ISSN: 2454-9940

# INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail :
editor.ijasem@gmail.com
editor@ijasem.org

www.ijasem.org

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

https://zenodo.org/records/11635160

Vol 18, Issue 2, 2024

# Deep Side A Deep Learning Framework for Drug Side Effect Prediction

**Mohd Shahid Afridi (160520733045) [1], Shaik Yaseen (160520733046) [2], Zaheer Uddin Ghazi (160520733167) [3] Md.Zainlabuddin [4] Dr.Mohammed Jameel Hashmi [5],**

[1,2,3] B.E Student, Dept. of Computer Science and Engineering, ISL Engineering College

[4]Assistant Professor (PhD), Dept. of Computer Science and Engineering, ISL Engineering College

[5] HOD, Dept. of Computer Science and Engineering, ISL Engineering College

## ABSTRACT

***When medications go off the rails in clinical trials because of adverse effects, it's bad news for everyone involved: trial subjects and sponsors alike. It is possible that algorithms that can predict adverse effects can guide the creation of new medications. The LINCS L1000 dataset provides a wealth of information on context-specific traits; it is a compilation of gene expression data from cell lines that has been affected by various drugs. The present benchmark for context-specific data just considers the high-quality LINCS L1000 trials, excluding all others. In this study, we want to find the best way to utilize this data so that we can make better predictions. We evaluate five distinct deep learning architectures. We show that among multi-layer perception-based architectures, a multi-modal design provides the best prediction performance when employing drug chemical structure (CS) and the whole set of drug changed gene expression profiles (GEX) as modalities. When compared to the GEX, we generally find the CS to provide more light. The most effective model, based on convolutional neural networks and using solely SMILES string representations of the drugs, outperformed the state-of-the-art by 13% in macro-AUC and 3% in micro-AUC. Additionally, we show that the model can predict medication-side effect pairs that were missing from the ground truth side effect dataset but were mentioned in the literature.***

## Introduction

One promising computational approach to lowering the health and financial risks of medicine development is the prediction of possible side effects prior to entering clinical trials. For the purpose of drug side effect prediction, several learning-based approaches have been proposed. These approaches use a variety of features, such as drug structures, drug-protein interactions, metabolic network activity, pathways, phenotype data, gene annotations, and metabolic network activity. Recently, deep learning models have been used for adverse impact forecasting, in addition to the previously mentioned methodologies. For instance, (i) [31] integrates semantic, biological, and chemical data relevant to drugs with case reports and clinical notes, and (ii) [4] evaluates the effectiveness of side effect prediction using various chemical fingerprints retrieved using deep architectures.
These approaches rely entirely on external drug information (such as drug-protein interactions) and do not take into account cell or condition specific factors (such as dose) when predicting adverse drug responses (ADRs, which are the same thing as medication side effects). In order to address this issue, Wang et al. (2016) examined data from the LINCS L1000 project [32]. Here we monitor gene expression in a panel of human cell lines in response to a broad range of small-molecule chemicals and medicines. By making use of the gene expression patterns of the treated cells, a comprehensive, objective, and cost-effective ADR prediction is now possible [32]. There are a lot of labels used to classify things in the article. Their research lends support to the theory that context-dependent information is provided by gene expression patterns to the side-effect prediction task. Even though there are a total of 20,338 substances in the LINCS dataset and 473,647 tests for them, their method utilizes the optimal

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

https://zenodo.org/records/11635160

Vol 18, Issue 2, 2024

experiment for each drug to minimize noise. Consequently, a lot of expression data is just sitting there, collecting dust, even if it may improve prediction accuracy. Their technique also employs a kind of feature engineering called biological term enrichment vectors by using gene expression features. In this research, we want to find out whether a deep learning framework can use integrated data on drug structures and gene expression more effectively, without using feature engineering.

