Waqar Kaleem Khan
01-249191-013

Predicting Drug Chemical Interactions Using Deep Learning

Masters of Science in Data Science

Supervisor: Dr.Muhammad Asfand E Yar

Department of Computer Science
Bahria University, Islamabad

November 18, 2021

MS-13
Thesis Completion Certificate

Student Name: **Waqar Kaleem Khan** Registration Number: **01-249191-013**
Program of Study: **Masters of Science in Data Science**
Thesis Title: **Predicting Drug Chemical Interactions Using Deep Learning**

It is to certify that the above student's thesis has been completed to my satisfaction and, to my belief, its standard is appropriate for submission for evaluation. I have also conducted plagiarism test of this thesis using HEC prescribed software and found similarity index at **14%** that is within the permissible set by the HEC. for MS/MPhil/PhD.

I have also found the thesis in a format recognized by the BU for MS/MPhil/PhD thesis.

Principle Supervisor's Signature:____________________
Principle Supervisor's Name: Dr.Muhammad Asfand E Yar

MS-14A
Author's Declaration

I, **Waqar Kaleem Khan** hereby state that my MS thesis titled **"Predicting Drug Chemical Interactions Using Deep Learning"** is my own work and has not been submitted previously by me for taking any degree from **"Bahria University, Islamabad"** or anywhere else in the country / world.

At any time if my statement is found to be incorrect even after my Graduate the university has the right to withdraw cancel my MS degree.

Waqar Kaleem Khan
01-249191-013
November 18, 2021

MS-14B
Plagiarism Undertaking

I, **Waqar Kaleem Khan** solemnly declare that research work presented in the thesis titled

Predicting Drug Chemical Interactions Using Deep Learning

is solely my research work with no significant contribution from any other person. Small contribution / help whenever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of Bahria University and the Higher Education Commission of Pakistan towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarised and any material used is properly referred / cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS degree, the university reserves the right to withdraw / revoke my MS degree and HEC and the university has the right to publish my name on HEC / University Website on which name of students who submitted plagiarised thesis are placed.

WAQAR KALEEM KHAN
01-249191-013
November 18, 2021

Abstract

Drug-drug interactions (DDIs) are one of the crucial concerns in pharmaceutical research. In the past decade, many researchers developed some machine learning based methods, but these ML methods focus on whether two drugs interact with each other not. The research in DDIs domain shows us that different subsequent even could be caused by DDIs which can be adverse effects or slowing the recovery process of a patient which is consuming multiple drugs at a time so for investigating the hidden mechanism behind the usage of multiple drugs at a time the prediction of drug-associated events can be more useful. Drug-to-drug interaction occurs when a patient consumes more than one drug at a time. Hence, due to using different drugs, any drug can influence the effect of another drug. The drug-to-drug interactions (i.e., DDI) are detected or identified using the pathways and enzymes interactions, therefore machine learning and deep learning techniques are used to find the DDI with each other. The deep learning models i.e., CNN, LSTM's, and RNN are used to analyze the DDI based on the 65 different types of drug interaction and its associated events using the selected database. The inputs used, in our model, out of the 65 types of drugs are smiles of drugs, enzymes, pathways of the drug to target, and the target. Therefore, the different number of layers, activation function, and features of drugs for the multi-model CNN, RNN, and LSTM's is used to achieve better accuracy, as compared to traditional prediction algorithms. We have done different experiments in terms of using different numbers of layers, activation functions, and different features of drugs the multi-model CNN model achieved an accuracy of 0.9000, F1-score of 0.8286, AUPR of 0.9478, AUC of 0.9981 the multi-modal LSTM's models achieved an accuracy of 0.8902, F1-score of 0.7792, AUPR of 0.9407 and AUC of 0.9978 and the multi-modal RNN's model achieved an accuracy of 0.8866, F1-score of 0.7779, AUPR of 0.9395 and AUC of 0.9979. The various computational experiments show that a combination of various drug features is performing better than one separate feature of drugs. Compared with other proposed methods like DDIMDL, DeepDDI, CNN-DDI, RF (Random Forest), KNN (K-nearest neighbor), LR (Linear Regression) our multi-modal CNN and LSTM's model method has better performance as compared to the other proposed methods mention above. While the RNN model results are better than LR, KNN, RF methods but approximately equal DDIMDL methods.
Availability The source code, complete results and data set are available in my GitHub repository. https://github.com/waqarkaleemkhan/Thesis-code-iteration-on-every-feature

Acknowledgments

Throughout the writing and implementation of this Thesis, I was in contact with many people, which include academicians, researchers, and practitioners.

First of all, I would like to thank my supervisor Dr. Muhammad Asfand-E-Yar for his encouragement, guidance, critics, advice, and motivation. I was pushed by your constructive suggestions to sharpen my analysis and to carry my work to a higher level.

I am extremely thankful to Yifan Deng for providing necessary technical suggestions as a Drug-drug interaction expert in the field of Machine learning DDIs.

I would also like to thank Wahidullah graphic designer who help me in Adobe Photo-shop to design my models.

Waqar Kaleem Khan
Islamabad, Pakistan

Contents

Abstract vii

1 Introduction 1
- 1.1 Introduction 1
- 1.2 Problem Statement 2
- 1.3 Objectives 2
  - 1.3.1 Existing Domain Work Analysis 2
  - 1.3.2 Data-set Selection 2
  - 1.3.3 Tunning Hyper Parameters of Model and Applying Different Models 3
  - 1.3.4 Final Implementation 3
- 1.4 Limitation and Scope of Research 3
- 1.5 Structure of Thesis 3

2 Literature Review 5
- 2.1 Traditionally Used Approaches for DDIs Classifications 5
- 2.2 Deep Neural Network (DNN) 5
- 2.3 Support Vector Machine (SVM) 6
- 2.4 Graph Convolutional Network (GCN) 7
- 2.5 Neural Networks 9
- 2.6 Autoencoders 11
- 2.7 Convolutional Neural Network (CNN) 12
- 2.8 Embedding Approach 13
- 2.9 Knowledge Graph and Graph Neural Networks (KG and GNN) 14
- 2.10 Convolutional LSTM's 15
  - 2.10.1 Summary of Literature 16

3 System Architecture and Research Design 19
- 3.1 Data Set 19
  - 3.1.1 History of Drug-Bank 20
- 3.2 Proposed Methodology 22
  - 3.2.1 Features Extraction 22
  - 3.2.2 1-D CNN as a Model 22
- 3.3 Recurrent Neural Network as a Model 24
  - 3.3.1 Our RNN Model Architecture 26
- 3.4 Long Short Term Memory (LSTM's) 27
  - 3.4.1 Our LSTM's Model Architecture 31

3.5 Cross-Entropy 32
3.5.1 Categorical cross-entropy 32
3.6 Optimizer 32
3.7 Drop Out 32
3.8 Batch Normalization 33

4 Analysis and Evaluation 35
4.1 Experimental Setup 35
4.2 Models Implementation 35
4.3 Evaluation Measure 35
4.3.1 Accuracy 36
4.3.2 Precision 36
4.3.3 Recall 36
4.3.4 F1-Score 36
4.3.5 AUC 36
4.4 CNN as a model 36
4.4.1 Inputs to the model 37
4.4.2 Layers 37
4.4.3 Implementation Details 37
4.4.4 Optimizer 37
4.4.5 Loss Function 38
4.4.6 Avoid Overfitting Techniques 38
4.4.7 Training and Testing Data 38
4.4.8 Evaluation Measure 38
4.5 RNN as a model 38
4.5.1 Inputs to the model 38
4.5.2 Layers 39
4.5.3 Implementation Details 39
4.5.4 Optimizer 39
4.5.5 Loss Function 39
4.5.6 Avoid Over-fitting Techniques 39
4.5.7 Training and Testing Data 40
4.5.8 Evaluation Measure 40
4.6 LSTM's as a model 40
4.6.1 Inputs to the model 40
4.6.2 Layers 40
4.6.3 Implementation Details 41
4.6.4 Optimizer 41
4.6.5 Loss Function 41
4.6.6 Avoid Over-fitting Techniques 41
4.6.7 Training and Testing Data 41
4.6.8 Evaluation Measure 41
4.7 Results 42
4.7.1 CNN model Results 42
4.7.2 CNN Each Event Results 43
4.7.3 RNN model Results 47
4.7.4 RNN Each Event Results 48

4.7.5 LSTM's model Results 51
4.7.6 LSTM Each Event Results 52
4.8 Comparing Results 54
4.9 Applying Drug-Drug Interaction Data set to Drug Target Interaction Existing Method 55

5 Conclusion **57**
5.1 Conclusion 57
5.2 Future Directions 58

References **62**

Index **63**

List of Figures

3.1 Proposed method 1 pipeline 23
3.2 Feed DNN with 3 hidden and one input and output layers 24
3.3 Simple RNN stucture 25
3.4 Proposed method 2 pipeline 27
3.5 LSTM's structure 28
3.6 LSTM's forget gate structure 29
3.7 LSTM's input gate structure 29
3.8 LSTM output gate structure 30
3.9 Proposed method 3 pipeline 31

4.1 Each and every event evaluation measures values using CNN's 44
4.2 Each and every event evaluation measures values using CNN's 45
4.3 CNN's ROC graph 46
4.4 Each and every event evaluation measures values using RNN's 48
4.5 Each and every event evaluation measures values using RNN's 49
4.6 RNN's ROC graph 50
4.7 Each and every event evaluation measures values using LSTM's 52
4.8 Each and every event evaluation measures values using LSTM's 53
4.9 LSTM's ROC graph 54

List of Tables

2.1 Literature Summary part 1.0 17
2.2 Literature Summary part 1.1 18

3.1 Drug bank history 21

4.1 CNN Model Layers and Hyper-parameters 38
4.2 RNN Model Layers and Hyper-parameters 39
4.3 LSTM Model Layers and Hyper-parameters 41
4.4 CNN results on different numbers of layers 42
4.5 CNN results on different activation function and Jacard Similarity measures 42
4.6 CNN results on different on different features and Jacard Similarity measures 43
4.7 RNN results on different layers and Jacard Similarity measures 47
4.8 RNN results on different on different features and Jacard Similarity measures 47
4.9 LSTM results on different numbers of layers and jacard similarity measures 51
4.10 LSTM results different features set and Jacard Similarity measures 51
4.11 Different Method Performance 55

Acronyms and Abbreviations

ANN	Artificial Neural Network
CNN	Convolutional Neural Network
Dl	Deep Learning
FDA	Food and Drug Administration
DDI	Drug Drug Interactions
Ml	Machine Learning
SVM	Support Vector Machine
LSTM’s	Long Short Term Memory
RNN’s	Recurrent Neural Network
CNN’s	Covolutional Neural Network
KG	Knowledge Graphs
KGN	Knowledge Graph Networks
SMILES	Simplified Molecular-Input-Line-Entry System

Chapter 1

Introduction

1.1 Introduction

In the last few decades, the speedy growth in the development of drugs has bestowed medical practitioners with additional options for treating the disease of patients. Therefore taking multiple drugs together can lead a patient to DDI. DDI occurs when a patient consumes more than one drug at a time together and one of which may influence the effectiveness of another drug. In the USA, a survey has been done in 2011 which shows that about 67% of senior citizens are consuming four or more different drugs at a time[1], hence consuming more than one drug at a time can lead to negative drug-drug interaction events or adverse drugs effect which can lead a patient to death[2],[3]. The observation of DDI has gotten substantial attention in drug safety surveillance and public health safety. For avoiding such types of events it is important to know more about DDI and the more DDIs we know it can also help in the fast cures of diseases like cancer, AIDS, asthma because the patient with these diseases needs a well-organized pair of drugs to interact positively and help in fast recovery. To make sure a positive impact on the treatment of patients, safe prescriptions, and avoid adverse effects it is important for medical practitioners to understand the DDIs, but mostly the DDIs are noticed after patients consume the drugs which cannot avoid the negative interaction of drugs on the patient health and cannot ensure the safety of the drugs. Using wet experiments for identifying DDIs is time taking, labor-intensive, and costly. For making the DDI prediction process more cost-effective and efficient many researchers proposed different calculation methods. There are three categories of computational prediction models which are similarity-based, network-based, and knowledge-based. The method based on knowledge-based uses text mining and statistical techniques for the extraction of DDIs from electronic medical records, spontaneous reports, and scientific literature. The similarity-based method is based on the two similar drugs may interact with each other or not. The method based on network-based methods uses the properties of drugs and the network structure of drugs to deduce the lack of interaction between drugs. From various sources like reports, literature, etc, many researchers have collected drug data and built different databases like Drug-bank, BioSNAP, and BindingDB which can help in the computational prediction method development for drug-drug interaction prediction. In this research, we will use multimodal deep learning and a Drug Bank version 5.0 data set for drug-drug interaction prediction.

1.2 Problem Statement

For the quick recovery and using combinations of the drugs is increasing in modern medication. This increase in combinations of drugs sometimes leads to adverse drug-drug interaction events and can cause health problems or slow down the recovery of a patient. Using the traditional method for predicting drug-drug interaction (i.e., DDI) is very slow, for quick recovery, it is important to speed up the process of predicting drug-drug interactions. Most researchers collected drug-related data and make it available publicly, which can be used for drug-drug interaction prediction using computational methods. For DDIs prediction various machine learning models are introduced but every model has its inherent issue. Currently, many researchers are exploring deep learning models for better accurate and effective DDIs predictions. In our proposed study we will be using a drug-bank dataset (reference required) and will apply various variations in the methods for better results as compared to the base paper.

1.3 Objectives

Below are some of the objects which will lead to major objectives Predicting Drugs Chemical Interaction using Deep Learning.

1.3.1 Existing Domain Work Analysis

Previous work in the field of drug-drug interaction prediction will be summarized and analyzed in literature review form. Which will help us to understand a detailed preview of what considerable techniques and models of deep learning and, machine learning is used for the drug-drug interaction prediction and previously and also the data-sets that are used for training the model mostly and will also help us in the identification of the research gap that is being tried to full-filled in our thesis.

1.3.2 Data-set Selection

A sufficient and valid data-set for the validation and training of the model will be selected. In the selection of data-set we will focused on the four drugs features like chemical structures, enzymes, pathways, target and interaction of drugs.

1.3.3 Tunning Hyper Parameters of Model and Applying Different Models

For the implantation and finer understanding of the problem, domain knowledge will be required. Different deep learning models like CNN, LSTM's, and RNN will be applied to the data set and the results of that models will be compared and will test, and tuned the models with different hyperparameters for better results.

1.3.4 Final Implementation

Finally we will apply different models on the data-set for achieving the desired results.

1.4 Limitation and Scope of Research

Drug bank available is recently designed, therefore none updated current or old data set or drug-bank is available. Therefore, we are applying the variation in the methods to check the results of the given base paper.

1.5 Structure of Thesis

Our thesis has five chapters which the first chapter is an introduction which is an introductory part of the thesis which includes the aspects which bring the public eye into the research problem. The sections include in the introductory chapter the study background, objectives of study, limitation, and scope of the study, and structure of the thesis. The second chapter is Literature Review this chapter includes the detailed literature review or we can say a background study for developing the objective of studies, hypothesis, and research questionnaire. In this chapter, we will discuss and analyze the prediction of drug-drug interaction, work done previously according to different deep learning and machine learning approaches. The third chapter is Proposed Methodology and Design of Research in this chapter we will discuss in detail the adopted methodology and utilized research approach, research parameters, incorporated research strategies, literature resources used, and theoretical framework. The fourth Chapter is Analysis and Evaluation in this chapter we will discuss the implementation of different models and the last part will contain the results of that models, and the last chapter is Conclusion in this chapter the work will be concluded.

