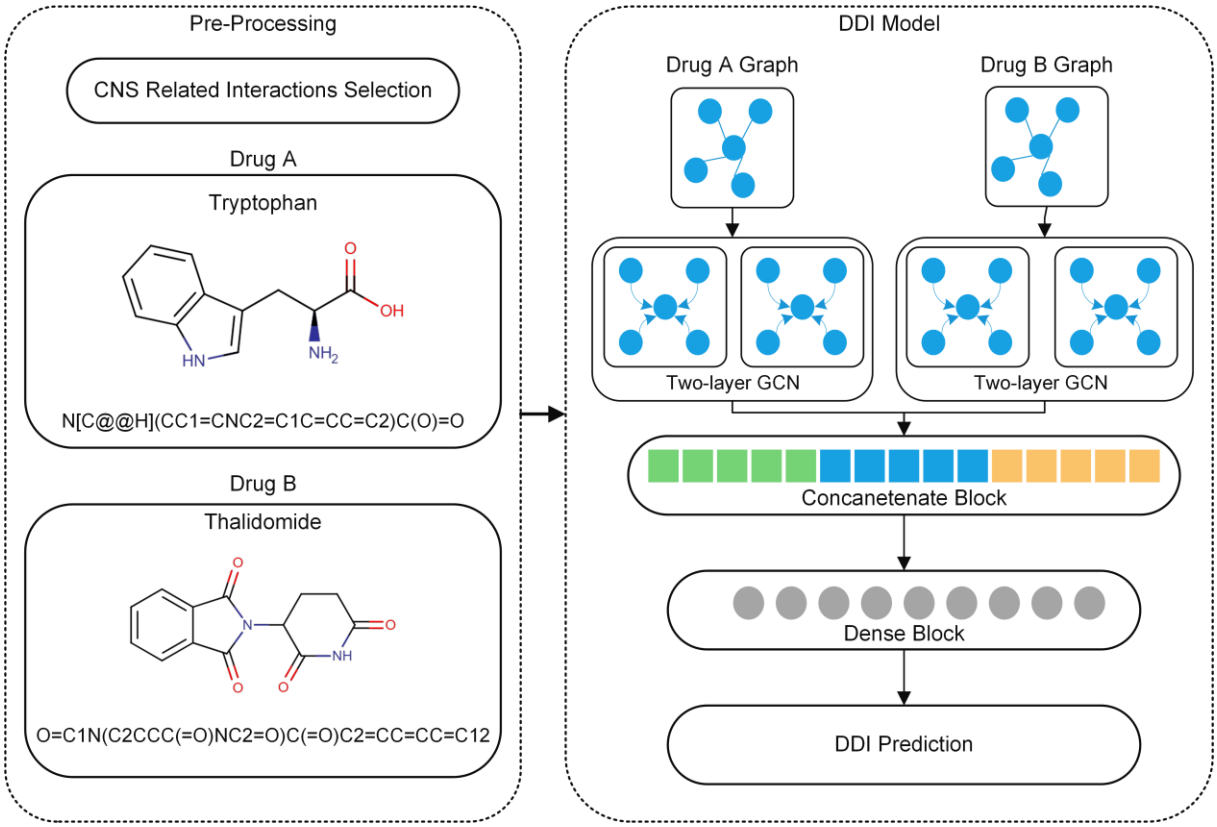



***Figure 1. Flowchart of the CNS-DDI method.***

# 3 MATERIALS AND METHODS

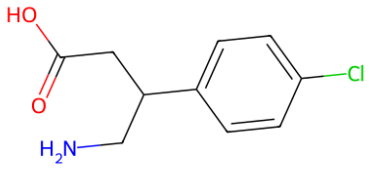
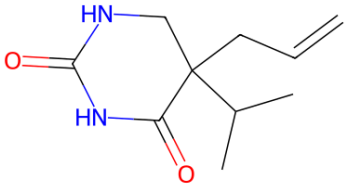
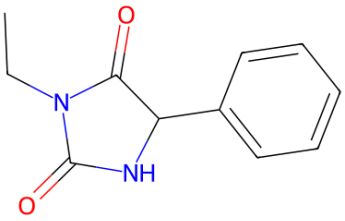
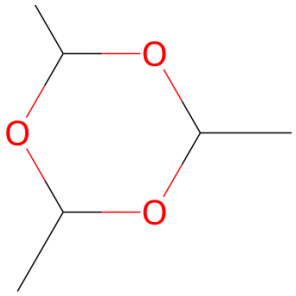
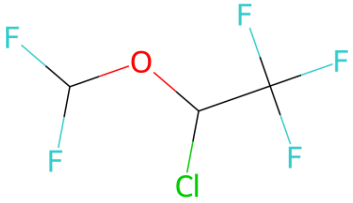
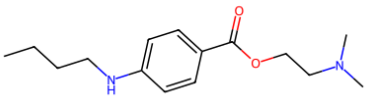
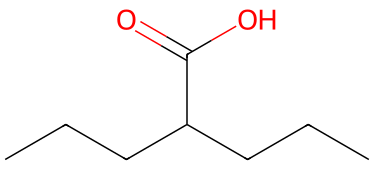
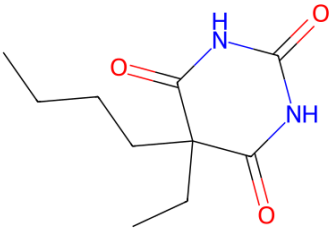
## 3.1 Dataset

This study applied a comprehensive data mining approach to model drug interactions and train the developed deep learning models. Accurate modelling of drug interactions is of critical importance in the field of pharmacology, especially for predicting and preventing adverse effects that may occur in polypharmacy situations. In this context, we used DrugBank version 5.1.13, which provides information about drugs and their biological targets and is widely used in bioinformatics and chemoinformatics [31]. DrugBank version 5.1.13 contains a total of 17,430 different drugs and drug interactions between these drugs. In this study, the dataset was filtered with a focus on CNS drugs and a subset of 156,179 drug interactions for 722 drugs was created.