## Literature review

An anti-phishing technique based on the visual Visual content of the receiving web page is the basis for similarity. The Spoof Catch phishing detection program was created by Wilayat et al. [1] and is based on visual resemblance. At the initial visit to a website, Spoof Catch detects the login page and stores a snapshot locally. It compares the user's locally saved screenshots of the login page with the screenshots taken the next time they visit the same website. If the hosts of the received and previously visited login pages match, we know the host is legitimate; otherwise, we know it is phishing. In [15], an effective method is proposed for distinguishing between legitimate websites and those that are suspected of being phishing. In order to determine whether the two sites are confusingly similar, this method makes use of three important web properties. These features include the content and its arrangement, images included inside the page, and the browser's overall visual representation of the website. A data collection including 41 actual phishing sites in addition to their true counterparts yielded impressive results in terms of the mistake rate in an experimental test. In [16], the authors propose a new method of phishing protection that takes into account the intricate spatial design aspects of websites. Specifically, two methods are proposed for extracting the spatial organization features as rectangle parts from a given webpage. By comparing the two websites with their unique spatial layout qualities that account for features of their spatial architecture, we may infer that the two pages are identical. To catalog every feature of a legitimate page collection's spatial arrangement, an R-tree is constructed. Therefore, the R-tree allows for suitable spatial inquiries, which aid in phishing detection based on similarities of the spatial arrangement element. In their study, Zhang et al. [13] employed a content-focused approach to identify harmful phishing methods. A whopping 95% of phishing URLs were correctly identified using the suggested technique that relies on the Term Frequency-Inverse Document Frequency (TF-IDF) filter [17]. In order to protect clients against phishing attacks, the authors of the [18] suggested using the browser plug-in PWDHASH++. A way to find aesthetic parallels between the two websites was proposed by the writers. The proposed approach, which has its roots in Gestalt theory [19], views a web page as an integrated whole. Algorithmic complexity analysis is used to formally analyze these indivisible super signals.

### HYBRID APPROACH FOR PHISHING DETECTION

In [20], the authors construct a feature that can identify phishing and spoofing in many dimensions. The deep learning algorithm is the backbone of this two-stage process. The authors put forward a deep learning-based Dynamic Category Decision Algorithm (DCDA). This approach was used to process over one million malicious URLs. It took less time to identify web-spoofing using their defense system that was based on the suggested algorithm, according to the results. In their work, the authors provide a mixed machine learning strategy to combat phishing attacks [21]. A total of five machine learning methods have been used in the construction of this model. The proposed four-layer model was trained on the required dataset, which had a large number of URLs, and then compared to the current models. The results proved that the new approach worked better and faster. The RIPPER algorithm, developed by Kaur and Sharma [22], is used for the purpose of detecting harmful emails. One intriguing aspect of their solution is its ability to automatically construct and send an email to the target server upon detection of a phishing URL. The email stops all communication originating from the rogue server and provides the attacker's IP, location, and contact information. Machine learning and the Resource Description Framework (RDF) were used together by the authors of [24] to improve the accuracy of their proposed model and decrease the number of false positives. In their study, the authors used several machine learning techniques to identify phishing and dangerous websites. These techniques included Linear Model (LM), Decision Tree (DT), Random

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

https://zenodo.org/records/11635160

Vol 18, Issue 2, 2024

Forest (RF), and Neural Networks (NNs). The test data was then used for this purpose.

**ANTI-PHISHING MACHINE LEARNING TECHNIQUES**

A number of researchers have created robust, efficient, and trustworthy algorithms for dangerous URL detection using machine learning techniques. Some page attributes have been described by Mao et al. [26] for the purpose of identifying phishing URLs. Their own logistic regression classifier for detecting phishing domains served as a filter. Nearly 8.24% of daily visitors fell victim to phishing websites, out of millions of URLs. The writers evaluated nine machine learning algorithms in their work [14]. These algorithms included LR, RF, AdaBoost, SVM, NN, Naıve Bayes, Bagging, and Bayesian additive regression. The training data set was built by classifying 1500 phishing URLs using machine learning. Using a scalable classifier based on machine learning, the authors of [27] devised a fresh method to phishing detection. They trained the proposed model using the datasets that were noisy. According to their results, this approach was able to detect over 90% of the malicious URLs. In [28], a PART-algorithm is used with the aim of spoof detection. Using the MAP-REDUCE [29] technology, they have improved the detection technique. Jain et al. [30] reviewed all of the world's phishing detection systems in their comprehensive assessment. One that processes natural language using machine learning