Chapter 2

Literature Review

2.1 Traditionally Used Approaches for DDIs Classifications

In the discovery of drugs detection of drug-drug interaction is one of the most important tasks. Traditionally two approaches are used for drug-drug interaction detection are the following first one is Vivo which refers to when work or research is done within or with a whole living organism i.e. animal models studies and the second type is Vitro which refers to when the work or research done outside of leaving organism i.e. cell studies in culture. Also, there are other types of studies for the detection of DDIs which include preclinical and clinical studies. Preclinical have two types of studies which are (Evaluating Metabolism-Based Drug Interaction) and (Evaluating Transporter-Mediated Drug Interactions). In the Evaluating Metabolism-Based Drug Interaction different test systems can be used which include CYP enzymes recombinant, tissue of the human liver, and microcosms of the subcellular liver which can help the enzymes to identify that metabolize an investigational drug and find out whether the drug is the enzyme inducer or inhibitor. In Evaluating Transporter-Mediated Drug Interactions used a test system in which a cell-based and membrane vesicles system is included. These test studies can be used for whether the drug is an inhibitor or a substrate of different transporters Clinical has two types of studies (Substrates Index and Perpetrators Index) and (Expected Concomitant Use Drugs) In the Substrates Index and Perpetrators, Index knew enzymatic subtractors or inhibitors are co-administered with the investigational drug for the simulation of worst-case scenarios. In Expected Concomitant Use Drugs, the transporter-mediated DDIs are investigated.

2.2 Deep Neural Network (DNN)

A DNN is a type of artificial neural network with a certain level of complexity, or we can say a neural network that has at least one hidden layer between the input and the output layer. There are various types of deep neural networks but these DNN always contain the alike components like weights, functions, neurons, and biases, etc. The above-named components function similar to the human brain and we can train these networks like any other machine learning algorithm[4][5]. In this study[6] the drug features were generated for 5000 drugs using smiles (“simplified molecular-input line-entry system”) which is downloaded from a database named drug-bank for prediction of drug-drug interaction later then use these features of drugs to predict 80 types of DDIs in DNN. The DNN used in this study has 1 input layer, 4 hidden layers, and an output layer. Where the input layer has a drug pair of two drugs and 4,432 features of those pairs and the output layer shows the 80 different types of DDIs in form of probability these different 80 DDI types are presented by (“one-hot”). In the first hidden layer, there are 2000 nodes, in the second layer, there are 1000 nodes, in the third layer there are 500 nodes and in the last layer there are 250 nodes and the activation function is used in the between of every two layers is (“ReLu”). For the prediction of DDIs, numerous methods based on ML have previously been proposed but many of them predict the interaction between two drugs. In this study,[7] a framework of multi models deep-learning has been proposed which for DDI events prediction combine multiple features of drugs with deep-learning the model name is (“DDIMDL”). The model “DDIMDL” initially builds sub-models based on a “deep neural network” by using these features of drugs “pathways, targets, substructure, and enzymes” and then for combining the submodels use a DNN joint framework to learn the representation of cross-modality of a pair of drugs. The working of the model is that in the first step for calculating the similarities of drug-drug using four features of drugs and after calculating the similarities use it as a representation of drug. Then fed the representation to the submodels after that the submodels are combined to learn the drug pair’s “cross-modality representation“ after learning the representation the DDIs predicted. The dataset used in this study is named drug bank and KEGG the interaction of drugs in drug banks are described by many sentences which in terms of DDIs events hard to understand so for understanding the DDIs easily the authors define a new DIIs representation in terms of a tuple structure which are “action”, “mechanism”, and “drugA and drugB” where the action is the increase and the decrease in the lemmatization and the mechanism means the effects of drugs for dividing the DDIs in a structure of four-touple “StandfordNLP is used”.

2.3 Support Vector Machine (SVM)

SVM is the supervised learning model in machine learning with associated learning algorithms that analyze data for regression and classification analysis. support vector machine builds a hyperplane or hyperplanes set in an infinite or high dimensional space which we can use for the regression and classification or finding outliers detection in data. The hyperplane with a high margin or distance a good separation can be archived or in general, we can say the higher the distance or the margin the classification generalization error will be lower[8]. As we know that many proposed computational models for DDIs prediction but these methods are facing many challenges in DDIs prediction because of the not verified negative samples experimentally for overcoming this problem the authors come up with a new solution and proposed a model named "DDI-PULearn"[9] to predict drug-drug interaction on large scale. The model use ("OCSVM") and ("KNN") for reliable negative seeds generation. Then use the negative seeds which are generated and all the valid DDIs (Positive labels) for training. For identifying the negative seeds from samples which is unlabeled the model is using an iterative "SVM" then by a method based on similarity the model represents abundant drug properties as a vector in form of positive label and negative identified seeds. Then the model uses PCA ("Principle Component Analysis") for transforming the vectors into low-dimensional space and uses the vector for binary classification as an input. Four different datasets are used from the (Drug-Bank) dataset substructures of chemical, substituents of drugs, and targets of drugs are extracted, from the second dataset named (DrugCentral) the targets of drugs are extracted, from the third dataset named (SIDER) side effects of drugs are extracted and from the last dataset name (CTD) pathways of drugs and drug indication are extracted. This study[10] intended to build a machine learning (ML) model using support vector machines (SVMs) for the prediction of DDIs from multiple similarity measures which include 2D molecular structure, interaction profile fingerprint, target, 3D pharmacophoric, and drug adverse effect. These similarity measures were extracted from two different databases named Drug-Bank and Side Effect Resource(SIDER). For every drug-drug pair, the above-mentioned five similarities were computed and for each similarity, a separate database was created to store the calculated similarities and then use these five databases to calculate the similarity matrix for every drug pair and after creating the similarities matrix for the classification of the DDIs a pairwise kernel was implemented which is used to train the model of SVM classification on the matrix of similarity to classify the interaction between in a pair of drugs in the form of True or False.

2.4 Graph Convolutional Network (GCN)

A GCN is a semi-supervised learning approach on graph-structured data. This network-based on a convolutional neural network that operates on the graphs directly. The big difference between GCNs and CNNs is that the CNN is particularly built to work on structured data which is regular (Euclidean) and where the GCNs are the generalized CCNs version where the nodes are irregular or (non-Euclidean) and the number of nodes connection. The motive of the research paper [11] was to predict DDIs from the drug network structure features as we know that the detection of DDIs in wet-lab experiments is a very costly and time taking process so for rapid DIIs prediction it is extremely needed to provide computational methods. Normally for DDIs prediction available computational methods are using biological and chemical features of drugs that they extract from various drug-related properties. But some properties of a drug are not available easily or open source so sometimes it can be very costly to get those properties for DDIs predictions. The authors proposed a new model named "DPDDI" in which the network structure feature of the drug will be extracted from the network of DDIs using a "graph convolutional network" DDIs and a deep neural network is used to predict the interaction between drugs. In the network of DDIs, the topological relationship is captured by GCN to learn the "low dimensional feature representation of drugs". For the possible prediction of "drug-drug interaction" of any of two drugs, latent feature vectors concatenate as a feature vector of the same 5 drug pairs by DNN predictor. The data sets used in this paper are the drug bank 4.0 version which has 1562 drugs and 180576 explained drug-drug interactions. To evaluate the strength of the model the drug bank version 5.0 dataset has been used which has 1934 different types o drugs and 230,887 interactions of drugs. For comparing network base features of the model with other derived drug features using various drugs also used chemical structures and "ATC" codes from drug bank data. The DPDDI has three phases i) Using GCN model to extract "low dimension embedding latent features" from the network of DDIs; ii) For representing the pairs of drugs the latent feature vectors (i.e Zi and Zj) of drug di and dj is aggregated; iii) To predict the DDIs the fused feature vector is fed to DNN. In the first step of the model "DPDDI Framework" framework, a two-layer GCN is built for obtaining latent drug features which will capture the complicated relationship between nodes of drugs in the DDIs network. Then in the next step the corresponding "latent feature of drugs" are concatenated and every pair of a drug is presented as a feature vector. In the last the feature vectors which are generated in the second step fed into DNN for training the predictor and predicting the possible DDIs. Many computational methods for DDIs predictions are depending heavily on the various drug-related features but for datasets of large scale the important features of most drugs are unavailable and another problem is robustness which can lead the model to be sensitive for the test dataset in terms of information pairwise similarity. To overcome these two problems the researchers [12] used (Graph representation learning) for accurate DDIs predictions. In a large dataset sometimes many features of drugs are not available which can affect the results of the model for avoiding this problem the DDIs prediction was done by the related drug information which is easily accessible. The model is taking input (SMILES) inform of canonical representation where the SMILES canonical representation is a molecule linear notation. RDKit is used for converting SMILES pairwise representation into a graph of molecular pair. And using RDKit we can take out an atom list and a matrix of multi-channel adjacency structural information from a molecular graph and in the next step, this information was fed to (Siamese GCN) to predict the interaction of drugs. This research is an escalation of a preceding publication and was a data build up[13]. Take 1923 Drug Target Interactions from benchmark data sets and formulate a new data set which involves 708 drugs from DrugBank 5.0 data set and 1412 targets from HPRD9.0. For calculating the similarities between drugs used textbfTanimoto coefficient based on the respective drugs chemical structures and for calculating the similarities between drugs used textbfSmith-Waterman Score which uses primary sequences for calculations. A drug target heterogeneous network is constructed and extracts the nodes and edge information of the network for fully understanding that how numerous aspects come into play for the DTIs predictions through the analysis of drug-target attributes. For the feature representation learning of each node, an adversarial graph encoder is used and a LightGBM-based classifier was used for the measuring DTI interactions score. The model was divided into 3 network parts the first network is for drug-drug interaction, the second network for target-target interaction, and the third for drug-target interaction. True False Positive, False Positive Rates, and True Positive Rates were used for calculating the model performance. For the consideration of data imbalance, the Area Under Curve and Precision-Recall Curve were used. In GAN l=50, k=200 was set for Weight Matrix settings, and the number of samples was set to 900. The researchers use the heterogeneous graph of drug-target nodes for drug-target interaction prediction by using the deep learning techniques and for showing the entire interaction heterogeneous graph the encoding and decoding techniques are used[14]. The model can be summarized as a multi-label link prediction algorithm. The model has a 5 interaction network and two similarities networks like drug-protein, drug-drug, protein-protein similarity drug protein-side effect, drug-disease, drug-drug similarity, and protein-protein. At least one of the relations will be contained in each edge of the graph network. Based on node neighborhood topology each node interacts with other network nodes. Normalization constants and ReLU activation function are used for a trainable matrix concerning the neighborhood nodes. By stacking the previous layer output which becomes an input to the next layer the encoders are created along with the activation functions. The reconstructed edge labels are used by the decoders. The sigmoid function is used for calculating the probability and for the training optimization the cross-entropy loss function is used. The data-set used in this research has been extracted from Drug Bank Database version3.0 for the Drug-Protein and Drug-Drug interactions and the protein-protein data has been extracted from the HPRD database. Drug-Protein and Drug-Disease side effects were taken from the Comparative SIDER and Toxicogenomics database. For the comparison, the AUROC method is used and for the estimation of performance, the ten-fold cross-validation was used in the form of AUPR and AUROC.

2.5 Neural Networks

Neural Networks also called Artificial Neural Networks are a set of algorithms that are inspired by the human brain which are developed to recognize patterns[15]. The base of neural networks is on the collection of connected nodes or unites which are called artificial neurons. Each node or artificial neuron receives the signal and then processes that signal and then signals the connected node or neuron. The signals are the real numbers and for the computation of the outputs some nonlinear function like ReLU, Softmax is used. We can also be called edges to connections. For the adjustment of learning processes, the edges and neurons have weights. The weights are used for the increase and decrease of the strength of the signal at the connection. This study[16] aimed to provide a computational model for unknown drug-drug interaction prediction using numerous data about drugs. The authors proposed a model named "NDD neural network-based drug-drug interaction model". The similarity of drugs depends on the target, pathways, side effects, the substructure of a drug, off-label side effects, and indication data. To attain high-level features the "NDD" model first uses a selection process of "heuristic similarity" and then after getting the similarities to combine it with the "non-linear similarity fusion method" then use "neural network" for the prediction of the interactions. The model is working in four steps which are first for every pair of drugs the drug similarity and "GIP Gaussian Interaction profile" is calculated in the second step select the subset of the similarities which have more details and less redundancy in the third step matrix of integrated similarity is obtained by integrating the similarities which have been selected which results in one matrix of all the details in the last step the matrix is fed to the neural network for the interaction prediction. There are four data sets used in this article which are "drug bank", "SIDER", "KEGG", and "PubChem". Researchers proposed a model which uses a hybrid of neural network architectures being Fully Connected Neural Network and Convolutional Neural Network[17]. The proposed model predicts both negative and positive interactions between related targets and drugs through the use of chemical structures of drugs called SMILES and the protein amino acid sequences. Used four different types of data-sets named DrugBank, BioLip, BindingDB, and Yamanishi et al's and compile a new dataset. The SMILES and protein sequences are encoded into integer values according to the encoding schema. For the normalization of the categorical values, one hot encoding is used which also converts encoded sequences into binary vectors. Before applying the max-pooling patterns are found through SMILES and protein sequences and then a single feature vector is produced. And then the produced feature vector is fed to (FCNN) and for overcoming the over-fitting drop-out is applied and in the last, the output layer returns binary values which specify either it is a positive or negative interaction. The proposed model is then compared with other deep neural and machine learning techniques like Support Vector Machine(SVM), Fully Connected Neural Network(FCNN), etc. The parameters used for comparative analysis are F1-Score, Accuracy, Sensitivity, Specificity, Area Under Precision-Recall Curve. Researchers proposed a method named DeepDDI[18] and developed a neural network based method for the prediction of 86 types DDIs. The DDIs are collected from Drug Bank database and the deep learning model based on the chemical substructure.