A text-mining process was conducted on the DrugBank dataset to identify CNS-related drug interactions. In this process, two main interaction types were emphasised to identify CNS-related interactions. In the DrugBank dataset, the types of interactions between drugs are expressed by various textual relations. The first data type selected for the study is the increase

in CNS depressant activity when two drugs interact. In the DrugBank dataset, this interaction type is expressed as “DrugA may increase the central nervous system depressant (CNS depressant) activities of DrugB”. Drug A and Drug B represent the names of the interacting drugs. In the collection of the dataset, we selected drugs that secondarily increase the risk or severity of CNS depression by entering into an interaction. This second type of interaction is described in the DrugBank dataset as “The risk or severity of CNS depression can be increased when DrugA is combined with DrugB”. A series of text-mining methods were applied to the DrugBank dataset to identify these two types of drug interactions. For each interaction record, relevant drug names, SMILES strings and molecular weights, LogP values, molecular weights and Monoisotopic Mass values were selected for all drugs and interaction pairs. In the data cleaning phase, duplicate interaction records and rows with missing SMILES information were removed from the dataset. In addition, the information of other data rows with missing information was completed using the PubChem database [32]. Unvalidated or inconsistent SMILES strings were removed from the dataset, and automated checks ensured the consistency of the dataset. As a result, a two-class dataset consisting of drug interaction pairs that increase CNS depressant activity and increase CNS depression risk was created. As a result of this process, 142,864 drug interactions that can increase the risk or severity of CNS depression and 13,315 drug interaction pairs that can increase CNS depressant activity were obtained in the dataset. There are 156,179 drug interaction pairs in the data set created by applying data mining. Table 1 provides the various drug interaction pairs found in the dataset. For classification processes, this leads to a class imbalance problem. In order to overcome this problem, the weighted Synthetic Minority Over-sampling Technique (SMOTE) method was applied to the minority class of drug pairs that increase CNS depressant activity, as will be discussed in the following sections.

***Table 1. Examples of various drug interaction pairs in the dataset.***

| **Interaction Type: Drug A may increase the central nervous system depressant (CNS depressant) activities of Drug B** | |
|---|---|
| **Drug A** | **Drug B** |
|  <br>Baclofen<br>NCC(CC(O)=O)C1=CC=C(Cl)C=C1 |  <br>Aprobarbital<br>CC(C)C1(CC=C)C(=O)NC(=O)NC1 |
|  <br>Ethotoin<br>CCN1C(=O)NC(C1=O)C1=CC=CC=C1 |  <br>Paraldehyde<br>CC1OC(C)OC(C)O1 |
| **Interaction Type: The risk or severity of CNS depression can be increased when Drug A is combined with Drug B** | |
| **Drug A** | **Drug B** |
|  <br>Isoflurane<br>FC(F)OC(Cl)C(F)(F)F |  <br>Tetracaine<br>CCCCNC1=CC=C(C=C1)C(=O)OCCN(C)C |
|  <br>Valproic acid<br>CCCC(CCC)C(O)=O |  <br>Butobarbital<br>CCCCC1(CC)C(=O)NC(=O)NC1=O |

## 3.2 SMILES

SMILES, a widely used notation system, was used to represent the molecular structure of drugs. SMILES is a notation format expressing drug molecules' atomic structure and chemical bonds as a one-dimensional ASCII string. This system allows chemical structures to be quickly processed and analysed in a computer environment. SMILES strings represent atoms with symbols, bonds with signs and branching with parentheses and numbers. It uses numbers to close loops. SMILES notation has been widely used in various fields, such as chemical computing, drug discovery and molecular modelling. This study used SMILES strings for each drug in the drug interaction data obtained from the DrugBank dataset. Molecular representations of drugs were obtained through these strings, and drug molecular formats were used as input feature data for the developed deep-learning model. The one-dimensional structure of SMILES strings makes it easier for deep learning algorithms to learn and analyse molecular structures automatically. This way, drug chemical properties and interactions can be modelled more effectively. Using the SMILES representation in our study allowed the model to directly and effectively understand the molecular structures of drugs.

## 3.3 SMILES to Graph

A transformation step from SMILES representations to graph structures was performed to transform the molecular structures of the drugs as input to the deep learning model. This transformation process transformed the molecular structures into a format suitable for graph-based neural networks by treating the atoms of molecules as nodes and the chemical bonds between atoms as edges. First, each SMILES string is converted into a molecule object. This function analyses the SMILES string to create a data structure representing the atoms and bonds of the molecule. The open-source RDKit library was used for this transformation [33]. If the SMILES string is incorrect or not recognised by RDKit, the transformation process for this molecule is skipped. Then, node and edge features were extracted from the molecule object. For nodes, the atomic number of each atom was used. The atomic number of drug molecules is a fundamental parameter representing each atom's uniqueness and chemical properties. Therefore, the atomic numbers of the molecules were determined as the node matrices of the graphs.

The starting and ending atomic indices of each chemical bond and the bond type were recorded for the edges. Bond types refer to single, double, triple or aromatic bonds represented by RDKit with numerical values. These values allow the model to distinguish between different

bond types. This information was recorded as edge indices and bond properties. As a result of these steps, a node array containing atomic numbers, an edge array containing edge indices and a bond property array containing bond types were obtained for each drug molecule. In addition, during data preparation, the molecular weights, LogP values, molecular weights and Monoisotopic Mass of the drugs were included in this representation as an array. This array structure represents the graph structure for input to the deep learning model.

## 3.4 SMOTE

A data augmentation method based on SMOTE was applied to address the class imbalance in the drug interaction dataset and to enable the model to learn the minority class better. Since in drug interaction data, the number of drug pairs that can increase CNS depressant activity in the interacting CNS is significantly less than the number of drug pairs that can increase the risk or severity of CNS depression; such class imbalance can cause problems in the training process of the model [34]. Therefore, using the weighted SMOTE data augmentation strategy, the number of samples in the minority class was increased, allowing the model to balance learning from both classes during training and validation.