**EXISTING SYSTEM:**

In the medical field, when one medication affects the way another one works pharmacologically, this is called a drug-drug interaction (DDI). Negative DDIs lead to severe medication responses, which may be fatal for patients or cause the medicine to be removed from the market. In contrast, positive DDIs often enhance patients' therapeutic results. Drug discovery and illness therapy now rely heavily on DDI identification. An existing system is used in this work to create a technique for DDI prediction using DDI-IS-SL, which stands for DDI based on integrated similarity and semi-supervised learning. Using the cosine similarity approach, DDI-IS-SL combines pharmacological, biological, and phenotypic data to determine how similar medications are in terms of their features. Also, using these known DDIs, we may determine how comparable medications are using the Gaussian Interaction Profile kernel. The scores representing the likelihood of an interaction between drugs are determined using a semi-supervised learning technique known as the Regularized Least Squares classifier. When compared to other approaches, DDI-IS-SL demonstrates superior prediction ability in 5-fold, 10-fold, and de novo drug validation. Furthermore, DDI-IS-SL outperforms its competitors in terms of average calculation time. Lastly, DDI-IS-SL's performance in actual applications is further shown by case studies.

## DISADVANTAGES OF EXISTING SYSTEM:

Data complexity: In order to identify drug side effects, most current machine learning algorithms need to correctly understand big and complicated datasets. • Availability of data: In order for machine learning algorithms to provide reliable predictions, they often need massive volumes of data. The reliability of the model could be compromised if there is a lack of data in enough amounts. Current machine learning algorithms can only learn as much as the data used to train them, which means that incorrect classification is a real possibility. The model's predictive abilities are severely limited if the data is mislabeled.

## PROPOSED SYSTEM:

Multiple-layer perception after taking all of the input vectors and applying a sequence of fully-connected (FC) layers, our MLP [22] model is ready to go. Batch normalization layers follow each FC layer [10]. With a drop probability of 0:2, we use ReLU activation [16] and dropout regularization [27]. To get the ADR prediction probabilities, the last layer's outputs are passed via the sigmoid activation function. The multi-label binary cross-entropy loss (BCE) is the loss function that is defined as the sum of negative log-probabilities across ADR classes. This system demonstrates the architecture of CS and GEX functionalities. Perception with residuals (ResMLP) The main difference between MLP and

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

https://zenodo.org/records/11635160

Vol 18, Issue 2, 2024

the residual multi-layer perception (ResMLP) design is the inclusion of residual connections between the fully-connected layers. For each intermediate layer, the input is added to its output element by element before moving on to the next layer for processing. The vanishing gradient issue may be significantly alleviated with these remaining connections [7]. Due to this, deeper architectures are able to train feature extractors, which may be more complicated and efficient with parameters. MMNN is short for "multi-modal neural network." Different multi-layer perception (MLP) sub-networks, which are part of the multi-modal neural network technique, are used to extract features from different types of input. Afterwards, the classification block receives fused outputs from these sub-networks. We examine two approaches, concatenation and summation, for feature fusion. The first one does element-wise summation, while the second one joins the domain-specific feature vectors into a bigger one. When it comes to summation-based fusion, the sub-networks for domain-specific feature extraction must be built with the intention of producing vectors of equal sizes. We call the MMNN networks that use concatenation MMNN.Concat and the MMNN networks that use summation MMNN.Sum. The goal of our multitask learning (MTL) based architecture, which is a multitask neural network (MTNN), is to include the ADReCS taxonomy-derived side effect categories. The method does this by outlining MLP sub-network blocks that are both generic and tailored to certain tasks. The combined GEX and CS feature set is sent into the shared block, which then produces a joint embedding. The multi-network is then transformed into a binary prediction score vector for a group of interconnected side-effect classes by each task-specific sub-network.