2.6 Autoencoders

Autoencoders are the type of artificial neural network which are used for learning the codings of data efficiently in an unsupervised manner[19]. In the autoencoders, the inputs are the same as outputs. The inputs are compressed into a lower-dimensional code and then from that representation reconstruct the output. There are three main components of autoencoder which are encoder, code, and decoder. The encoder part compresses the inputs and produces a code and the decoder part uses that code and reconstructs the output from that code and where the codes are the "Compression" or "Summary" of the given inputs. For secure and extra efficacious co-prescription of drugs, it is very important to predict the effects of drug-drug interactions. As we know that for many diseases like AIDS, cancer, and asthma therapy of multi-drug is becoming very promising because multi-drug therapy can increase the efficacy of the drug, decrease the toxicity of the drug, or can reduce the resistance of the drug. Also, multi-drug therapy can cause drug-drug interaction which can lead to adverse drug events. Thus the accurate prediction of the drug-drug Interaction effect is very important. The objective of this research paper[20] is to improve the accuracy of classification of previous studies a new deep learning model was proposed by the authors which are using the known functions and the supplementary features of the target genes. Instead of using “TSP”, “GSP”, and “SSP” combined they constructed separately three structure similarity profiles because the input size of the above three combined profiles is very large and used autoencoders for reducing the features. After reducing the feature dimensions the reduced pairs are concatenated and fed to a feed-forward deep neural network to predict DDI type. In this study[21], the authors initiate the adversarial autoencoders with knowledge graphs (KG) embeddings to develop a new framework for the DDIs prediction the base of the autoencoders is on (Gumbel-Softmax relaxation) and (Wasserstein distances). The purpose of developing a new (KG) embedding framework is that the existing methods of (KG) for the prediction of DDIs is generating the negative samples using the uniformly random mode, as a result, training an effective model these samples are very oversimple. The autoencoder used in the proposed method is employed for generating a high-quality negative sample and the autoencoder hidden vector is considered a convincing candidate for the drug. After generating the negative samples these samples and the positive samples are fed to the discriminator to improve the performance of the embedding model. For solving the vanishing gradient problem (Wasserstein distance) and the (Gumbel-Softmax relaxation) are used to train the model. There are two different datasets are used named (DeepDDI) which is generated from a drug bank database in this dataset there are 1,710 drugs, 86 various interactions, and 192,284 pairs of drugs and the other dataset name is (Decagon) the dataset has 637 drugs, 200 interactions and 1,121,808 pairs of drugs. Existing Methods for DDI prediction is mostly distributed into 3 groups which are “knowledge-based, similarity-based and, network-based” newly studies have exhibited that for high accuracy development heterogeneous drug features integrations are very important but it also comes up with many more challenges like incomplete data, nonlinear relations, and heterogeneous properties. So the authors proposed a method named “DDI-MADE” [22] for the prediction of DDIs. Multi-model deep autoencoders to learn from a network of various drug features simultaneously a unified drug representation. Then for the representation of drug pairs on the learned drug embeddings, various operators are adopted, and then for the DDIs predictions used the “random forest” to train the model. The method has a shared hidden layer and various channels of auto-encoders after the process of training in the hidden layer the “low dimensional space” is we want to obtain. The dataset used in this study is drug bank version 5.1.0 which is characterized in 314 pathways, 285 enzymes 699 chemical substructures, and 2367.

2.7 Convolutional Neural Network (CNN)

CNN or(ConvNet) is a type of deep neural network which is specialized for the recognition of image[23]. ConvNet was developed in the mid of 1980s[24]. But for a while, the ConvNet was forgotten because for the real-world application it was impractical with complicated images. After 2012 when they are revived most of the computer vision fields were conquered by ConvNet and are growing at a rapid pace[25]. The ConvNet consists of three types of layers first one is input layers, the second is hidden layers and the last is output layers. The hidden layer in the convolutional neural network includes layers that perform convolutions. Commonly in the hidden layer of ConvNet includes a layer that performs convolution kernel dot product with the input matrix layer’s commonly used activation function is ReLU[25] and usually the product is Frobenius inner product. Features maps are generated by the convolutional operations which act as an input to the next layer and then followed by other layers like normalization layers, fully connected layers, and pooling layers. The purpose of the research article[26] was to predict accurately drug interactions. Earlier the prediction of drug interactions was interpreted ambiguously because in small quantities of drugs the tests were performed in person. Many methods of machine learning have recently been proposed to predict DDI but over-fitting is a major problem with these models. So in accurately predicting DDI, these models did not work very well. To manage the over-fitting issues the authors have proposed an "integrated convolutional mixture density recurrent neural network" in the proposed model CNN, RNN, and MDN are integrated. The proposed model working is described as follows. For capturing the high dimensional input features using 1D CNNs. Next, use the LSTM-RNN for time series data modeling. The output layer consists of a mix of Gaussian densities to improve the accuracy of the forecast. (NCI-ALMANAC)this data set is constructed by collecting synergies of drugs over 3 institutions “NCI’s Frederick National Lab, SRI International, and Cancer Study” data set contain 29000 features. This study[27] intended to develop a method that can predict DDIs from multiple data sources. As we know that many approaches of data-driven DDIs prediction depend on the single data source for improving the prediction of DDIs it is important to use multiple sources of data. Different machine learning techniques are used but these techniques are ineffectual in terms of dealing with the skew in data. Predicting DDIs from multiple sources of data the authors present a new approach of "Machine Learning". There are three different datasets named "Drug-bank", "PharmGKB", and "KEGG" which have 12,000 drug features these features of drugs are integrated using Knowledge graphs. For training, the prediction model different approaches of embedding are used for embedding the nodes in the graph. While using different embedding approaches from the results it is showing that the better method for embedding is complex embedding which is created using "PyTorch-BigGraph" with the combination of prediction models of classical machine learning and "Convolutional-LSTM".

2.8 Embedding Approach

An embedding is a relatively low dimensional space into which high dimensional vectors can be translated. Three types of embeddings are used in the below research which are SMILES strings drug structure embeddings, relational data embeddings, and image embedding of drug structures. Researchers utilize multiple knowledge sources[28] for using rich drugs representation to propose an effective approach for the prediction of DDI. For learning an embedding of drugs the Drug-Target Interaction Network has been used by using the metapath2vec algorithm and the Variational Auto Encoder has been used to gain the representation of drugs from rich chemical structure representation of drugs. The problem of DDI prediction is modeled as a link prediction problem in the network of DDI which contains known interactions of drugs. The nodes in the DDI network are represented as their embeddings. A link prediction algorithm which based on Graph Auto-Encoders for the prediction of additional edges in the network, which are the potential interactions. The proposed method is evaluated with the three DDI data-sets named BioSnap, SemMedDB, and DrugBank and the experimental results show that the proposed method outperforms the earlier method in terms of (AUC, AUPR, and F1-score) on the above three data-sets and achieved an AUC of 0.73 on DrugBank data-set, 0.76 on SemMedDB data-set, and 0.76 on BioSNAP data-set. Researchers use different types of data sets for the prediction of DDI [29] the data used in this study are in form of images and strings. They have used three types of data the drug structure images, string of chemical substructures (SMILES), and relational representation of different associations between the protein and drugs like (target, enzymes, and transporter). For learning a similarity metric from drug structure image embeddings used a Siamese architecture in which the inputs are mapped into a target space the distance between the mappings is minimized for the similar pairs of example in the target space and for the dissimilar examples the distance is maximized. The siamese architecture is consists of two identical sub-networks and each sub-network takes an image of size 500x500x1 of gray-scale as input and the network consist of 4 convolutional layers with the filter of 64,128,128, and 265 respectively. For each convolutional layer, the kernel size is 9 x 9 and the relu is used as an activation function. For the embeddings of the relational data, the DDIs can be considered as the characterization of the relationships between the different proteins and drugs like transporters, and enzymes using different features like metabolism, excretion, distribution, absorption, etc. For the drug, structure Smiles strings embeddings used SMILESVec model in which the strings of SMILES divide into various interacting sub-structures after that another method named word2vec is used for generating the embeddings for these sub-structures and then all of the above embeddings are then combined to generate the drugs final embeddings. After generating all the three different types of embedding then aggregate the embedding for generating the lower-level representation. For both cases SMILES string and image embeddings use a size of 100 x 1. The proposed method achieved an accuracy of 0.884.

2.9 Knowledge Graph and Graph Neural Networks (KG and GNN)

The knowledge graph is used to organize multiple source data, capture information in a given domain about entities of interest for example (places, people, and places), and produce the connection between them[30]. Knowledge graph neural networks are connection models which pass the messages between the graph nodes for capturing the dependency of the graph. Unlike standard neural networks, graph neural networks retain a state that can represent information from its neighborhood with arbitrary depth. The Researchers propose an end-to-end framework called "Knowledge Graph Neural Network" [31] to resolve the issue of the previous studies in which smaller attention is paid to the potential correlation between other entities and drugs like genes and targets and the (KG) adopted for the recent studies in the prediction of DDIs learn node latent embedding directly but obtaining the rich neighborhood in information’s they are limited the proposed method can capture the drug and its potential neighborhoods effectively by mining their associated relations in KG. For extracting both semantic relations and high-order structures of the KG, for each entity in KG, the model learn from neighborhoods as their local receptive and then neighborhood information is integrated with the bias from the current entity representation using this method the drugs potential long-distance correlations and the receptive field can be naturally extended to various leap away to model high-order topological information. Researchers developed a multi-scale feature fusion deep learning framework named MUFFIN[32] the framework can learn jointly the representation of drug based on both the knowledge graph and the drug-self structure information with rich biomedical information. In the framework, a bi-level cross strategy is designed which includes scalar and cross-level components to fuse multi-modal features well and the restriction of limited labeled data can alleviate deep learning models by crossing the features learned from drug molecular graph and the large scale KG where the cross-level is the aggregation and extraction of global features and local features by operating cross product for various features and the scalar level using the element-wise product to extract many fine-grained fusion features. The proposed framework includes three modules the first one is the representation learning module the message passing neural network and the knowledge graph representation method are used to the semantic features and the molecular structure feature from knowledge graph and molecular graph, accordingly. The second module is a feature fusion module in bi-level strategy is designed which includes two units named cross-level and scalar-level. The third module is named the classifier module in which the features learned from the above modules are concatenated and then for the prediction of DDIs used different classifiers according to the different classification tasks.

2.10 Convolutional LSTM's

Convolutional LSTM's are typically used for the prediction of Spatio-temporal which has a convolutional structure in both transitions of state-to-state and input-to-state. The Researchers introduce a novel deep learning framework named Deep Heterogeneous Drug Target Affinity for Drug Target Interactions[33]. The framework is divided into three parts the first part uses Dense Net augmented with SE operation for learning the protein structures, the second part uses a heterogeneous graph network for learning the topological representation of drug molecules, and the third part uses bidirectional Convolutional Long Short-Term Memory(ConvLSTM) architecture to learn the sequential SMILES characteristics of input compounds. Then for calculating the affinity scores out of the ConvvLSTM is used. For encoding the amino acid sequences the Polypeptide frequency of Word Frequency is used which provides features of the protein. Attaining the polypeptide frequency through the model portion is inspired by DenseNet which has three blocks of dense convolutional where each block is carrying the sum of information from its forerunner block. For channel exciting and squeezing the Squeezing and Execute block is used which combined regional average pooling for creating the Global Average Pooling to create channel descriptors for the entire input channel and for the three-channel model a features space encoding feature vectors to 1 x 1 x F is created by using the sigmoid function and for capturing the channel's nonlinear interaction the excitation process is used. The features of Drugs are learned through the SMILES conversion into a graphical representation used RDKit. For the preserving of all meta-path information, the Heterogeneous graph Attention Network is used instead of a simple Graph Attention Network because simple GAN only focuses on nodes but the nodes of a heterogeneous network are linked from various similarities through abundant paths. For achieving the GANs the weights of the neighboring nodes are learned per meta-path and then calculated the meta-path differences for fetching the ideal weight combinations. The semantic path values are then concatenated because the variance increases when each semantic path produces its own path values. For converting the Smiles to vector notations the Smi2vec method is used then the matrix of converted smiles is fed to bi-directional ConvLSTM. The output layer of the model has three fully Connected Layers with 512,768 and 1024 nodes. For avoiding the over-fitting dropout layer is used and used a Mean Squared Error for loss. This research has been done for fasting the development of a suitable vaccine for CoVid-19. The framework achieved the highest Concordance Index of 0.924 and 0.927 and Mean Square Error of 0.195 and 0.11 on two different data sets called Davis and Kiba.

2.10.1 Summary of Literature

Existing work mostly focuses on the interaction of two drugs, or some of the studies like [8] predict the interaction of specific diseases drugs or some of the studies only used one feature drug like the researches of [18] uses only one feature of drug which is chemical substructures for the prediction of 86 types of DDIs events. The authors of DDIMDL[7] used four features of drugs and used a deep neural network based on sub-models for every feature of the drug for the prediction of DDI associated events and the results show that using multiple features of drugs and using sub-models perform better in the prediction of DDI associated events. We have also used four features of drugs same as DDIMDL and instead of using a simple deep neural network we have used three deep learning methods which are CNN, LSTM, and RNN based on sub-models and our three models perform better than above mentioned work in terms of accuracy, precision, recall, AUC, and AUPR.

References	Problem Statement	Techniques and Algorithms	Results
[26]	To overcome over-fitting and efficiently predict drug-drug interaction using deep learning	(CNN), (RNN), (MDN), (LSTM-RNN)	ACC of 98.4%
[11]	drug-drug interaction prediction without using the various drug-related properties	(GCN), (DNN), (5-fold CV)	ACC of 94.0%
[20]	For decreasing the size of every single profile autoencoders perform well rather than using (PCA) and the prediction accuracy is increasing whenever used just (SSP).	(Autoencoders), (FeedForward DNN)	ACC of 94.0%
[16]	Combining similarity selections and similarity integration with a neural network to built accurate DIIs predictor.	(Neural networks), (GIP), (SNF), (5-fold CV)	AUC 99.2%
[22]	For the prediction of DDIs how to use multiple drug datasets and how the multiple structures of the nonlinear network can be captured.	(DNN), (Deep Autoencoders), (3, 5-fold CV)	ACC of 99.5%
[27]	From three different data sources, 12,000 features of drugs are used which is integrated using knowledge graphs, and a complex embedding method was created	(Convolutional-LSTM), (Knowledge Graphs), (5-fold CV)	(AUPR 93.0%)
[6]	From drug bank, dataset generated features of drugs using Smiles and used the generated features in DNN to predict 80 different DDIs for a pair of drugs.	(DNN), (SVM)	ACC 94.0%
[7]	A joint framework of DNN is used for prediction of DDIs	(DNN), (StandfordNLP), (Similarity Matrix), (5-fold CV)	ACC of 88.5%

Table 2.1: Literature Summary part 1.0

References	Problem Statement	Techniques and Algorithms	Results
[9]	Using (one-class support vector machine) and (KNN) to generate the negative seeds from unlabeled data and uses in classifier for better DDIs prediction.	(SVM), (OCSVM), (KNN), (3, 5-fold CV)	F1-score 86.2%
[12]	GCN was used to solve the problem of robustness and model sensitivity to a test dataset interm of pairwise information similarity by using drug-related information instead of various drug features.	(Siamese GCN), (Single Layer NN),	AUC of 80.0% on large scale dataset and on small scale dataset AUC of 94.02%
[10]	Combining vector-based and similarity-based models of ML with different training datasets and set of similarities which are well defined to check the DDIs prediction .	(SVM), (10-fold CV)	AUROC of 98.0%
[21]	Autoencoders was used with (KG) embedding for generating the high-quality negative samples for improving the (KG) embedding for DDIs prediction.	(Autoencoder), (KG embedding), (Wasserstein distance), (Gumbel-Softmax relaxation)	PR-AUC 76.0%

Table 2.2: Literature Summary part 1.1

Chapter 3

System Architecture and Research Design

This Chapter has two parts the first part contains the details about the data-set used in the study and the second part contains the proposed methodology for this study.

3.1 Data Set

The data-set used in this study was named Drug-Bank. Drug-Bank online is a broad, free available, online database that contains drugs and drugs target information both cheminformatics and bioinformatics resources. In a drug bank, the researchers combine drug details for example (pharmaceutical, chemical, and pharmacological) data with extensive drug target for example (structure, sequence, and pathway) information. Drug-Bank online database is used widely by the students, physicians, medicinal chemists drug industry because of its wide scope, detailed description of data, and broad referencing. Drug-Bank database contains 12,151 drugs and its broad information which includes the drug name, chemical substructure (SMILES) or we can say the chemical formula of drugs, targets, enzyme, pathways, description, protein, etc, also contains 3844 drugs approved from FDA and 5867 experimental drugs. we will be using different features of drugs like name enzyme, chemical substructure, target, and pathways, and the DDIs of the drugs. The target and enzyme of the corresponding drug can be obtained directly from the database while for the interaction of drugs we will use the description of drugs which is in form of a sentence for example "the process of metabolism can be increased when using DrugA and DrugB". We will be using the drugs which have all the above four features and have a minimum of 10 or more interactions with other drugs. In the first step, we will process the data which include extraction of drugs features (enzyme, targets, chemical substructure, and pathways) then from the description extracting DDIs of the drugs and selecting drugs that have interactions with other drugs in the second step we will be preparing data in terms of training data-set and testing data-set.