The SMOTE method creates new synthetic instances from existing instances in the minority class. The basic SMOTE algorithm finds k-nearest neighbours for each minority class instance. It makes a new synthetic instance by randomly selecting a point between these neighbours and the original instance. This process aims to reduce class imbalance by increasing the number of instances in the minority class. The weighted SMOTE technique used in this study aims to optimise the boosting process by assigning different weights to different data samples. These weights are determined according to the importance and difficulty of the available examples. During the model's training, higher weights are given to the more critical or more difficult instances so that the model focuses on these instances. A weight was first assigned to each instance in the training dataset for the weighted SMOTE implementation. These weights are set high for interacting instances and lower for non-interacting instances. When applying SMOTE for the minority class for interaction pairs, new synthetic examples were created by considering these weights. In the Weighted SMOTE method, the new data point to be generated is the convex combination of the two selected points in the feature space. Mathematically, the new synthetic sample $x_{\text{new}}$ is defined as in equation 1.

$$x_{\text{new}} = x_i + \lambda w_i (x_{zi} - x_i) \tag{1}$$

Here, $x_i$ denotes a random sample selected for a minority class, and $x_{zi}$ denotes a point selected from its k-nearest neighbors of the same class. $w_i$ is the weight coefficient assigned to instance $x_i$. $\lambda$ is a multiplier chosen in the range [0,1] and controls the degree of interpolation.

In our study, instead of directly modifying the structural representations of chemical molecules, the weighted SMOTE method interpolates in feature space to generate new synthetic samples. Thus, variations of synthetic data based on the properties of existing compounds were generated. Furthermore, during the data generation process, a control mechanism was added through the RDKit library to prevent the generation of completely random or invalid molecules from the new synthetic data. Random data points were generated to replace the data that failed this control mechanism. Thus, the process of increasing the minority of data was realized. As a result of weighted SMOTE process, 142,864 drug interactions that can increase the risk or severity of CNS depression and 142,864 drug interaction pairs that can increase CNS depressant activity were obtained in the dataset.

## 3.5 Graph Convolutional Network

Graph-based neural networks are deep learning models that have recently received significant attention for modelling correlated data structures, especially for analysing complex networks such as chemical compounds and biological interactions. [35]. GNNs, unlike traditional neural networks, consist of structures that can receive relational data as input through nodes and edges instead of one-dimensional strings. In this way, GNNs are model structures that perform well in classifying data in non-oclide geometries such as molecular structures, social networks and interaction networks. GCNs are neural networks with a more specialised structure of GNNs [36]. GCNs is a method that applies a convolution process on the graph data used as input, initially through randomly determined filters, and then produces results through a series of operations. The most potent aspect of GCNs compared to other graph-based methods is that they allow a sharper understanding of spatial features from the data in the graph structure.

In GCN methods, a graph is defined as $G = (V, E)$. In the graph definition, $V$ is defined as the set of nodes and $E \subseteq VxV$ is defined as the set of edges. To operate on the graph, node features are usually expressed by a feature matrix $X \in \mathbb{R}^{NxF}$. Where $N$ is the number of nodes and $F$ is the feature size of each node. The edge information is represented by the adjacency matrix $A \in R^{NxN}$. The graph convolution process propagates neighbourhood information over nodes by combining node features. This process is expressed as follows:

$$H^{(l+1)} = \sigma\left(\hat{A}H^{(l)}W^{(l)}\right) \tag{2}$$

Where $H^{(l)}$ is the node features in the *l-th* layer, $\hat{A}$ is the normalised adjacency matrix and $W^{(l)}$ is the learnable weights. The normalised adjacency matrix $\hat{A}$ balances the effect of node neighbourhoods, provides numerical stability, and is expressed as follows.

$$\hat{A} = \tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2} \tag{3}$$

Where $\tilde{A} = A + I$ is the self-connections of nodes are included by adding the unit matrix. $\tilde{D}$ denotes the degree matrix of $\tilde{A}$. Hence, for a graph-level classification, node features are pooled and then classified:

$$z = Pool(H^{(L)}), \quad \hat{y} = soft\,max(z) \tag{4}$$

In this update rule, the features of each node are updated with a weighted average of the features of its neighbouring nodes and a learnable weight matrix [37]. Degree normalisation ensures nodes with many neighbours have less influence on the update process. This way, the model learns to balance, and nodes with many neighbours are not over-dominated.

The GCN architecture used in this study consists of multiple GCN layers with non-linear activation functions applied between them. These layers aim to learn the structural and relational properties of atoms within drug molecules, represented as graph structures, to obtain more meaningful features for DDI prediction. After the GCN layers, the obtained node features are aggregated and passed through fully connected layers to predict the presence or absence of drug interactions. Using a GCN-based architecture allows us to effectively model the molecular structure of drugs and improve DDI prediction performance. The developed model is designed to process drug molecules as graph-structured data, where nodes represent atoms and edges represent chemical bonds. The model operates on two separate graph inputs, each corresponding to one of the interacting drug molecules. Each input graph consists of three components: node features, including atomic numbers of atoms; edge indices, which define the connectivity between atoms; and global molecular descriptors that encode additional molecular characteristics. The dimensions of these inputs are dynamically determined based on the largest molecule in terms of node count and the most complex molecule in terms of edge count within the dataset. This design enables the model to efficiently handle molecules of varying sizes.