## ADVANTAGES OF PROPOSED SYSTEM:

For both training and testing purposes, the proposed system made use of many ml classifiers. To achieve precise accuracy on datasets, the suggested system built Convolutional Neural Networks (CNN), which is renowned for offering a potent means of automatically learning complicated characteristics in vision tasks.

## Methodology:

The following graphic shows the data being split into sets, which are then trained for various models. A training set (consisting of 80% of the dataset) and a pre-training set (20%) were first created. • Pre-train (80%) and pre-test (20%) were the two halves of the pre-training set. • The training set is now fractioned into a train set comprising 80% of the data and a validation set comprising 20%. There is another split in this train set: 80% into the train set and 20% into the test set. I now have two non-overlapping sets: one for train validation and one for test. • The best models for the given dataset were found using the pretrain set. I selected the top four models from the pretest batch. The mean absolute errors were used to compare their performance. The optimal parameter was chosen after tuning the hyper parameters of the top four models.

## MODULES:

## Data collection:

We will take Drug data set from Kaggle which has features as tweet data and labels as Drugor not.

## Data preprocessing:

The xtrain variable stores the features derived from the dataset, whereas the train variable stores the labels. We produce additional features and labels after preprocessing the data using a typical scalar function.

## Testing training:

pieces x and y of the data are passed to the testing and training functions at this point, which split it into four pieces. In order to send data to the algorithm, we use train variables, and to measure the algorithm's correctness, we use test variables.

**Initializing Multiple Algorithms and training with Logistic regression:**

The first step in training a machine learning algorithm is to define its parameters, such as the features and labels it will use. Predictions may be made when the data is modeled and saved in the system as a pickle file.

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

https://zenodo.org/records/11635160

Vol 18, Issue 2, 2024

We train a number of algorithms on the dataset, and then we compare the models' accuracy and choose the one with the highest score to make predictions.

## Predict data:

At this point, the web interface displays the results of the process, which include taking fresh data as input, loading learned models using pickle, preprocessing the numbers, and last, passing them on to the predict function.

## System architecture

Architecture of a system is a model that describes its structure, behavior, and other aspects from a conceptual standpoint. A formal depiction and explanation of a system is an architectural description. Structured to facilitate reasoning about the system's architecture and actions.

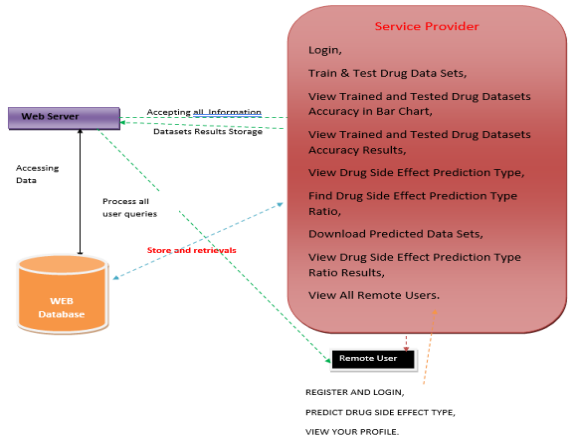



***Figure 1 System Architecture***

## 3-Tier Architecture:

In response to the shortcomings of the two-tier design, the three-tier software architecture (also known as three-layer architecture) arose in the 1990s. Intersecting the client-side user interface with the server-side data management components is the third tier, also known as the middle tier server. With features like queuing, application execution, and database staging, this middle tier can handle hundreds of users (in contrast to just 100 users with the two layer design) and manage processes where business logic and rules are implemented. An successful distributed client/server design that hides the complexities of distributed processing from the user while providing enhanced speed, flexibility, maintainability, reusability, and scalability is achieved using a three-tier architecture, as opposed to a two-tier design. The three-layer architecture has become the standard for net-centric information systems and Internet applications due to these features.