3.1.1 History of Drug-Bank

The Drug-Bank project is started in 2006 in the lab of Dr. David Wishart's at Alberta University. This project was begun for academic researchers to help them to get detailed drug structure information. It became a part of "The Metabolomics Innovation Center" (TMIC) in 2011. Then in 2015, the project was draw-out into "OMx Personal Health Analytics In" because of the continued growth in scope and popularity. Below is the complete history of Drug-Bank versions (1.0, 2.0, 3.0, and 4.0).

Category	v-1.0	v-2.0	v-3.0	v-4.0
No. of data fields per DrugCard	88	108	148	208
No. of search types	8	12	16	18
No. of illustrated drug-action pathways	0	0	168	232
No. of drugs with metabolizing enzyme data	0	0	762	1,037
No. of drug metabolites with structures	0	0	0	1,239
No. of drug-metabolism reactions	0	0	0	1,308
No. of illustrated drug metabolism pathways	0	0	0	53
No. of drugs with drug transporter data	0	0	516	623
No. of drugs with taxonomic classification information	0	0	0	6,713
No. of SNP-associated drug effects	0	0	113	201
No. of drugs with patent/pricing/manufacturer data	0	0	1,208	1,450
No. of food–drug interactions	0	714	1,039	1,180
No. of drug–drug interactions	0	13,242	13,795	14,150
No. of ADMET parameters (Caco-2, LogS)	0	276	890	6,667
No. of QSAR parameters per drug	5	6	14	23
No. of drugs with drug-target binding constant data	0	0	0	791
No. of drugs with NMR spectra	0	0	0	306
No. of drugs with MS spectra	0	0	0	384
No. of drugs with chemical synthesis information	0	38	38	1,285
No. of FDA-approved small molecule drugs	841	1,344	1,424	1,558
No. of biotech drugs	113	123	132	155
No. of nutraceutical drugs	61	69	82	87
No. of withdrawn drugs	0	57	68	78
No. of illicit drugs	0	188	189	190
No. of experimental drugs	2,894	3,116	5,210	6,009
Total No. of experimental and FDA small molecule drugs	3,796	4,774	6,684	7,561
Total No. of experimental and FDA drugs (all types)	3,909	4,897	6,816	7,713
No. of all drug targets (unique)	2,133	3,037	4,326	4,115
No. of approved-drug enzymes/carriers (unique)	0	0	164	245
No. of all drug enzymes/carriers (unique)	0	0	169	253
No. of external database links	12	18	31	33

Table 3.1: Drug bank history

* This table data is taken from DrugBank Wikipedia page [34]

3.2 Proposed Methodology

In this part of the chapter, we will go over three different models and feature extractions techniques which are used for the prediction of DDI events on the Drug-Bank Database and will describe the architecture of the models.

3.2.1 Features Extraction

The first step in the proposed methodology is to extract the drug features vectors using drugs data chemical substructure, enzymes, targets, and pathways. For getting the binary features vector first we adopt encoding (0,1) where the values (1,0) of the features represent the presence and absence of the components. For the above four features of drugs the PubChem define different types of dimensional bit vectors for example there are 881 chemical substructure types, 202 enzymes types, 1162 targets types, and 957 pathways types here we will take an enzyme for an example the enzymes drug feature can be represented a 202 dimensional bit vector where the values are only 0 and 1 In which 1 represent the presence of enzymes and the 0 represents the absence of the types of enzymes. Then in the next, we calculate the similarity metrics of drugs features which are chemical substructure, enzymes, targets, and pathways by Jaccard similarity matrix. According to measures of drug-drug similarity between two drugs the similarity bit feature vectors, we measure. So we can get four similarity matrices of 572 x 572 and then we can represent each matrix in form of S= $\left(\text{Sij} \right)$ where the value of Sij is in between 0 and 1 also include the 0 and 1 so the higher the value the higher the two drugs similarity. In the second step, we fed the extracted features to model for the classification.

3.2.2 1-D CNN as a Model

Convolutional Neural Network or(ConvNet) is a type of deep neural network which is specialized for the recognition of image[23]. ConvNet was developed in the mid of 1980s[24]. But for a while, the ConvNet was forgotten because for the real-world application it was impractical with complicated images. After 2012 when they are revived most of the computer vision fields were conquered by ConvNet and are growing at a rapid pace[25]. During the past decade for different machine learning and computer vision operations, the convolutional neural network has become a de-facto standard. ConvNet is the feed-forward networks with subsampling layers and alternating convolutional. A 2D deep CNNs with millions numbers of parameters and with a numerous number of the hidden layer have the capability to learn complex objects and patterns providing that can be trained on a huge size visual database. This distinctive ability makes these CNN's the main tool for different engineering applications for 2D signals like videos frames and images. But this may not be a valid option in different applications like 1D signals mostly when training data is scanty or application-specific. To address the above issue 1D CNNs have been proposed which instantly achieved a very good performance in different types of application like classification of personalized biomedical data, structural health monitoring, and early diagnosis and another huge advantage of 1D CNNs are that perform 1D Convolutions (additions and scalar multiplication) through which we can achieve the desire results on real-time and low-cost hardware[30].

3.2.2.1 Our 1D CNN Architecture

In the first step we give four features of drugs like (chemical substructures, enzymes, pathways, and targets) as an input for the drug-drug similarities calculations using the similarities measures which are explained above in 3.2.1 part then uses that similarities matrix as a representation of drugs and then that representation of drugs are respectively fed into a sub-model based on 1D-Convolutional neural network. Then in the last step, we combine the sub-model for learning the cross-modality representation of drug pairs and predict events of DDI with cross-modality representation.

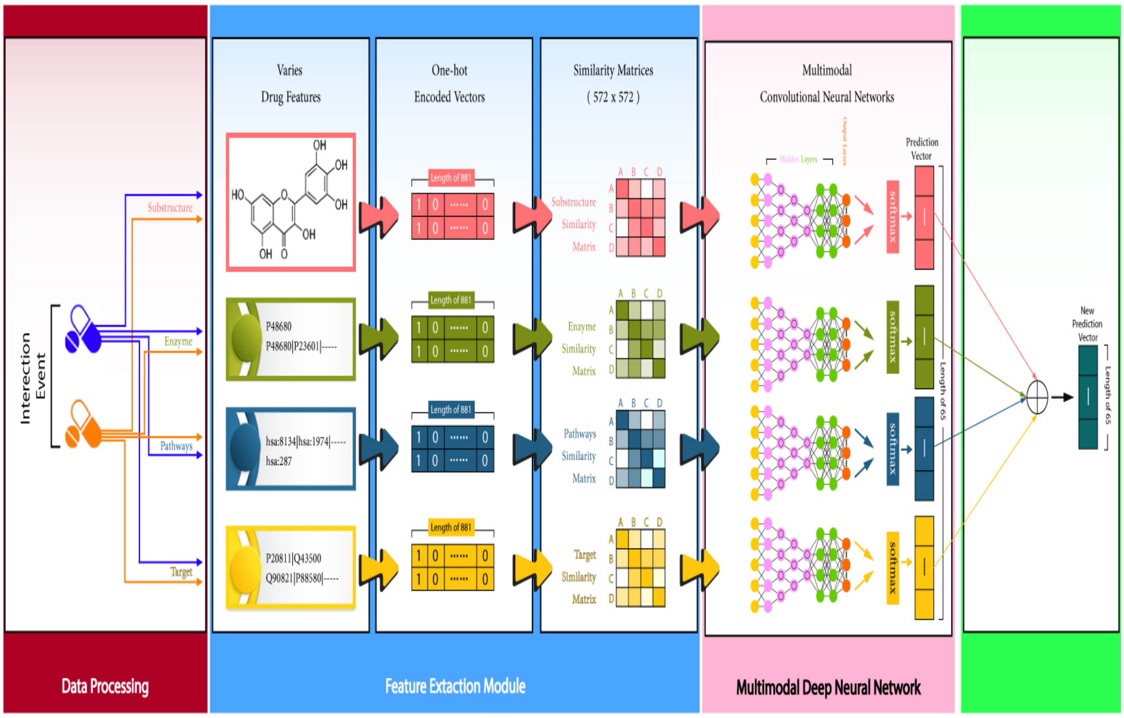


Figure 3.1: Proposed method 1 pipeline

3.3 Recurrent Neural Network as a Model

Recurrent Neural Networks are types of Artificial Neural Network where the previous step output is fed to the current step as an input[35] wherein the traditional neural network the inputs and the outputs are independent of each other but in the scenario where we want to predict the next sentence or the word, the previous record of the data is required so it is compulsory to remember the previous record of the data thus with the traditional approaches we can't achieve this to overcome this problem the RNN came into existence which uses the hidden layer to solve the above problem. The most important feature of the RNN is the Hidden state which remembers the previous information about the sequence. RNN has a "memory" that keeps the information which has been calculated previously.

3.3.0.1 RNN working structure

For Understanding the working structure of RNN we will take an example below, Let's suppose we have a deeper network with 1 input layer, 3 hidden layers, and 1 output layer so, like other artificial neural networks each and every hidden layer of the network will have its own set of biases and weights, lets suppose we represent the first hidden layer weights and biases with (w1,b1), the second hidden layer weights and biases with (w2,b2) and for the third hidden layer weights and biases with (w3,b3) which means that these layers are not dependent on each other for example these layers do not remember the previous output.

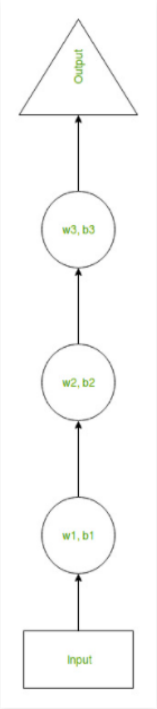


Figure 3.2: Feed DNN with 3 hidden and one input and output layers

Now RNN will apply the following steps. 1) In the first step, RNN will convert the independent activation’s to dependent activation’s by providing all the layers with the same biases and weights, thus lowering the complexity of increasing parameters and remembering every previous output by feeding every output an input to the next hidden layer. 2) In the second step the three layers can be connect with each other like that the bias and weights of hidden layers is the same, into s single recurrent layer.

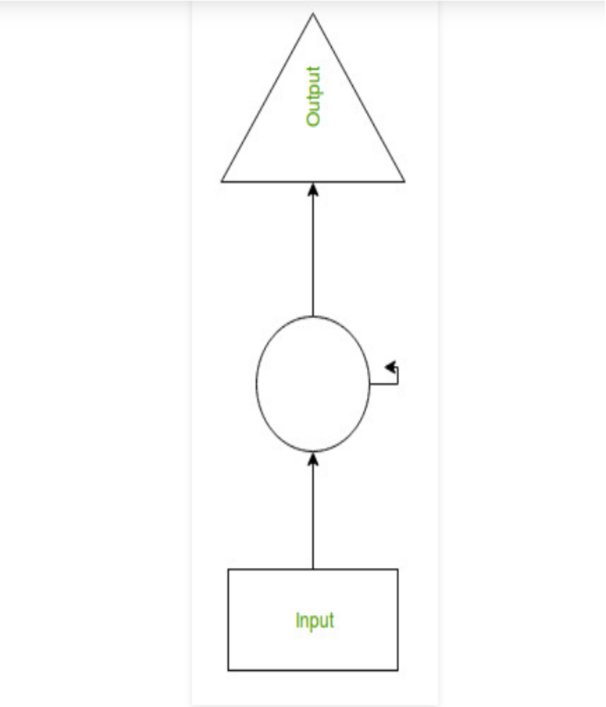


Figure 3.3: Simple RNN stucture

Current state calculating formula $\mathrm{h}_t = f\left(h_{t-1}, X_t\right)$
$Where\ h_t\ stand\ for\ current\ state,\ h_{t-1}\ stands\ for\ previous\ state\ and$
$X_t\ stands\ for\ input\ state$

Tanh activation function formula
$h_t = tanh\left(W_{hh}h_{t-1} + W_{xh}X_t\right)$
$Where\ W_{hh}\ stand\ for\ weight\ at\ recurrent\ neuron\ and\ W_{xh}\ is\ weight\ at\ input\ neuron$

Output calculation formula$Y_t = W_{hy}ht$
$Y_t\ stands\ for\ output\ and\ W_{hy}\ is\ weight\ at\ output\ layer$

3.3.1 Our RNN Model Architecture

In the first step we give four features of drugs like (chemical substructures, enzymes, pathways, and targets) as an input for the drug-drug similarities calculations using the similarities measures which are explained above in 3.2.1 part then uses that similarities matrix as a representation of drugs and then that representation of drugs are respectively fed into a sub-model based on Recurrent neural network. Then in the last step, we combine the sub-model for learning the cross-modality representation of drug pairs and predict events of DDI with cross-modality representation.

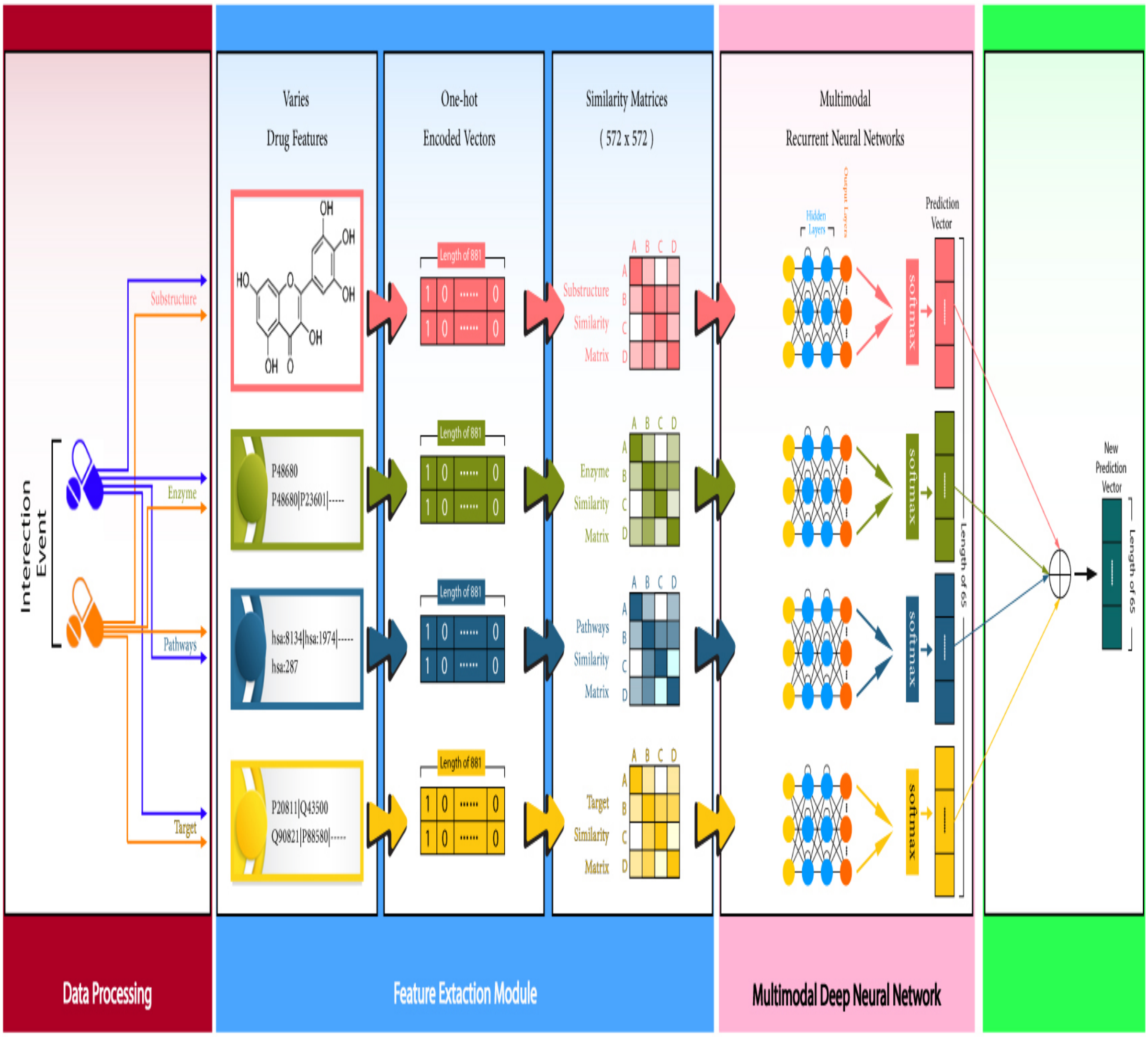


Figure 3.4: Proposed method 2 pipeline

3.4 Long Short Term Memory (LSTM’s)

Long Short Term Memory is a type of recurrent neural network in the RNN the output of the last step is fed as an input to the current step. The Lstm was developed by Hochreiter Schmidhuber the main agenda behind the LSTM’s was to overcome the RNN problem of long-term dependencies in which the RNN can’t predict the words which are stored in long-term memory but the recent information can predict accurately.