The graph input of each drug is first processed by a custom graph convolution layer, which takes node features and edge indices as input and applies the message-passing mechanism to propagate information across the molecular graph. This enables the model to

capture atomic-level relationships and structural dependencies crucial for DDI prediction. Within the graph convolution layer, the node embeddings are projected into a 128-dimensional vector space, applying a nonlinear transformation with the ReLU activation function. This step allows the model to learn high-level molecular representations, where each atom is represented as a 128-dimensional feature vector. The transformed node features collectively encode the entire molecular graph, ensuring that the extracted representations preserve essential chemical and topological properties. To model drug interactions, the feature representations of the two drug molecules are concatenated, forming a joint feature vector that captures potential interactions. This concatenation step integrates both local atomic features and global molecular structures, enabling the model to learn patterns indicative of interaction likelihood. The combined feature vector is then passed through a fully connected feedforward network consisting of three dense layers with 256, 128, and 64 neurons, each followed by the ReLU activation function. These layers allow the model to refine the learned representations and capture more complex interaction relationships. Finally, a single neuron output layer with a sigmoid activation function produces a probability score indicating the likelihood of an interaction between the two drugs, making the model suitable for binary classification tasks. The Adam optimizer is employed for parameter optimization, with a learning rate set to 0.001, determined experimentally as the optimal value. The model is trained using the binary cross-entropy loss function, which effectively handles class imbalances and ensures stable convergence. The hyperparameters of the developed GCN architecture are provided in Table 2.

***Table 2. Hyperparameters of the designed GCN architecture.***

| Hyperparameter | Value |
|---|---|
| GCN Layer | 128-128 |
| Dense Layer Neuron Number | 256, 128, 64 |
| Learning Rate | 0.001 |
| Optimization Algorithm | Adam |
| Activation Functions | ReLU, Sigmoid |
| Loss Function | Binary Cross-entropy |

# 4 EXPERIMENTAL RESULTS

## 4.1 Performance Metrics

The classification performance of the developed GCN-based deep learning model is comprehensively evaluated through various performance metrics. The model's performance was assessed during the training and validation phases using evaluation metrics such as accuracy, precision, and area under the ROC curve (AUC). Precision helps measure the proportion of positive predictions made by the model that are actually correct. AUC quantifies the model's ability to distinguish between positive and negative classes, representing the trade-off between the true positive rate and false positive rate. Accuracy, which evaluates the overall prediction correctness of the model, expresses the proportion of correct predictions across all instances. However, due to the class imbalance in the dataset, the accuracy metric alone may not fully reflect the model's performance. Therefore, Cohen's kappa coefficient was also used. This metric is particularly useful for imbalanced datasets and measures the agreement between the model's predictions and the true labels, accounting for chance. The kappa coefficient ranges from -1 to +1, with 0 indicating no agreement beyond chance. Additionally, the zero-one loss metric, which directly measures the error rate, represents the proportion of incorrect predictions made by the model. This metric indicates how many instances the model misclassifies out of the total and provides insight into the model's generalization capability.

## 4.2 Model Training and Validation

The data sets were randomly divided into training and evaluation. The proposed model was run under the same conditions for each data set. All steps of the proposed method were implemented using Python-based open-source libraries. The experiments used a single NVIDIA GeForce GTX 3070 GPU machine with Intel Core i7-11700H CPU @ 4.90 GHz and 32 GB RAM. In model training, the data set was randomly divided into training and validation data in a 7:3 ratio. The validation data of the model was selected without applying the SMOTE method and was obtained from the data that was not processed during the training process. The model parameters were measured and recorded at each epoch during training. Early stopping was used during model training. The training was completed in a maximum of 100 epochs. The model weights, updated after each training step, were saved for use in the validation process. As a result, the best-performing weights, based on validation performance, were selected and used in the final model deployment. The model's performance during training is illustrated in Figure 2.

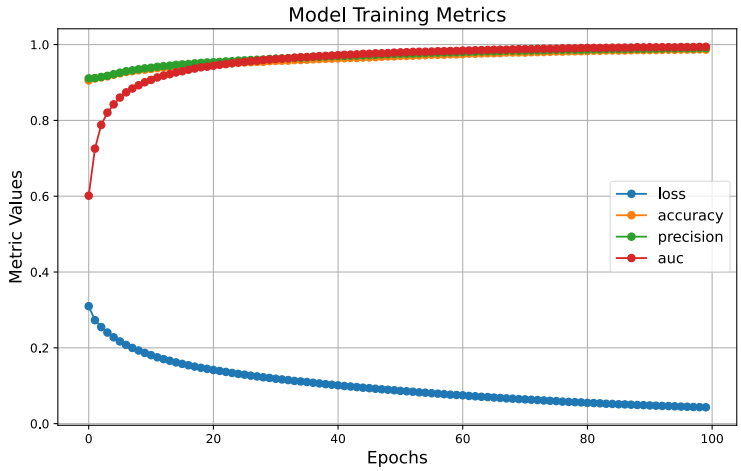



***Figure 2. Performance metrics of the model during training.***

The training loss value starts at 0.309873 and decreases steadily throughout the epochs, reaching 0.043219 at epoch 100. This decrease confirms that the model is learning the training data gradually, and its predictions are becoming more precise. This continuous decrease in the loss function indicates that the optimisation algorithm works effectively and that the model parameters are updated in the appropriate direction. Training accuracy, which refers to the rate at which the model correctly classifies the training data, was 0.905307 in the first epoch. This value indicates that the model initially shows a strong classification performance. The accuracy value increased continuously in the following epochs and reached 0.986721 at epoch 100. This increase suggests that the model's accuracy has become more consistent in positively predicting data. The precision metric measures the proportion of positive predictions made by the model that are actually correct. The precision, which started at 0.910889 in the first epoch, gradually increased during the training period and reached 0.990274 in the 100th epoch. This result shows that the model successfully correctly predicts positive class instances, and the false positive are reduced. Finally, the AUC value is an important metric that evaluates the model's ability to discriminate classes. The AUC value, which was 0.601326 at the beginning of the training period, increased steadily throughout the training, reaching 0.994136 at epoch 100. This shows that the model's ability to distinguish between positive and negative classes has improved significantly. The overall evaluation of these metrics during the training process indicates that the model's learning capability has effectively improved and adapted to the training data. A similar trend was observed during the validation process. The model's performance on the validation data is shown in Figure 3.