## Advantages of Three-Tier:

- Separates functionality from presentation.
- Clear separation – better understanding.
- Changes limited to well define components.
- Can be running on WWW.
- Effective network performance.

## HARDWARE REQUIREMENTS:

- System : Intel(R) Core(TM) i3-7020U CPU @ 2.30GHz
- Hard Disk : 1 TB.
- Input Devices : Keyboard, Mouse
- Ram : 4 GB.

### SOFTWARE REQUIREMENTS:

- Operating system : Windows XP/7/10.
- Coding Language : Python
- Tool : Anaconda
- Interface : OPENCV

## System testing

One of the most important parts of computer programming is the testing and debugging process. Without proper programming, the system would never be able to deliver the intended result. Asking user development for help finding all faults and flaws is the greatest way to do testing. Testing is done using the sample data. In testing, what counts is not

INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

ISSN2454-9940

www.ijasem.org

https://zenodo.org/records/11635160

Vol 18, Issue 2, 2024

the amount of data utilized, but its quality. The purpose of testing is to guarantee the system's accuracy and efficiency prior to directives for actual operation.

## Testing objectives:

The fundamental goal of testing is to find many mistakes in a methodical and efficient way. To put it officially, testing involves running software with the goal of discovering a bug.

- A successful test is one that uncovers an as yet undiscovered error.
- A good test case is one that has probability of finding an error, if it exists.
- The test is inadequate to detect possibly present errors.
- The software more or less confirms to the quality and reliable standards.

## Levels of Testing

**Code testing:**

The program's logic is examined in this. Using the provided files and folders as an example, we evaluated and confirmed the logic for updating different types of sample data.

**Specification Testing:**

Putting this specification into action begins with the program's intended behavior and how it should function under different scenarios. The whole system is tested using test cases that cover a wide range of scenarios and possible combinations of circumstances.

**Unit testing:**

We integrate and test each module separately during unit testing. With unit testing, the emphasis is on verifying the most fundamental aspect of the program architecture at the module level. This is another name for testing modules. We test each system module independently. This testing is executed while the software is being developed. By the end of the testing phase, we know that every module is producing the desired results. Additionally, there are field validation checks. For instance, in order to determine the correctness of the data submitted by the user, the validation check is executed. Launching the system is a breeze.

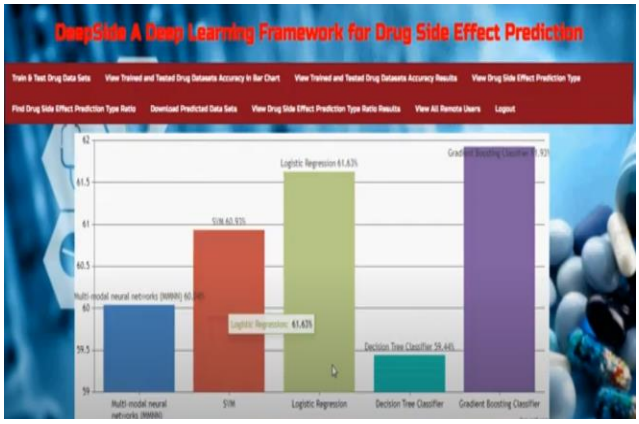
**Each Module can be tested using the following two Strategies:**