3.4.0.1 LSTM Structure

LSTM's has the structure of chain type which contains cells and four neural networks. Cells are blocks of different memory. Cells are used to retain the information's and the gates are used for memory manipulations. There are three types of gates which are discussed below.

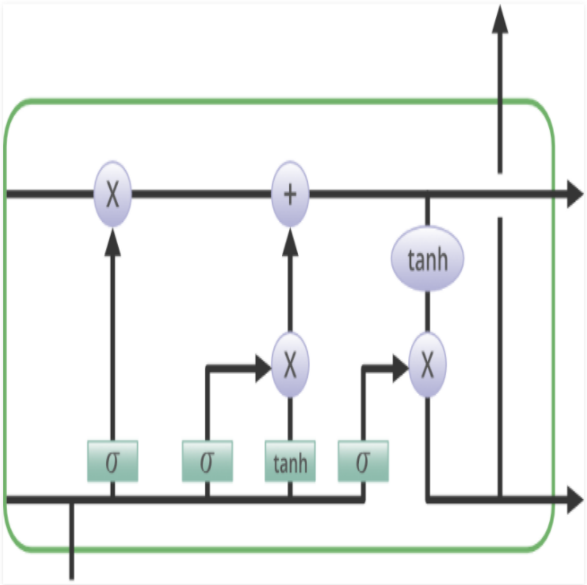


Figure 3.5: LSTM's structure

1) Forget Gate: The forget gate is used for removing the information which is no longer useful in the cell state. Two inputs are fed to the gate and then multiply these inputs with the weight matrix which is followed by the bias addition.

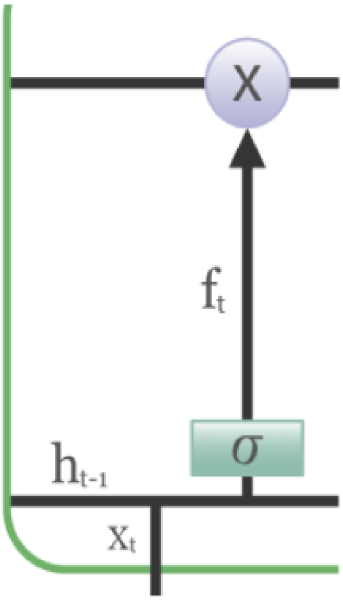


Figure 3.6: LSTM's forget gate structure

2) Input gate: The input gate is used for the useful information addition to the cell state. First used the sigmoid function for the information regulation and then for the remembrance of the value filter the values like forget gate. Then used a tanh function for vector creation which gives an output value of -1 to +1.

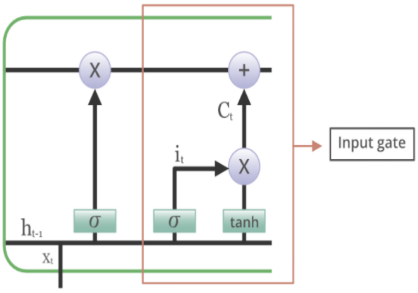


Figure 3.7: LSTM's input gate structure

3) Output gate: The output gate is used for useful information extraction from the current cell state. First, apply tanh function on the cell to generate a vector then use the sigmoid function to regulate the information and then filter the values to be remembered.

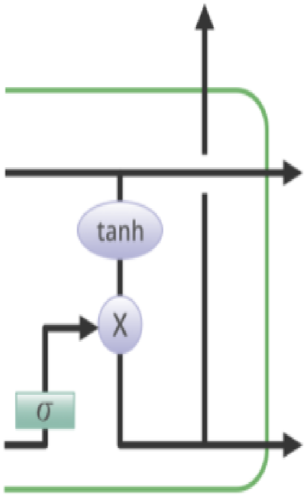


Figure 3.8: LSTM output gate structure

3.4.1 Our LSTM's Model Architecture

In the first step we give four features of drugs like (chemical substructures, enzymes, pathways, and targets) as an input for the drug-drug similarities calculations using the similarities measures which are explained above in 3.2.1 part then uses that similarities matrix as a representation of drugs and then that representation of drugs are respectively fed into a sub-model based on Long Short Term Memory. Then in the last step, we combine the sub-model for learning the cross-modality representation of drug pairs and predict events of DDI with cross-modality representation.

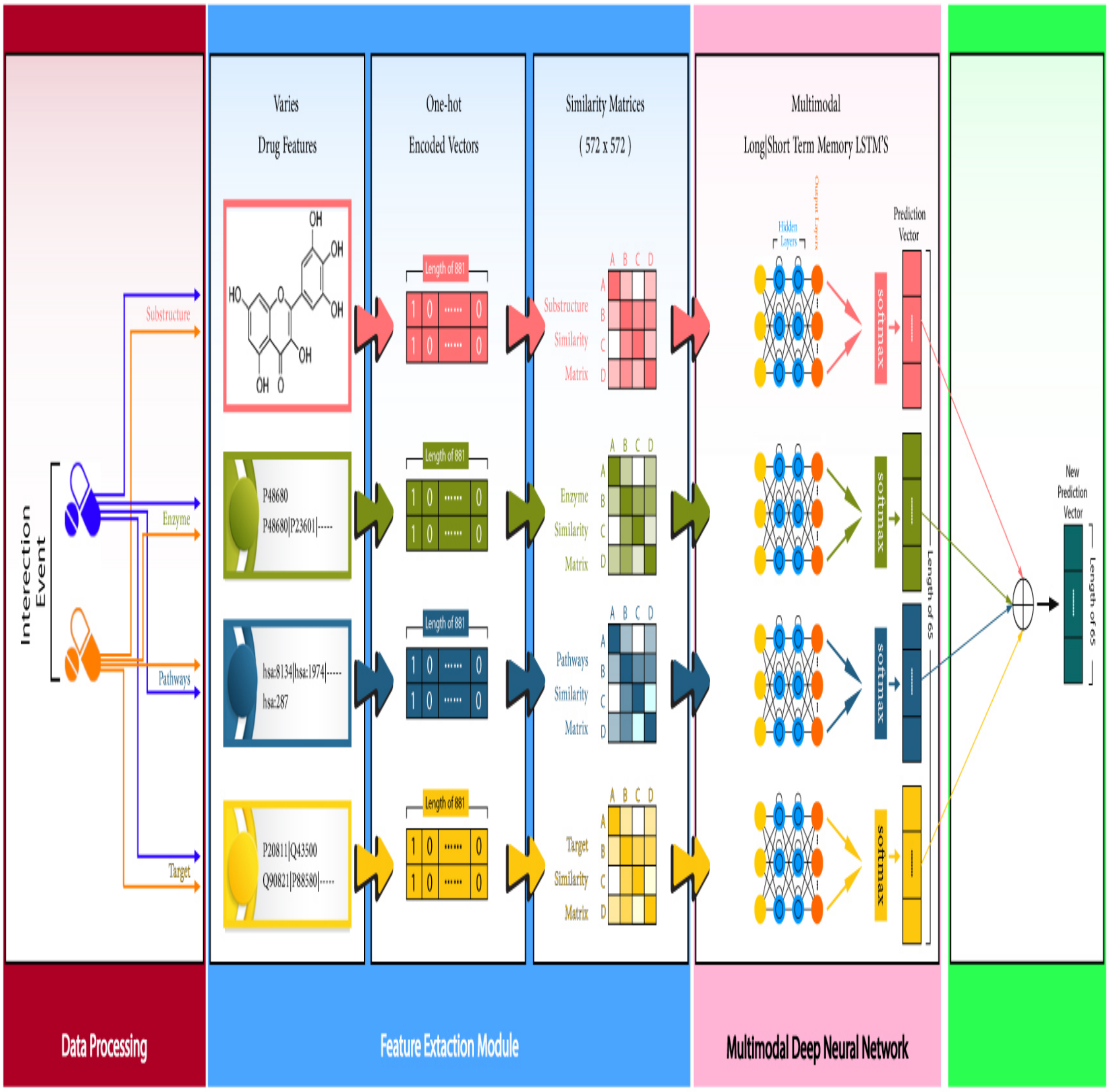


Figure 3.9: Proposed method 3 pipeline

3.5 Cross-Entropy

Cross-entropy is widely used in deep learning and machine learning models as a loss function. Cross-entropy is a measure from the field of information technology that calculates (p,q) two distributions of probability it is constructed upon entropy[36]. It is similar to KLdivergence where KLdivergence calculates relative entropy between the probability distributions, and the cross-entropy calculates the total entropy between two probability distributions. the cross-entropy between two probability distribution (p and q) can be define as H$_{(p,q)}$ $=$ $-E_p[log_q]$. $where E_p[*]\ is\ the\ operator\ of\ the\ expected\ value\ with\ respect\ to\ p\ distribution.$

3.5.1 Categorical cross-entropy

Categorical cross-entropy is used as loss function in our three models (CNN, RNN, LSTM's) it is used for the multi-class classification tasks. Multi-class classification are tasks where from multiple categories the output belongs to one class in our case we are predicting 65 events of drug-drug interaction so for that reason we use categorical cross-entropy. The mathematical formula of the categorical cross-entropy loss function is below.

$$\sum_{i=1}^{outputsize} t_i.logy_i.$$

3.6 Optimizer

Adam is used as an optimizer in the model which is presented by the researcher in this article[37]. Instead of using the traditional stochastic gradient descent procedure for the network weight updation iteratively based on training data we can use adam. The researcher describes in the paper that the adam optimizer is combining the advantages of two other stochastic gradient descent extensions which are Adaptive Gradient Algorithm and Root Mean Square Propagation. The benefits of adam are given below.
1) It is easy to implement.
2)The Hyper-parameters required little tuning. 3) It work good on little memory 4) Not Computationally expensive.

3.7 Drop Out

The drop out in neural networks refers to the dismissal of neurons in neural networks in both visible and secret ways. Deep neural networks with a huge number of parameters are very in demand in ML systems so in such a system the over-fitting is a very serious problem[38]. Using large network are slow to use and also at test time while combining the prediction of numerous different huge neural networks makes it hard to deal with over-fitting. So we use drop-out in our models for overcoming the above problem.

3.8 Batch Normalization

Training deep neural networks with more amount of layers is difficult because these layers can be sensitive to the learning algorithm configuration and to the random initial weights. Inputs distribution can be one of the possible reasons for the above difficulty where the network may change when the weights are updated after each mini-batch iterations and this can make the learning algorithm to chase moving target forever. So to resolve the above issue we will use batch normalization in our models. Batch Normalization is an approach to train deep neural networks where batch-normalization standardizes the inputs to layers for every mini-batch.

Chapter 4

Analysis and Evaluation

In this chapter we will discuss the implementation of different models on drug bank database, experimental setups and the evaluation of the model.

4.1 Experimental Setup

PyCharm 2020 is used for code writing and implementation it is an IDE (Integrated Development Environment). And for training and testing the model the system we have used with following specification which are Dell Inspiron 3521 with 3rd gen 1.8 gigahertz processor, 4GB DDR3 RAM, 512 HDD and Window10 as an operating system.

4.2 Models Implementation

We have used three different models in our study which are Convolutional Neural Networks (CNN's), Recurrent Neural Networks (RNN's) and Long Short Term Memory Neural Networks (LSTM's) for the prediction of drug-drug interactions events on the Drug-bank database below we will explained each and every model and its hyper-parameters.

4.3 Evaluation Measure

For the results we generate classification report in our study the evaluation measures includes accuracy measures, F1-score, precision, AUC, AUPR and recall on both macro and micro techniques. All the above measure will be used for the model results evaluations.

4.3.1 Accuracy

Accuracy is the number of correctly predicted data points out of all the data points.
Mathematical Formula

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

4.3.2 Precision

Precision is the fragment of true positive examples between the classified positive example by the model.
Mathematical Formula

$$Precision = \frac{TP}{TP+FP}$$

4.3.3 Recall

Recall is the fragment of positive examples between the total positive examples. It is also known as sensitivity.
Mathematical Formula

$$Recall = \frac{TP}{TP+FN}$$

4.3.4 F1-Score

F1-score is the harmonic mean of the recall and precision. A model can have high F1-score value of 1 which can be consider a perfect model.
Mathematical Formula

$$F1 - Score = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

4.3.5 AUC

AUC stands for Area Under the Curve which measure the ability of the classifier to differentiate between classes and is used as a summary of the ROC curve.

4.4 CNN as a model

First we use multi-modal 1D-CNN as a model on the drug bank data-sets for the prediction of drug-drug interaction events and achieved an accuracy of 90.00%.

4.4.1 Inputs to the model

In the first step we have input four features of drugs then calculate binary features vectors by using on-hot encoding for each and every features of drugs then use similarity measures like Jacard Similarity for the calculation of similarity matrices for Chemical Substructure, Enzymes, Pathways and Targets which produces a metrics of (572 x 572) for each and every features of drugs and then prepare these metrics for the CNN model.

4.4.2 Layers

1. The layers of the model are explained below.

    - In the CNN model we use the input layer with the filters size of 1 and the kernel size of 5 with the "tanh" activation function we use the "tanh" as an activation function because while calculating the similarity matrix using the jacard similarity measures it produces some negative values so we use "tanh" to use both positive and negative values.
    - Then we use the flatten layer to convert the data into one dimensional array for inputting to the next layer.
    - After flatten layer uses a dense layer of 1024,512,256 neuron and used an "elu" as an activation function.
    - And with each and every dense layer we use "BachNormalization" layer for the normalization of the previous output layer and for avoiding over-fitting the model and also use a drop-out layer of value 0.3 with every dense layer.
    - Then we use a dense layer of 65 neuron as an output layer because we are classifying 65 different types of drug-drug interaction with the "softmax" activation function.