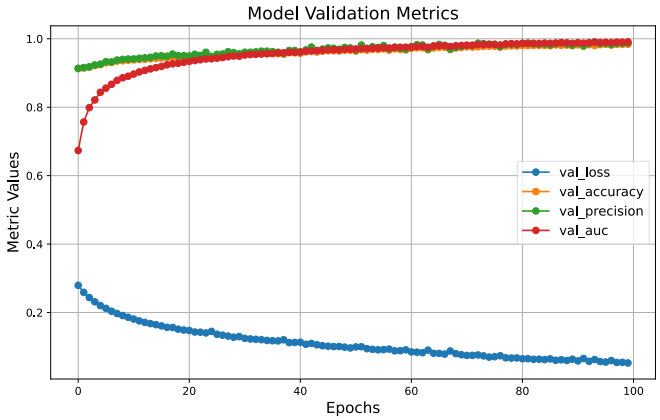



***Figure 3. Performance metrics of the model during validation.***

The validation step is critical to assess the generalisation ability of the model and its performance on unprecedented data. Validation loss, measuring the model's prediction errors on validation data, was recorded as 0.279408 in the first epoch. Although it showed a general decreasing trend during the training period, it decreased to 0.147460 at epoch 20 and fluctuated thereafter. However, by the 100th epoch, it decreased to 0.052584. This shows that the model successfully learns and generalises from the validation data but sometimes runs the risk of overfitting. Validation accuracy measures the correct classification rate of the model on the validation data. This value, which started at 0.912877 in the first epoch, generally increased during the training process and reached 0.983959 in the 100th epoch. This increase confirms that the model successfully classifies the validation data and has a good generalisation capability. Validation precision indicates how many of the model's false positive predictions on the validation data are positive, was set at 0.913406 in the first epoch and increased to 0.987654 in the 100th epoch. This result shows that the model successfully correctly predicted the positive class in the validation data, and the false positive predictions decreased. Validation AUC, which measures the model's success in discriminating classes on the validation data, started at 0.673383 in the first epoch and increased to 0.990869 in the 100th epoch. This increase reveals that the model's ability to discriminate classes in the validation data has continuously improved, and its generalisation ability has increased. In addition to these data, Cohen's Kappa value started at 0.892480. It reached 0.98, indicating that the model agreed significantly in its predictions. In contrast, the zero-one loss value decreased from 0.016906 to less than 0.01, indicating that the misclassification rate of the model decreased. When these metrics are evaluated, it is observed that the model performs well during the validation process. The optimal results achieved in both the training and validation phases are presented in Table 3.

*Table 3. Model training and validation results.*

| Metric | Loss | Accuracy | Precision | AUC | Cohen Kappa | Zero-One Loss |
|---|---|---|---|---|---|---|
| Training | 0.043219 | 0.986721 | 0.990274 | 0.994136 | 0.904540 | 0.014344 |
| Validation | 0.052584 | 0.983959 | 0.987654 | 0.990869 | 0.892480 | 0.016906 |

## 4.3 GUI Design

A user-friendly interface for predicting DDIs has been designed. The interface is developed in Python using the PyQt library. It allows users to enter drug names, view the molecular structures of drugs in 2D, display SMILES strings and predict potential interaction types. The main objective is to make the deep learning-based model accessible to a broader user base for DDIs prediction. The application is designed to help even users who are not drug interaction experts easily understand complex interactions. The home screen of the GUI designed for CNS-DDI is given in Figure 4.

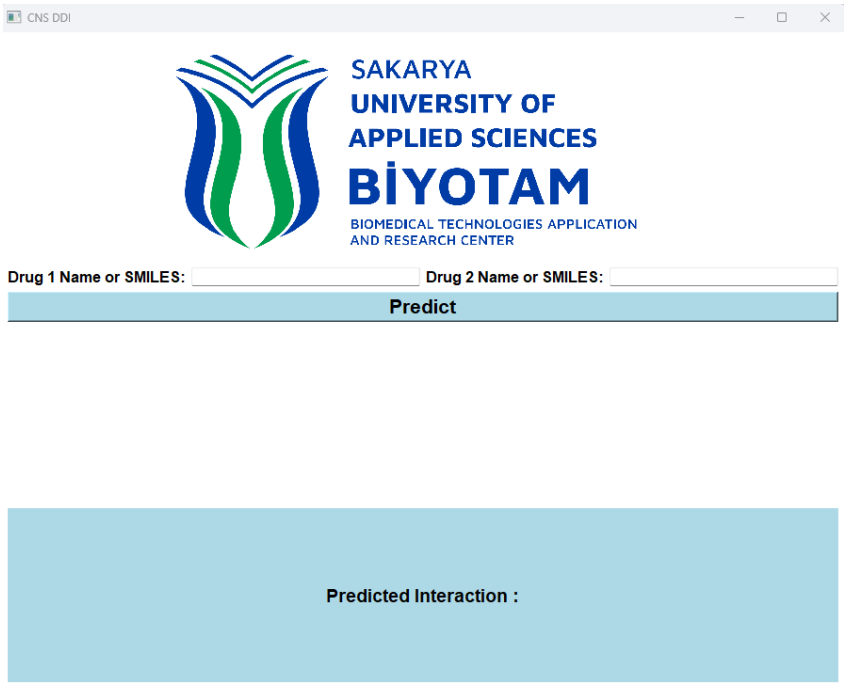



*Figure 4. Home screen of the GUI designed for CNS-DDI.*

The designed GUI allows drug interactions to be analyzed using graph-based neural networks. The GUI allows the user to provide the names of two different drugs or their SMILES representations as input. Provided that the user provides the names of the drug pairs, the GUI searches for these drug names in the DrugBank dataset and finds the SMILES strings of the drugs. The SMILES strings are then translated into graph format. The graph forms and other