- Black Box Testing
- White Box Testing

## Output Screens

**Main Page**


Login page




User page

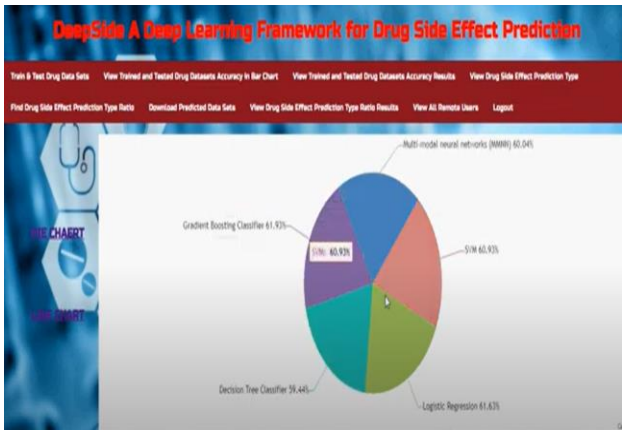
INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

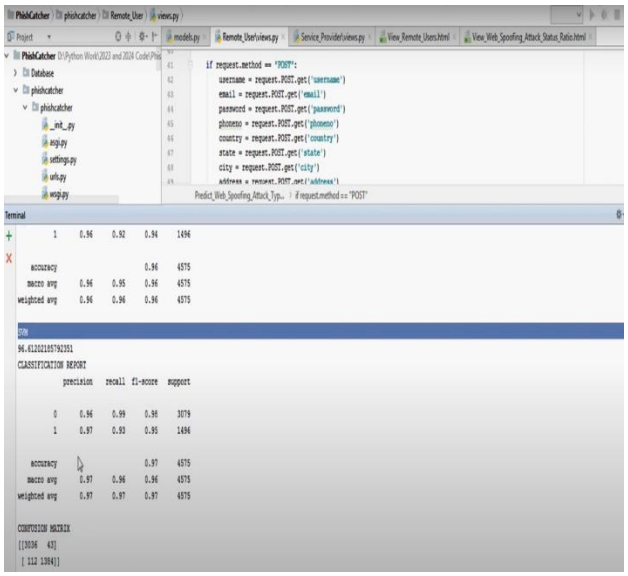
ISSN2454-9940

www.ijasem.org

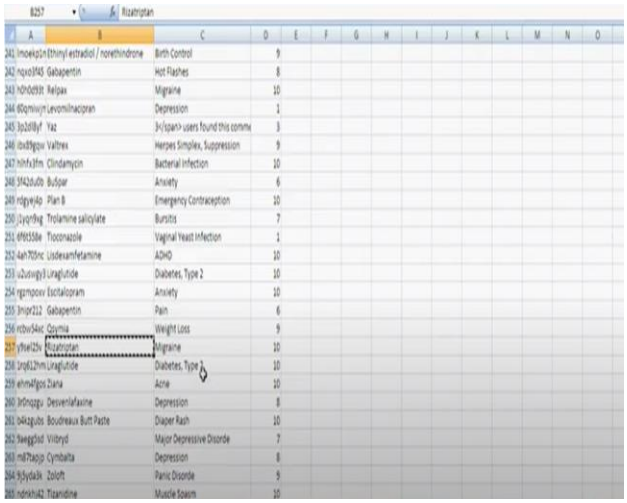
https://zenodo.org/records/11635160

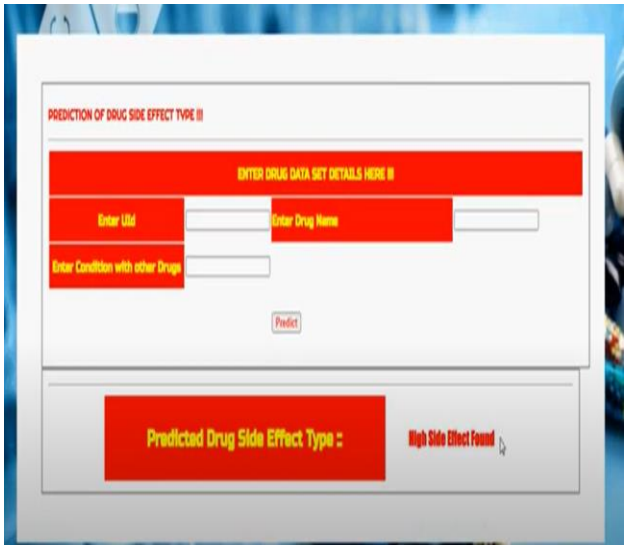
Vol 18, Issue 2, 2024




Dataset values




## CONCLUSION

It takes a lot of time and effort to produce a pharmaceutical medicine. The whole drug development process may have to be halted or restarted if unexpected adverse drug reactions occur throughout the procedure. So, it's crucial to anticipate the drug's negative effects a priori during the design process. To account for factors like dosage, time interval, and cell line, our Deeside system predicts ADRs using chemical structural information in conjunction with context-related (gene expression) characteristics. In comparison to models that rely

only on chemical structure (CS) fingerprints, the suggested MMNN model outperforms them in terms of accuracy by combining GEX and CS as features. Considering that our objective is to estimate the condition-independent side effects, the stated accuracy is significant. Lastly, by using convolution on the SMILES representation of the drug's chemical structure, the SMILESConv model surpasses all previous methods.

## REFERENCES

1. Atias, N., Sharan, R.: An algorithmic framework for predicting side e_ects of drugs. Journal of Computational Biology 18(3), 207{218 (2011)

2. Bresso, E., Grisoni, R., Marchetti, G., Karaboga, A.S., Souchet, M., Devignes, M.D., Sma_l-Tabbone, M.: Integrative relational machine-learning for understanding drug side-e_ect pro_les. BMC bioinformatics 14(1), 207(2013)

3. Cai, M.C., Xu, Q., Pan, Y.J., Pan, W., Ji, N., Li, Y.B., Jin, H.J., Liu, K., Ji, Z.L.: Adrecs: an ontology database for aiding standardization and hierarchical classi_cation of adverse drug reaction terms. Nucleic acids research 43(D1), D907{D913 (2014)