4.4.3 Implementation Details

The details of the final CNN modal we have used for our research are given in table 4.1. The D-Fun column represent Dense layer activation function, I-Fun column represent Input layer activation function, BatchNorm column represent Batch-Normalization with every dense layer and O-Fun column represent Output layer activation function of the models. We have used python keras library for writing code of out model.

4.4.4 Optimizer

Adam (adaptive moment estimation) optimizer is used in the model for optimization.

layers	I-Fun	D-Fun	O-Fun	Optimizer	Loss-Fun	Dropout	Batch Norm	Flatten Layers
4	tanh	elu	softmax	adam	categorical-crossentropy	0.3	3	1

Table 4.1: CNN Model Layers and Hyper-parameters

4.4.5 Loss Function

Categorical-Crossentropy loss is used in the model as a loss function.

4.4.6 Avoid Overfitting Techniques

1. For avoiding over fitting problem below two techniques are used.
   - **Bach Normalization :** Bach Normalization layer is used with every dense layer to avoid over-fitting.
   - **Drop Out** Drop out of value 0.3 is added with every dense layer to avoid over-fitting of the model.

4.4.7 Training and Testing Data

For training and testing the model the 5-k fold cross validation is used to divide data into 5 subsets and then use four subsets for the training of model and one subset for the validation of the model.

4.4.8 Evaluation Measure

The model results reports are generated which includes all the measures like AUC,AUPR, Accuracy, F1-score, Recall, and Precision.

4.5 RNN as a model

We use multi-modal RNN's as second model in our study on the drug bank data-sets for the prediction of drug-drug interaction events and achieved an accuracy of 88.66%.

4.5.1 Inputs to the model

In the first step we have input four features of drugs then calculate binary features vectors by using on-hot encoding for each and every features of drugs then use similarity measures like Jacard Similarity for the calculation of similarity matrices for Chemical Substructure, Enzymes, Pathways and Targets which produces a metrics of (572 x 572) for each and every features of drugs and then prepare these metrics for the RNN's model.

4.5.2 Layers

1. The layers of the model are explained below..
   - In the RNN model we use a dense layer of 65 neuron for inputs .
   - Then use three dense layers of 1024,512,256 neuron with elu "activation" functions.
   - And with each and every dense layer we use "BachNormalization" layer for the normalization of the previous output layer and for avoiding over-fitting the model and also use a drop-out layer of value 0.3 with every dense layer.
   - Then we use a dense layer of 65 neuron as an output layer because we are classifying 65 different types of drug-drug interaction with the "softmax" activation function.

4.5.3 Implementation Details

The details of the final RNN modal we have used for our research are given in table 4.2. The D-Fun column represent Dense layer activation function, I-Fun column represent Input layer activation function, BatchNorm column represent Batch-Normalization with every dense layer and O-Fun column represent Output layer activation function of the models. We have used python keras library for writing code of out model.

layers	I-Fun	D-Fun	O-Fun	Optimizer	Loss-Fun	Dropout	Batch Norm
4	None	elu	softmax	adam	categorical-crossentropy	0.3	3

Table 4.2: RNN Model Layers and Hyper-parameters

4.5.4 Optimizer

Adam (adaptive moment estimation) optimizer is used in the model for optimization.

4.5.5 Loss Function

Categorical-Cross-entropy loss is used in the model as a loss function.

4.5.6 Avoid Over-fitting Techniques

1. For avoiding over-fitting problem below two techniques are used.
   - **Bach Normalization :** Bach Normalization layer is used with every dense layer to avoid over-fitting.

- **Drop Out** Drop out of value 0.3 is added with every dense layer to avoid over-fitting of the model.

4.5.7 Training and Testing Data

For training and testing the model the 5-k fold cross validation is used to divide data into 5 subsets and then use four subsets for the training of model and one subset for the validation of the model.

4.5.8 Evaluation Measure

The model results reports are generated which includes all the measures like AUC,AUPR, Accuracy, F1-score, Recall, and Precision.

4.6 LSTM's as a model

We use multi-modal LSTM's as a third model in our study on the drug bank data-sets for drug-drug interaction associated events prediction and achieved an accuracy of 89.02%.

4.6.1 Inputs to the model

In the first step we have input four features of drugs then calculate binary features vectors by using on-hot encoding for each and every features of drugs then use similarity measures like Jacard Similarity for the calculation of similarity matrices for Chemical Substructure, Enzymes, Pathways and Targets which produces a metrics of (572 x 572) for each and every features of drugs and then prepare these metrics for the LSTM's model.

4.6.2 Layers

1. The layers of the model are explained below.

    - In the LSTM's model we use a dense layer of 572 neuron for inputs and use "tanh" as an activation function.
    - Then use three dense layers of 1024,512,256 neuron with "elu" activation functions.
    - And with each and every dense layer we use "BachNormalization" layer for the normalization of the previous output layer and for avoiding over-fitting the model and also use a drop-out layer of value 0.3 with every dense layer.
    - Then we use a dense layer of 65 neuron as an output layer because we are classifying 65 different types of drug-drug interaction with the "softmax" activation function.

4.6.3 Implementation Details

The details of the final LSTM modal we have used for our research are given in table 4.3. The D-Fun column represent Dense layer activation function, I-Fun column Stands for Input layer activation function, BatchNorm column represent Batch-Normalization with every dense layer and O-Fun column represent Output layer activation function of the models. We have used python keras library for writing code of out model.

layers	I-Fun	D-Fun	O-Fun	Optimizer	Loss-Fun	Dropout	Batch Norm
4	tanh	elu	softmax	adam	categorical-crossentropy	0.3	3

Table 4.3: LSTM Model Layers and Hyper-parameters

4.6.4 Optimizer

Adam (adaptive moment estimation) optimizer is used in the model for optimization.

4.6.5 Loss Function

Categorical-Cross-entropy loss is used in the model as a loss function.

4.6.6 Avoid Over-fitting Techniques

1. For avoiding over-fitting problem below two techniques are used.
    - **Bach Normalization :** Bach Normalization layer is used with every dense layer to avoid over-fitting.
    - **Drop Out** Drop out of value 0.3 is added with every dense layer to avoid over-fitting of the model.
2.

4.6.7 Training and Testing Data

For training and testing the model the 5-k fold cross validation is used to divide data into 5 subsets and then use four subsets for the training of model and one subset for the validation of the model.

4.6.8 Evaluation Measure

The model results reports are generated which includes all the measures like AUC,AUPR, Accuracy, F1-score, Recall, and Precision.

4.7 Results

All the models are trained on the drug bank data-set and achieved an accuracy of 90.00% on the CNN model, 88.66% on the RNN model, and 89.02% on the LSTM's model.

4.7.1 CNN model Results

First we train our CNN model on four features of drugs separately with jacard similarity measures then use combinations of features for better results and different experiment show that the following three features give best result which are Smiles (Chemical Substructures), Targets, Enzymes. CNN model was trained on 29790 and tested on 7474 interaction. Below tables have results for RNN model with different experiments of layers, activation functions, features set and jacard similarity measure.

layers	ACC	F1	AUC	AUPR	Precision	Recall
2	0.7631	0.6222	0.9664	0.8267	0.7425	0.6587
3	0.8963	0.8086	0.9981	0.9449	0.8859	0.7705
4	0.9000	0.8286	0.9981	0.9478	0.8840	0.7967

Table 4.4: CNN results on different numbers of layers

layers	I-Fun	D-Fun	Similarity Matrices	ACC	F1	AUC	AUPR	Precision	Recall
2	tanh	tanh	jacard	0.7631	0.6222	0.9664	0.8267	0.7425	0.6587
3	tanh	tanh	jacard	0.8826	0.7767	0.9979	0.9338	0.8840	0.7417
3	sigmoid	sigmoid	jacard	0.4166	0.4371	0.9664	0.3904	0.1926	0.0393
3	tanh	elu	jacard	0.8963	0.8086	0.9981	0.9449	0.8859	0.7705
3	tanh	relu	jacard	0.7952	0.5440	0.9961	0.8636	0.6775	0.5011
4	tanh	relu	jacard	0.8830	0.7779	0.9978	0.9344	0.8306	0.7512
4	tanh	elu	jacard	0.9000	0.8286	0.9981	0.9478	0.8840	0.7967

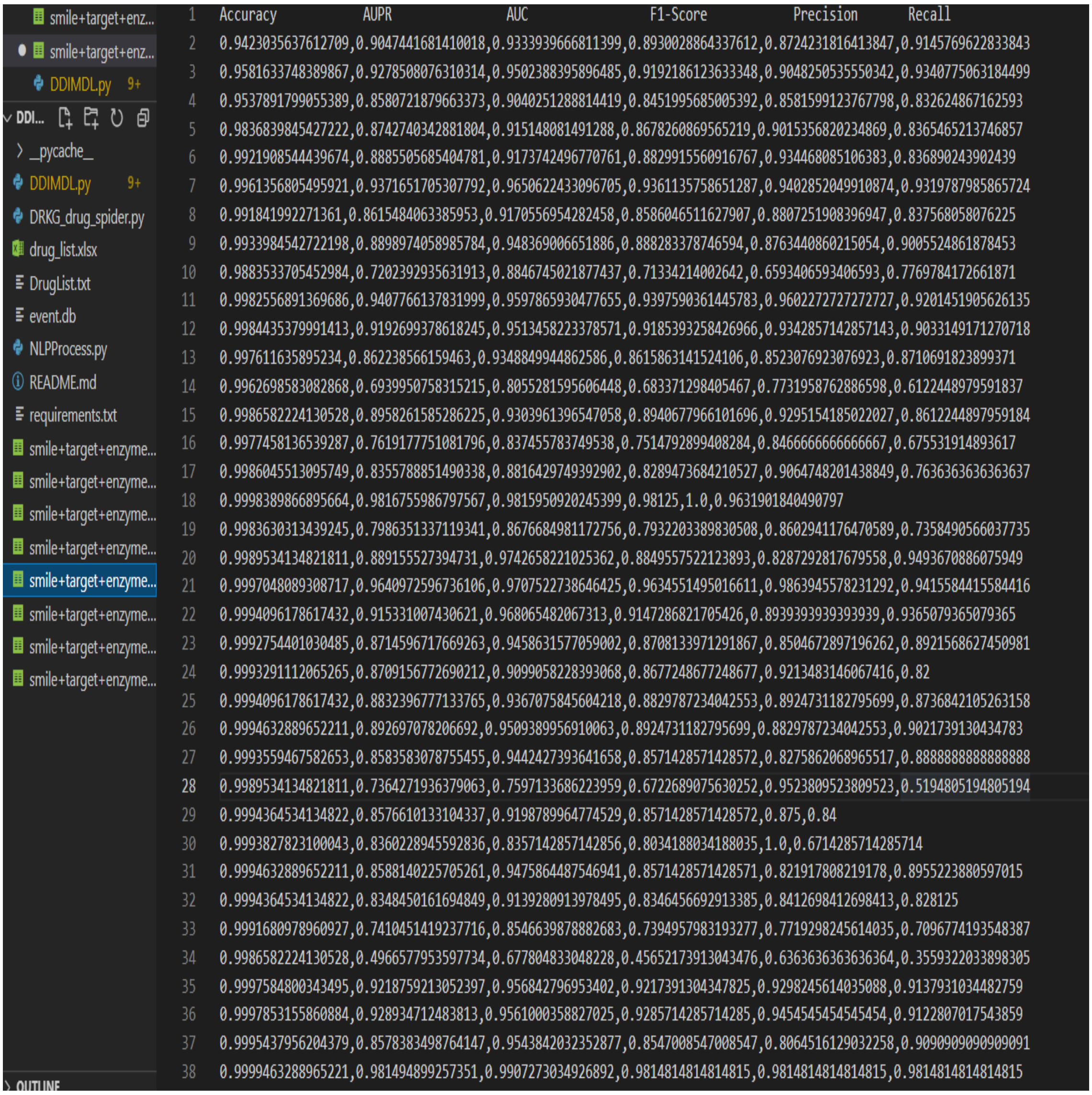
Table 4.5: CNN results on different activation function and Jacard Similarity measures

Set of Features	**Similarity Matrices**	**ACC**	**F1**	**AUC**	**AUPR**	**Precision**	**Recall**
Smiles	jacard	0.8861	0.8099	0.9983	0.9505	0.8377	0.7992
Target	jacard	0.8441	0.7517	0.9977	0.9254	0.7931	0.7271
Enzyme	jacard	0.6808	0.4584	0.9920	0.7691	0.5679	0.4102
Pathway	jacard	0.8317	0.7357	0.9975	0.9169	0.7722	0.7165
Smiles + Target	jacard	0.8986	0.8264	0.9985	0.9549	0.8611	0.8067
Smiles + Enzyme	jacard	0.8933	0.7943	0.9976	0.9309	0.8548	0.7640
Smiles + Pathway	jacard	0.8993	0.8294	0.9986	0.9552	0.8545	0.8131
Target + Pathway	jacard	0.8509	0.7667	0.9980	0.9313	0.8001	0.7537
Target + Enzyme	jacard	0.8658	0.7498	0.9972	0.9158	0.8059	0.7220
Pathway + Enzyme	jacard	0.8635	0.7563	0.9973	0.9160	0.7995	0.7319
Smile + Pathway + Enzyme	jacard	0.8998	0.8102	0.9981	0.9464	0.8536	0.7847
Smile + Target + Enzyme	jacard	0.9000	0.8286	0.9981	0.9478	0.8840	0.7967
Smile + Target + Pathway	jacard	0.8849	0.8077	0.9986	0.9530	0.8429	0.7879
Target + Pathway + Enzyme	jacard	0.8672	0.7833	0.9976	0.9292	0.8229	0.7630
Smiles + Target + Pathway + Enzyme	jacard	0.8953	0.8164	0.992	0.9503	0.8694	0.7888

Table 4.6: CNN results on different on different features and Jacard Similarity measures