physical properties of the drug molecules are applied as input to the deep learning model developed in this study. The interactions of the input drug pairs are then predicted by the model. The main components of the interface include input fields where drug names or SMILES strings can be entered, a component that visualizes the two-dimensional molecular structures of drugs, and an output section that presents the predicted drug-drug interaction in textual form. For molecular visualization, the RDKit library was used to generate high-resolution 2D molecular drawings directly from SMILES codes. After entering the relevant drugs, the user can click on the "Predict" button to view the interaction predicted by the model. Prediction results are shown under the "Predicted Interaction" heading. All these features make the application a valuable tool for both academic research and practical applications in the field of drug interactions. Figure 5 shows an example of a drug interaction predicted using the developed CNS-DDI method. Figure 5 shows the interaction results of two drugs that were not used for model development during the testing and training process. In this way, the GUI, which receives the names of the drugs or SMILES strings as input, predicts the type of interaction.

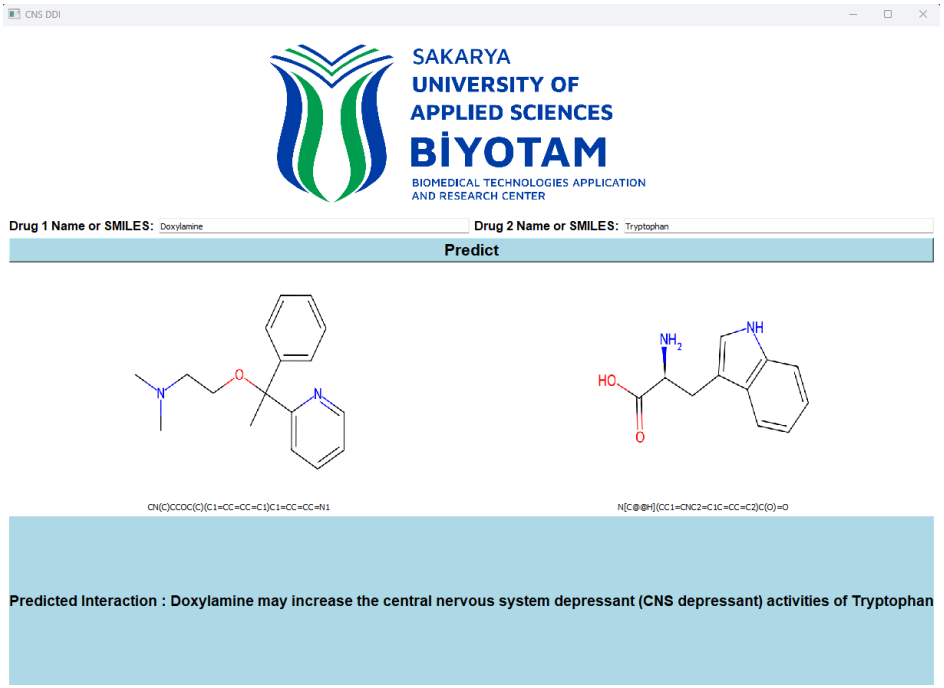



***Figure 5. GUI was developed for CNS related DDI prediction and is an example of interaction.***

In addition, a control mechanism has been added to determine the correctness and validity of the SMILES strings and drug names given as input in the designed GUI. If the SMILES string of a drug is typed in the input panel, the algorithm first checks the molecular validity of this SMILES string and then predicts the interaction of these two drugs. Similarly,

if the user provides the drug name as input, the interface checks the drug name from the DrugBank database and then obtains the SMILES string from the DrugBank database. In case the SMILES string or the drug name is invalid, the CNS-DDI model does not predict and notifies the user. Figure 6 shows an example of an invalid SMILES string to illustrate this situation.

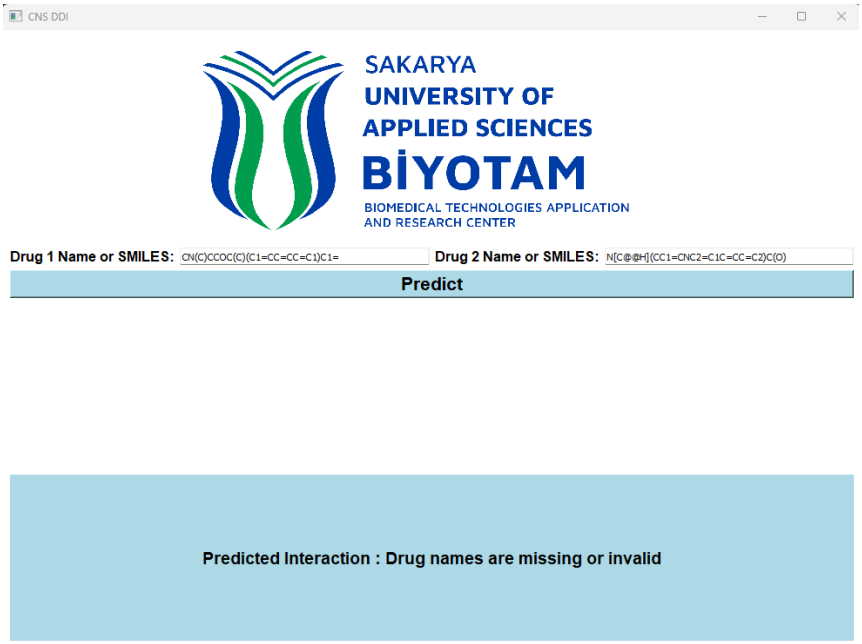



***Figure 6. Screenshot of the GUI warning the user if the drug name or SMILES string is invalid.***

## 5 CONCLUSION

In this study, a novel deep learning method is developed to predict drug interactions affecting the central nervous system. The proposed method models the molecular structures of drugs using a graph-based approach and integrates this structural data with relational information from interaction networks. This approach aims to learn interaction patterns and make reliable DDIs predictions by considering the atomic-level properties of drugs and their neighborhood relationships. The methodology used in the study includes CNS-related drug interaction data from the DrugBank database and SMILES strings representing the chemical structures of these drugs. In the DrugBank dataset, drug pairs without SMILES strings or physicochemical properties were removed from the selected data after text mining. Then SMILES strings were standardized with tokenization and padding techniques to make them suitable for the input format of the model. Additional properties of drug molecules, such as molecular weights, logP values and monoisotopic mass, were also included in the input data to