4. Dey, S., Luo, H., Fokoue, A., Hu, J., Zhang, P.: Predicting adverse drug reactions through interpretable deep learning framework. BMC Bioinformatics 19 (12 2018). https://doi.org/10.1186/s12859-018-2544-0

5. Dimitri, G.M., Li_o, P.: Drugclust: A machine learning approach for drugs side effects prediction. Computational Biology and Chemistry 68, 204 { 210 (2017). https://doi.org/https://doi.org/10.1016/j.compbiolchem.2017.03.008

6. Groopman, J.E., Itri, L.M.: Chemotherapy-induced anemia in adults: incidence and treatment. Journal of the National Cancer Institute 91(19), 1616{1634 (1999)

7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)

8. Huang, L.C., Wu, X., Chen, J.Y.: Predicting adverse side e_ects of drugs. BMC genomics 12(5), S11 (2011)

9. Huang, L.C., Wu, X., Chen, J.Y.: Predicting adverse drug reaction pro_les by integrating protein interaction networks with drug structures. Proteomics 13(2), 313{324 (2013)

10. Io_e, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)

11. Kim, S., Thiessen, P.A., Bolton, E.E., Chen, J., Fu, G., Gindulyte, A., Han, L., He, J., He, S., Shoemaker, B.A., et al.: Pubchem substance and compound databases. Nucleic acids research 44(D1), D1202{D1213 (2015)

12. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classi_cation with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097{1105 (2012)

13. Kuhn, M., Letunic, I., Jensen, L.J., Bork, P.: The sider database of drugs and side e_ects. Nucleic acids research 44(D1), D1075{D1079 (2015)

14. Landrum, G., et al.: Rdkit: Open-source cheminformatics (2006)

15. Lee, W., Huang, J., Chang, H., Lee, K., Lai, C.: Predicting drug side e_ects using data analytics and the integration of multiple data sources. IEEE Access 5, 20449{20462 (2017). https://doi.org/10.1109/ACCESS.2017.2755045

.