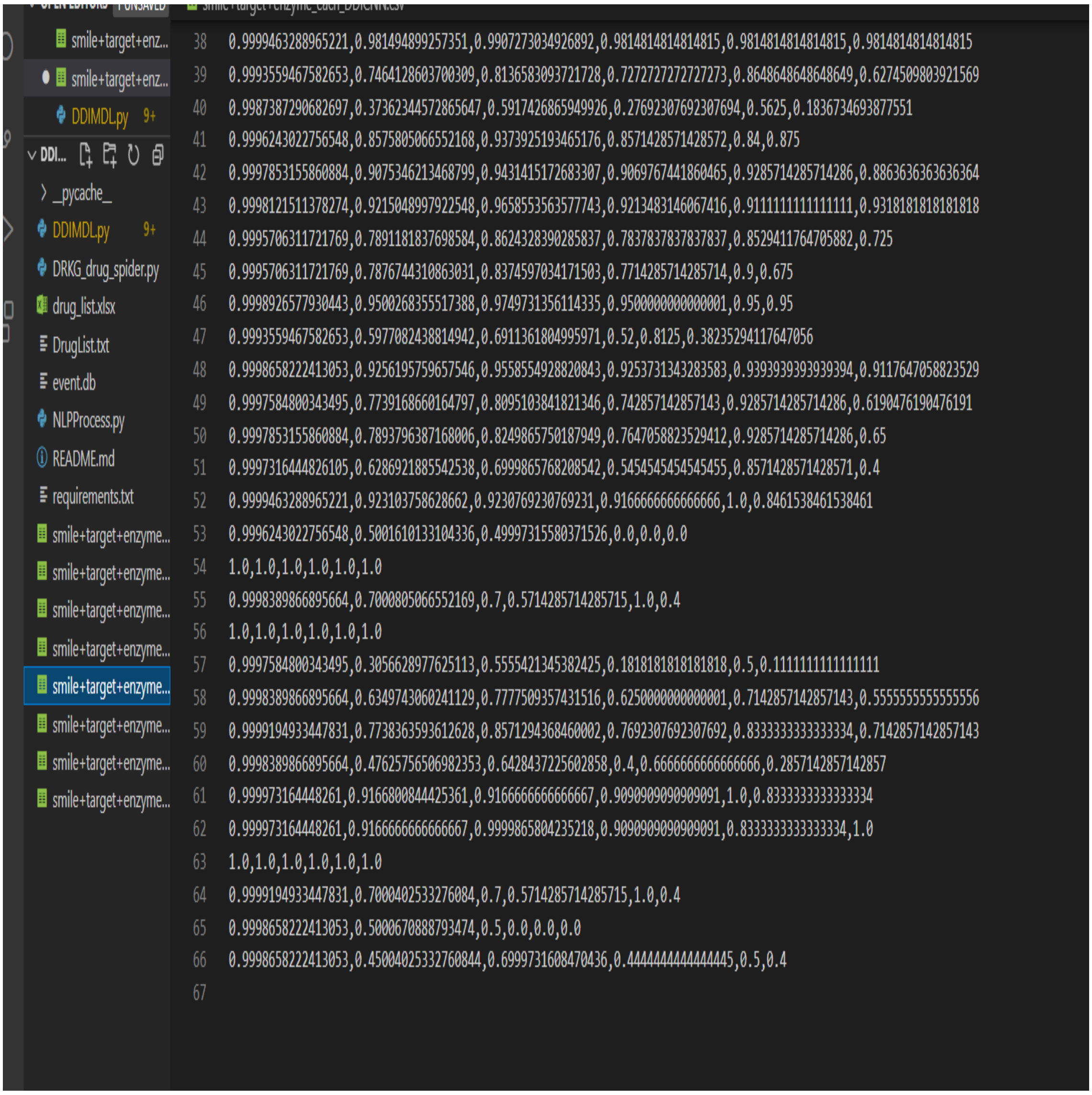
4.7.2 CNN Each Event Results

As we know that we have 65 types of different drug-associated events which are we predicting in our work. There fore after the results of drug features further we investigate the performance of our model for each of the events and the matrices score is calculated for each independently by using real values and predicted scores. It is likely that the events with higher frequency can gain better performances. Our models gives AUPR scores greater than 0.5 for the events numbered from 1 to 46 as you can see in the below screenshots except 39. Below are the screenshots of each and every event's predictions results of the CNN model.



```
Accuracy            AUPR                AUC                 F1-Score            Precision       Recall
0.9423035637612709,0.9047441681410018,0.9333939666811399,0.8930028864337612,0.8724231816413847,0.9145769622833843
0.9581633748389867,0.9278508076310314,0.9502388395896485,0.9192186123633348,0.9048250535550342,0.9340775063184499
0.9537891799055389,0.8580721879663373,0.9040251288814419,0.8451995685005392,0.8581599123767798,0.832624867162593
0.9836839845427222,0.8742740342881804,0.915148081491288,0.8678260869565219,0.9015356820234869,0.8365465213746857
0.9921908544439674,0.8885505685404781,0.9173742496770761,0.8829915560916767,0.934468085106383,0.836890243902439
0.9961356805495921,0.9371651705307792,0.9650622433096705,0.9361135758651287,0.9402852049910874,0.9319787985865724
0.991841992271361,0.8615484063385953,0.9170556954282458,0.8586046511627907,0.8807251908396947,0.837568058076225
0.9933984542722198,0.8898974058985784,0.948369006651886,0.888283378746594,0.8763440860215054,0.9005524861878453
0.9883533705452984,0.7202392935631913,0.8846745021877437,0.71334214002642,0.6593406593406593,0.7769784172661871
0.9982556891369686,0.9407766137831999,0.9597865930477655,0.9397590361445783,0.9602272727272727,0.9201451905626135
0.9984435379991413,0.9192699378618245,0.9513458223378571,0.9185393258426966,0.9342857142857143,0.9033149171270718
0.997611635895234,0.862238566159463,0.9348849944862586,0.8615863141524106,0.8523076923076923,0.8710691823899371
0.9962698583082868,0.6939950758315215,0.8055281595606448,0.683371298405467,0.7731958762886598,0.6122448979591837
0.9986582224130528,0.8958261585286225,0.9303961396547058,0.8940677966101696,0.9295154185022027,0.8612244897959184
0.9977458136539287,0.7619177751081796,0.837455783749538,0.7514792899408284,0.8466666666666667,0.675531914893617
0.9986045513095749,0.8355788851490338,0.8816429749392902,0.8289473684210527,0.9064748201438849,0.7636363636363637
0.9998389866895664,0.9816755986797567,0.9815950920245399,0.98125,1.0,0.9631901840490797
0.9983630313439245,0.7986351337119341,0.8676684981172756,0.7932203389830508,0.8602941176470589,0.7358490566037735
0.9989534134821811,0.889155527394731,0.9742658221025362,0.8849557522123893,0.8287292817679558,0.9493670886075949
0.9997048089308717,0.9640972596736106,0.9707522738646425,0.9634551495016611,0.9863945578231292,0.9415584415584416
0.9994096178617432,0.915331007430621,0.968065482067313,0.9147286821705426,0.8939393939393939,0.9365079365079365
0.9992754401030485,0.8714596717669263,0.9458631577059002,0.8708133971291867,0.8504672897196262,0.8921568627450981
0.9993291112065265,0.8709156772690212,0.9099058228393068,0.8677248677248677,0.9213483146067416,0.82
0.9994096178617432,0.8832396777133765,0.9367075845604218,0.8829787234042553,0.8924731182795699,0.8736842105263158
0.9994632889652211,0.892697078206692,0.9509389956910063,0.8924731182795699,0.8829787234042553,0.9021739130434783
0.9993559467582653,0.8583583078755455,0.9442427393641658,0.8571428571428572,0.8275862068965517,0.8888888888888888
0.9989534134821811,0.7364271936379063,0.7597133686223959,0.6722689075630252,0.9523809523809523,0.5194805194805194
0.9994364534134822,0.8576610133104337,0.9198789964774529,0.8571428571428572,0.875,0.84
0.9993827823100043,0.8360228945592836,0.8357142857142856,0.8034188034188035,1.0,0.6714285714285714
0.9994632889652211,0.8588140225705261,0.9475864487546941,0.8571428571428571,0.821917808219178,0.8955223880597015
0.9994364534134822,0.8348450161694849,0.9139280913978495,0.8346456692913385,0.8412698412698413,0.828125
0.9991680978960927,0.7410451419237716,0.8546639878882683,0.7394957983193277,0.7719298245614035,0.7096774193548387
0.9986582224130528,0.4966577953597734,0.677804833048228,0.45652173913043476,0.6363636363636364,0.3559322033898305
0.9997584800343495,0.9218759213052397,0.956842796953402,0.9217391304347825,0.9298245614035088,0.9137931034482759
0.9997853155860884,0.928934712483813,0.9561000358827025,0.9285714285714285,0.9454545454545454,0.9122807017543859
0.9995437956204379,0.8578383498764147,0.9543842032352877,0.8547008547008547,0.8064516129032258,0.9090909090909091
0.9999463288965221,0.981494899257351,0.9907273034926892,0.9814814814814815,0.9814814814814815,0.9814814814814815
```

Figure 4.1: Each and every event evaluation measures values using CNN’s



```
0.9999463288965221,0.981494899257351,0.9907273034926892,0.9814814814814815,0.9814814814814815,0.9814814814814815
0.9993559467582653,0.7464128603700309,0.8136583093721728,0.7272727272727273,0.8648648648648649,0.6274509803921569
0.9987387290682697,0.37362344572865647,0.5917426865949926,0.27692307692307694,0.5625,0.1836734693877551
0.9996243022756548,0.8575805066552168,0.9373925193465176,0.8571428571428572,0.84,0.875
0.9997853155860884,0.9075346213468799,0.9431415172683307,0.9069767441860465,0.9285714285714286,0.8863636363636364
0.9998121511378274,0.9215048997922548,0.9658553563577743,0.9213483146067416,0.9111111111111111,0.9318181818181818
0.9995706311721769,0.7891181837698584,0.8624328390285837,0.7837837837837837,0.8529411764705882,0.725
0.9995706311721769,0.7876744310863031,0.8374597034171503,0.7714285714285714,0.9,0.675
0.9998926577930443,0.9500268355517388,0.9749731356114335,0.9500000000000001,0.95,0.95
0.9993559467582653,0.5977082438814942,0.6911361804995971,0.52,0.8125,0.38235294117647056
0.9998658222413053,0.9256195759657546,0.9558554928820843,0.9253731343283583,0.9393939393939394,0.9117647058823529
0.9997584800343495,0.7739168660164797,0.8095103841821346,0.742857142857143,0.9285714285714286,0.6190476190476191
0.9997853155860884,0.7893796387168006,0.8249865750187949,0.7647058823529412,0.9285714285714286,0.65
0.9997316444826105,0.6286921885542538,0.6999865768208542,0.5454545454545455,0.8571428571428571,0.4
0.9999463288965221,0.923103758628662,0.9230769230769231,0.9166666666666666,1.0,0.8461538461538461
0.9996243022756548,0.5001610133104336,0.49997315580371526,0.0,0.0,0.0
1.0,1.0,1.0,1.0,1.0,1.0
0.9998389866895664,0.7000805066552169,0.7,0.5714285714285715,1.0,0.4
1.0,1.0,1.0,1.0,1.0,1.0
0.9997584800343495,0.3056628977625113,0.5555421345382425,0.1818181818181818,0.5,0.1111111111111111
0.9998389866895664,0.6349743060241129,0.7777509357431516,0.6250000000000001,0.7142857142857143,0.5555555555555556
0.9999194933447831,0.7738363593612628,0.8571294368460002,0.7692307692307692,0.8333333333333334,0.7142857142857143
0.9998389866895664,0.47625756506982353,0.6428437225602858,0.4,0.6666666666666666,0.2857142857142857
0.999973164448261,0.9166800844425361,0.9166666666666667,0.9090909090909091,1.0,0.8333333333333334
0.999973164448261,0.9166666666666667,0.9999865804235218,0.9090909090909091,0.8333333333333334,1.0
1.0,1.0,1.0,1.0,1.0,1.0
0.9999194933447831,0.7000402533276084,0.7,0.5714285714285715,1.0,0.4
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
0.9998658222413053,0.45004025332760844,0.6999731608470436,0.4444444444444445,0.5,0.4
```

Figure 4.2: Each and every event evaluation measures values using CNN’s

ROC graph of CNN’s model are given below

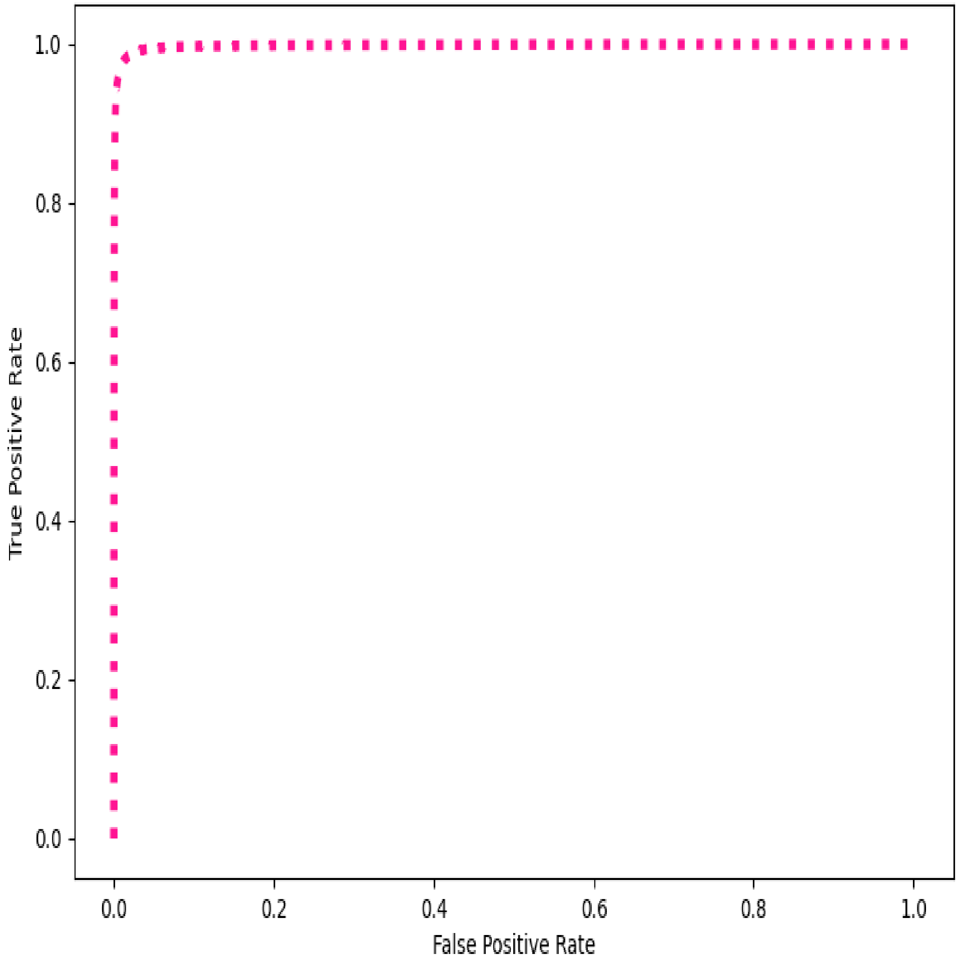


Figure 4.3: CNN’s ROC graph

4.7.3 RNN model Results

For training and testing we have followed the same procedure as for CNN model which has been discuss in section 4.5.1 below tables have results for RNN model with different experiments of layers, activation functions, features set and jacard measures.

layers	Similarity Matrices	ACC	F1	AUC	AUPR	Precision	Recall
2	jacard	0.7841	0.6220	0.9554	0.8361	0.7155	0.6023
3	jacard	0.8738	0.6903	0.9973	0.9292	0.7806	0.6440
4	jacard	0.8866	0.7779	0.9979	0.9395	0.8503	0.7396