improve the model's performance. The developed GCN-based model is designed to address the molecular structure of drugs as a graph structure to increase the understanding of the complex patterns and interactions in the molecular structure. The model achieved 98.67% accuracy, 99.03% precision, and 0.994 AUC during the training process. The model performed similarly well in the validation step, reaching 98.40% accuracy, 98.77% precision and 0.991 AUC. In addition, Cohen's Kappa coefficient, which measures the model's success compared to chance-based predictions, was calculated as 0.904 in training and 0.892 in validation, indicating that the model achieved significant agreement in its predictions. The loss of zero one was 0.014 in training and 0.017 in validation, proving that the model's misclassification rate is extremely low. These results show that the model can make highly accurate and reliable predictions on both training and validation data. Another important outcome of the study is a GUI application designed to make the developed model usable by a wider user community. The proposed method provides a more comprehensive analysis than existing methods by simultaneously modelling the molecular structure of drugs and their relational structure in interaction networks. The model, which has high accuracy rates, can make reliable DDIs predictions and is expected to contribute significantly to future drug discovery studies. The developed model and GUI are valuable for academic research and practical applications. It aims to improve the model further and increase its generalizability by using different data sets and deep learning architectures. Future research will focus on increasing the scope and accuracy of the models. Improving the predictive ability for more diverse drug pairs involving different interaction mechanisms is a priority. In particular, it is planned to integrate additional data layers to model the relationships of drugs with protein targets such as metabolizing enzymes, transporters, receptors, etc. that influence their pharmacokinetic and pharmacodynamic profiles. This may involve the use of more sophisticated machine learning techniques, such as graph-based deep learning architectures with attention mechanisms or graph-based deep learning architectures that combine molecular representations with relevant biological information. In addition, the potential of deep learning for the discovery of new chemical entities with targeted effects on the CNS is being exploited. In particular, approaches such as computationally designing and screening molecules with desired properties using generative deep learning models will be evaluated. Such advances are expected to contribute to both the refinement of deep learning models and the development of new CNS therapies.

## Statement of Research and Publication Ethics

This study does not involve human participants, animal subjects, or any clinical data; therefore, it does not require ethical approval. The study is complied with research and publication ethics.

## Artificial Intelligence (AI) Contribution Statement

Artificial intelligence tools were used solely for language and grammar correction during the preparation of this manuscript. No AI tools were involved in the research design, data analysis, interpretation of results, or writing of the scientific content. All intellectual contributions and scholarly work were carried out exclusively by the author.

## REFERENCES


[1] T. V Dunwiddie and S. A. Masino, "The role and regulation of adenosine in the central nervous system," *Annu. Rev. Neurosci.*, vol. 24, no. 1, pp. 31–55, 2001.

[2] J. F. Thayer, A. L. Hansen, E. Saus-Rose, and B. H. Johnsen, "Heart rate variability, prefrontal neural function, and cognitive performance: the neurovisceral integration perspective on self-regulation, adaptation, and health," *Ann. Behav. Med.*, vol. 37, no. 2, pp. 141–153, 2009.

[3] P. Brodal, *The central nervous system*. oxford university Press, 2010.

[4] D. E. Pankevich, B. M. Altevogt, J. Dunlop, F. H. Gage, and S. E. Hyman, "Improving and accelerating drug development for nervous system disorders," *Neuron*, vol. 84, no. 3, pp. 546–553, 2014.

[5] A. S. Kesselheim, T. J. Hwang, and J. M. Franklin, "Two decades of new drug development for central nervous system disorders," *Nat. Rev. Drug Discov.*, vol. 14, no. 12, p. 815, 2015.

[6] D. J. Begley, "Delivery of therapeutic agents to the central nervous system: the problems and the possibilities," *Pharmacol. Ther.*, vol. 104, no. 1, pp. 29–45, 2004.

[7] E. C. M. de Lange, "The mastermind approach to CNS drug therapy: translational prediction of human brain distribution, target site kinetics, and therapeutic effects," *Fluids Barriers CNS*, vol. 10, pp. 1–16, 2013.

[8] A. Misra, S. Ganesh, A. Shahiwala, and S. P. Shah, "Drug delivery to the central nervous system: a review," *J Pharm Pharm Sci*, vol. 6, no. 2, pp. 252–273, 2003.

[9] C. Palleria *et al.*, "Pharmacokinetic drug-drug interaction and their implication in clinical management," *J. Res. Med. Sci. Off. J. Isfahan Univ. Med. Sci.*, vol. 18, no. 7, p. 601, 2013.

[10] T. Roblek, T. Vaupotic, A. Mrhar, and M. Lainscak, "Drug-drug interaction software in clinical practice: a systematic review," *Eur. J. Clin. Pharmacol.*, vol. 71, pp. 131–142, 2015.

[11] L. Magro, U. Moretti, and R. Leone, "Epidemiology and characteristics of adverse drug reactions caused by drug–drug interactions," *Expert Opin. Drug Saf.*, vol. 11, no. 1, pp. 83–94, 2012.

[12] E. N. van Roon *et al.*, "Clinical relevance of drug-drug interactions: a structured assessment procedure," *Drug Saf.*, vol. 28, pp. 1131–1139, 2005.

[13] T. Zhang, J. Leng, and Y. Liu, "Deep learning for drug–drug interaction extraction from the literature: a review," *Brief. Bioinform.*, vol. 21, no. 5, pp. 1609–1627, 2020.

[14] M. A. Pala and M. Z. Yıldız, "Improving cellular analysis throughput of lens-free holographic microscopy with circular Hough transform and convolutional neural networks," *Opt. Laser Technol.*, vol. 176, p. 110920, 2024.

[15] M. A. Pala, M. E. Çimen, A. Akgül, M. Z. Yıldız, and A. F. Boz, "Fractal dimension-based viability analysis of cancer cell lines in lens-free holographic microscopy via machine learning," *Eur. Phys. J. Spec. Top. 2021*, pp. 1–12, Nov. 2021, doi: 10.1140/EPJS/S11734-021-00342-3.

[16] K. Han *et al.*, "A review of approaches for predicting drug–drug interactions based on machine learning," *Front. Pharmacol.*, vol. 12, p. 814858, 2022.

[17] Y. Deng, X. Xu, Y. Qiu, J. Xia, W. Zhang, and S. Liu, "A multimodal deep learning framework for predicting drug–drug interaction events," *Bioinformatics*, vol. 36, no. 15, pp. 4316–4322, 2020.

[18] X. Li, Z. Xiong, W. Zhang, and S. Liu, "Deep learning for drug-drug interaction prediction: A comprehensive review," *Quant. Biol.*, vol. 12, no. 1, pp. 30–52, 2024.

[19] Y.-H. Feng and S.-W. Zhang, "Prediction of drug-drug interaction using an attention-based graph neural network on drug molecular graphs," *Molecules*, vol. 27, no. 9, p. 3004, 2022.

[20] J. You, R. D. McLeod, and P. Hu, "Predicting drug-target interaction network using deep learning model," *Comput. Biol. Chem.*, vol. 80, pp. 90–101, 2019.

[21] E. Kim and H. Nam, "DeSIDE-DDI: interpretable prediction of drug-drug interactions using drug-induced gene expressions," *J. Cheminform.*, vol. 14, no. 1, p. 9, 2022.

[22] X. Han, R. Xie, X. Li, and J. Li, "Smilegnn: drug–drug interaction prediction based on the smiles and graph neural network," *Life*, vol. 12, no. 2, p. 319, 2022.

[23] S. Vilar, R. Harpaz, E. Uriarte, L. Santana, R. Rabadan, and C. Friedman, "Drug—drug interaction through molecular structure similarity analysis," *J. Am. Med. Informatics Assoc.*, vol. 19, no. 6, pp. 1066–1074, 2012.

[24] H. Ibrahim, A. M. El Kerdawy, A. Abdo, and A. S. Eldin, "Similarity-based machine learning framework for predicting safety signals of adverse drug–drug interactions," *Informatics Med. Unlocked*, vol. 26, p. 100699, 2021.

[25] S. Jain, E. Chouzenoux, K. Kumar, and A. Majumdar, "Graph regularized probabilistic matrix factorization for drug-drug interactions prediction," *IEEE J. Biomed. Heal. Informatics*, vol. 27, no. 5, pp. 2565–2574, 2023.

[26] Z. Li, X. Tu, Y. Chen, and W. Lin, "HetDDI: a pre-trained heterogeneous graph neural network model for drug–drug interaction prediction," *Brief. Bioinform.*, vol. 24, no. 6, p. bbad385, Nov. 2023, doi: 10.1093/bib/bbad385.

[27] X. Lin, Z. Quan, Z.-J. Wang, T. Ma, and X. Zeng, "KGNN: Knowledge Graph Neural Network for Drug-Drug Interaction Prediction," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, {IJCAI-20}*, C. Bessiere, Ed., International Joint Conferences on Artificial Intelligence Organization, 2020, pp. 2739–2745. doi: 10.24963/ijcai.2020/380.

[28] Y. Zhang *et al.*, "Emerging drug interaction prediction enabled by a flow-based graph neural network with biomedical network," *Nat. Comput. Sci.*, vol. 3, no. 12, pp. 1023–1033, 2023.

[29] J. Gao, Z. Wu, R. Al-Sabri, B. M. Oloulade, and J. Chen, "AutoDDI: Drug–Drug Interaction Prediction With Automated Graph Neural Network," *IEEE J. Biomed. Heal. Informatics*, 2024.

[30] H. Liu, S. Li, and Z. Yu, "Predicting drug-drug interactions using heterogeneous graph neural networks: HGNN-DDI," *Appl. Comput. Eng.*, vol. 79, pp. 77–89, 2024.

[31] C. Knox *et al.*, "DrugBank 6.0: the DrugBank knowledgebase for 2024," *Nucleic Acids Res.*, vol. 52, no. D1, pp. D1265–D1275, 2024.

[32] S. Kim *et al.*, "PubChem substance and compound databases," *Nucleic Acids Res.*, vol. 44, no. D1, pp. D1202–D1213, 2016.

[33] G. Landrum, "Rdkit documentation," *Release*, vol. 1, no. 1–79, p. 4, 2013.

[34] M. R. Prusty, T. Jayanthi, and K. Velusamy, "Weighted-SMOTE: A modification to SMOTE for event classification in sodium cooled fast reactors," *Prog. Nucl. Energy*, vol. 100, pp. 355–364, 2017.

[35] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE Trans. neural networks Learn. Syst.*, vol. 32, no. 1, pp. 4–24, 2020.

[36] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: a comprehensive review," *Comput. Soc. Networks*, vol. 6, no. 1, pp. 1–23, 2019.

[37] A. Akgul, Y. Karaca, M. A. Pala, M. E. Çimen, A. F. Boz, and M. Z. Yildiz, "Chaos Theory, Advanced Metaheuristic Algorithms And Their Newfangled Deep Learning Architecture Optimization Applications: A Review," *Fractals*, vol. 32, no. 03, p. 2430001, 2024.