Table 4.7: RNN results on different layers and Jacard Similarity measures

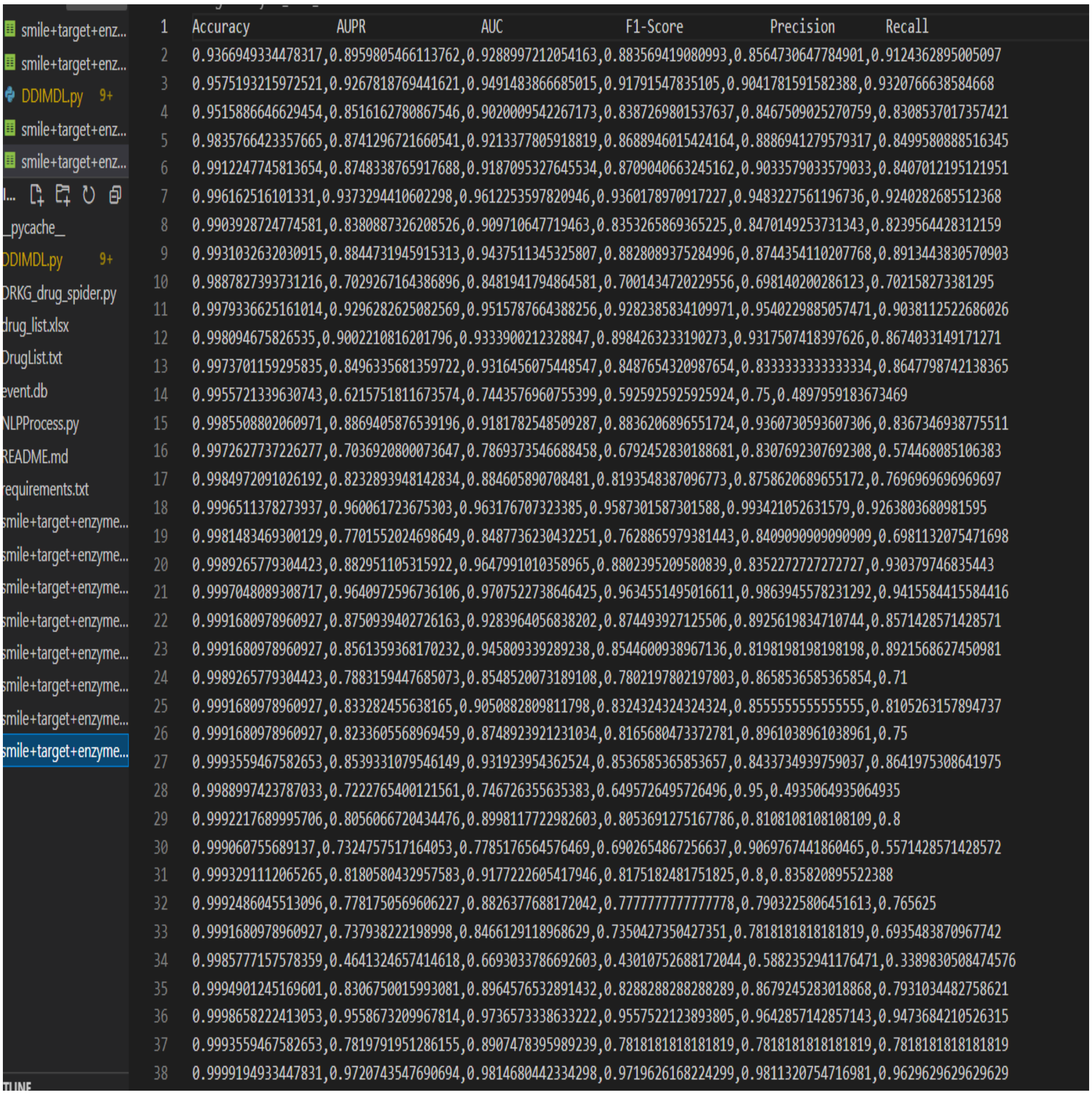
Set of Features	Similarity Matrices	ACC	F1	AUC	AUPR	Precision	Recall
Smiles	jacard	0.8632	0.7623	0.9979	0.9334	0.8056	0.7402
Target	jacard	0.8346	0.7434	0.9969	0.9145	0.7846	0.7190
Enzyme	jacard	0.6747	0.4330	0.9911	0.7563	0.5230	0.3900
Pathway	jacard	0.8218	0.7186	0.9970	0.9076	0.7462	0.7042
Smiles + Target	jacard	0.8874	0.7890	0.9982	0.9460	0.8420	0.7650
Smiles + Enzyme	jacard	0.8725	0.7484	0.9973	0.9185	0.8318	0.7108
Smiles + Pathway	jacard	0.8846	0.7824	0.9983	0.9445	0.8287	0.7567
Target + Pathway	jacard	0.8434	0.7446	0.9976	0.9220	0.7835	0.7209
Target + Enzyme	jacard	0.8554	0.7325	0.9969	0.9072	0.8088	0.6965
Pathway + Enzyme	jacard	0.8502	0.7115	0.9968	0.9051	0.7735	0.6874
Smiles + Pathway + Enzyme	jacard	0.8856	0.7766	0.9979	0.9396	0.8483	0.7396
Smiles + Target + Enzyme	jacard	0.8866	0.7779	0.9979	0.9395	0.8503	0.7396
Smiles + Target +Pathway	jacard	0.8746	0.7732	0.9983	0.9447	0.8181	0.7479
Target + Pathway + Enzyme	jacard	0.8583	0.7518	0.9974	0.9229	0.8100	0.7205
Smiles + Target +Pathway + Enzyme	jacard	0.8821	0.7785	0.9982	0.9433	0.8530	0.7449

Table 4.8: RNN results on different on different features and Jacard Similarity measures

4.7.4 RNN Each Event Results

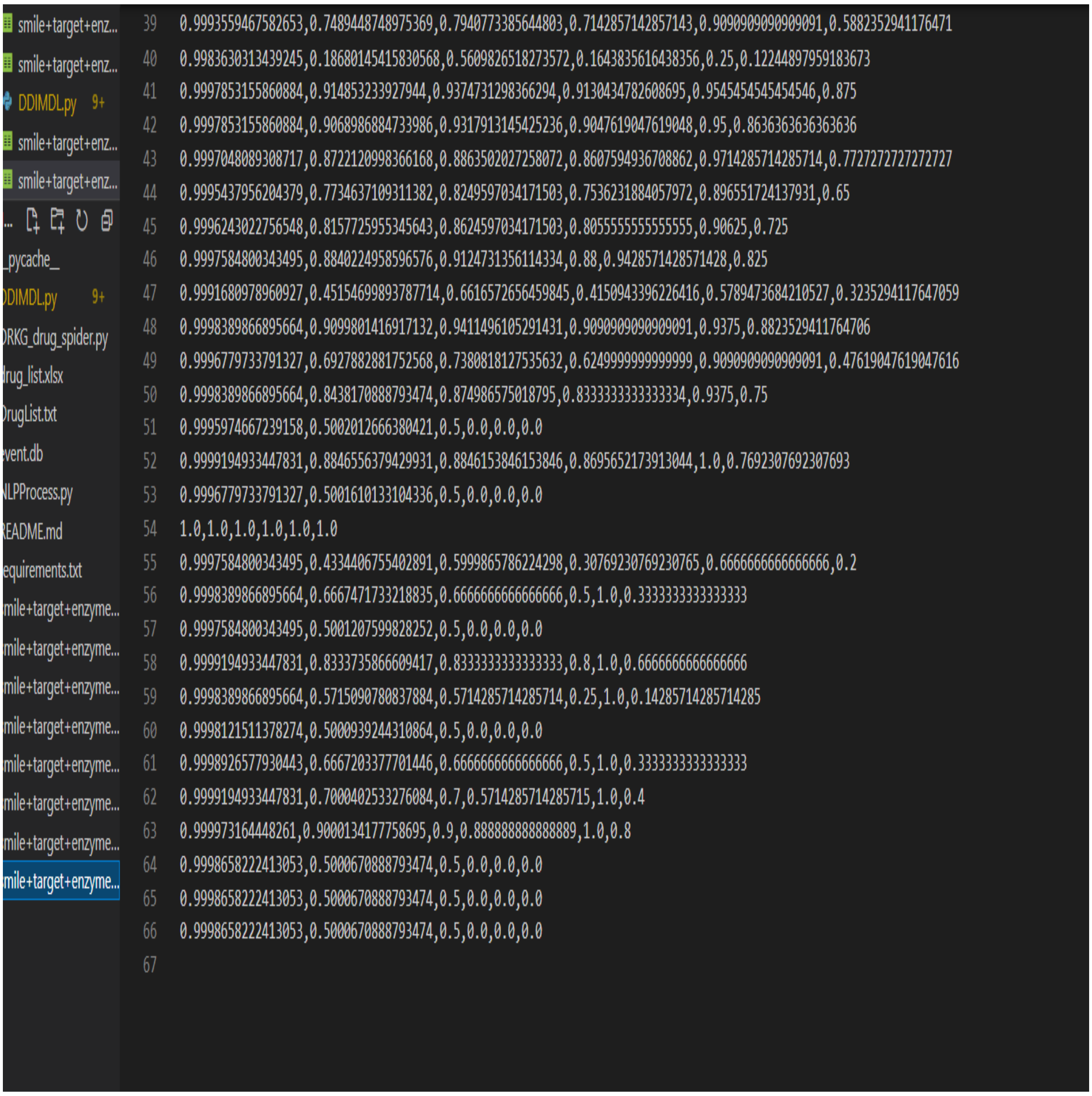
As we know that we have 65 types of different drug-associated events which are we predicting in our work. There fore after the results of drug features further we investigate the performance of our model for each of the events and the matrices score is calculated for each independently by using real values and predicted scores. It is likely that the events with higher frequency can gain better performances. Our models gives AUPR scores greater than 0.5 for the events numbered from 1 to 46 as you can see in the below screenshots except 39.

Below are the screenshots of each and every event's predictions results of the RNN model.



```
Accuracy            AUPR                AUC                 F1-Score            Precision       Recall
0.9366949334478317,0.8959805466113762,0.9288997212054163,0.883569419080993,0.8564730647784901,0.9124362895005097
0.9575193215972521,0.9267818769441621,0.9491483866685015,0.91791547835105,0.9041781591582388,0.9320766638584668
0.951588664629454,0.8516162780867546,0.9020009542267173,0.8387269801537637,0.8467509025270759,0.8308537017357421
0.9835766423357665,0.8741296721660541,0.9213377805918819,0.8688946015424164,0.8886941279579317,0.8499580888516345
0.9912247745813654,0.8748338765917688,0.9187095327645534,0.8709040663245162,0.9033579033579033,0.8407012195121951
0.996162516101331,0.9373294410602298,0.9612253597820946,0.9360178970917227,0.9483227561196736,0.9240282685512368
0.9903928724774581,0.8380887326208526,0.909710647719463,0.8353265869365225,0.8470149253731343,0.8239564428312159
0.9931032632030915,0.8844731945915313,0.9437511345325807,0.8828089375284996,0.8744354110207768,0.8913443830570903
0.9887827393731216,0.7029267164386896,0.8481941794864581,0.7001434720229556,0.698140200286123,0.702158273381295
0.9979336625161014,0.9296282625082569,0.9515787664388256,0.9282385834109971,0.9540229885057471,0.9038112522686026
0.998094675826535,0.9002210816201796,0.9333900212328847,0.8984263233190273,0.9317507418397626,0.8674033149171271
0.9973701159295835,0.8496335681359722,0.9316456075448547,0.8487654320987654,0.8333333333333334,0.8647798742138365
0.9955721339630743,0.6215751811673574,0.7443576960755399,0.5925925925925924,0.75,0.4897959183673469
0.9985508802060971,0.8869405876539196,0.9181782548509287,0.8836206896551724,0.9360730593607306,0.8367346938775511
0.9972627737226277,0.7036920800073647,0.7869373546688458,0.6792452830188681,0.8307692307692308,0.574468085106383
0.9984972091026192,0.8232893948142834,0.884605890708481,0.8193548387096773,0.8758620689655172,0.7696969696969697
0.9996511378273937,0.960061723675303,0.963176707323385,0.9587301587301588,0.993421052631579,0.9263803680981595
0.9981483469300129,0.7701552024698649,0.8487736230432251,0.7628865979381443,0.8409090909090909,0.6981132075471698
0.9989265779304423,0.882951105315922,0.9647991010358965,0.8802395209580839,0.8352272727272727,0.930379746835443
0.9997048089308717,0.9640972596736106,0.9707522738646425,0.9634551495016611,0.9863945578231292,0.9415584415584416
0.9991680978960927,0.8750939402726163,0.9283964056838202,0.874493927125506,0.8925619834710744,0.8571428571428571
0.9991680978960927,0.8561359368170232,0.945809339289238,0.8544600938967136,0.8198198198198198,0.8921568627450981
0.9989265779304423,0.7883159447685073,0.8548520073189108,0.7802197802197803,0.8658536585365854,0.71
0.9991680978960927,0.833282455638165,0.9050882809811798,0.8324324324324324,0.8555555555555555,0.8105263157894737
0.9991680978960927,0.8233605568969459,0.8748923921231034,0.8165680473372781,0.8961038961038961,0.75
0.9993559467582653,0.8539331079546149,0.931923954362524,0.8536585365853657,0.8433734939759037,0.8641975308641975
0.9988997423787033,0.7222765400121561,0.746726355635383,0.6495726495726496,0.95,0.4935064935064935
0.9992217689995706,0.8056066720434476,0.8998117722982603,0.8053691275167786,0.8108108108108109,0.8
0.999060755689137,0.7324757517164053,0.7785176564576469,0.6902654867256637,0.9069767441860465,0.5571428571428572
0.9993291112065265,0.8180580432957583,0.9177222605417946,0.8175182481751825,0.8,0.835820895522388
0.9992486045513096,0.7781750569606227,0.8826377688172042,0.7777777777777778,0.7903225806451613,0.765625
0.9991680978960927,0.737938222198998,0.8466129118968629,0.7350427350427351,0.7818181818181819,0.6935483870967742
0.9985777157578359,0.4641324657414618,0.6693033786692603,0.43010752688172044,0.5882352941176471,0.3389830508474576
0.9994901245169601,0.8306750015993081,0.8964576532891432,0.8288288288288289,0.8679245283018868,0.7931034482758621
0.9998658222413053,0.9558673209967814,0.9736573338633222,0.9557522123893805,0.9642857142857143,0.9473684210526315
0.9993559467582653,0.7819791951286155,0.8907478395989239,0.7818181818181819,0.7818181818181819,0.7818181818181819
0.9999194933447831,0.9720743547690694,0.9814680442334298,0.9719626168224299,0.9811320754716981,0.9629629629629629
```

Figure 4.4: Each and every event evaluation measures values using RNN's



```
0.9993559467582653,0.7489448748975369,0.7940773385644803,0.7142857142857143,0.9090909090909091,0.5882352941176471
0.9983630313439245,0.18680145415830568,0.5609826518273572,0.1643835616438356,0.25,0.12244897959183673
0.9997853155860884,0.914853233927944,0.9374731298366294,0.9130434782608695,0.9545454545454546,0.875
0.9997853155860884,0.9068986884733986,0.9317913145425236,0.9047619047619048,0.95,0.8636363636363636
0.9997048089308717,0.8722120998366168,0.8863502027258072,0.8607594936708862,0.9714285714285714,0.7727272727272727
0.9995437956204379,0.7734637109311382,0.8249597034171503,0.7536231884057972,0.896551724137931,0.65
0.9996243022756548,0.8157725955345643,0.8624597034171503,0.8055555555555555,0.90625,0.725
0.9997584800343495,0.8840224958596576,0.9124731356114334,0.88,0.9428571428571428,0.825
0.9991680978960927,0.45154699893787714,0.6616572656459845,0.4150943396226416,0.5789473684210527,0.3235294117647059
0.9998389866895664,0.9099801416917132,0.9411496105291431,0.9090909090909091,0.9375,0.8823529411764706
0.9996779733791327,0.6927882881752568,0.7380818127535632,0.6249999999999999,0.9090909090909091,0.47619047619047616
0.9998389866895664,0.8438170888793474,0.874986575018795,0.8333333333333334,0.9375,0.75
0.9995974667239158,0.5002012666380421,0.5,0.0,0.0,0.0
0.9999194933447831,0.8846556379429931,0.8846153846153846,0.8695652173913044,1.0,0.7692307692307693
0.9996779733791327,0.5001610133104336,0.5,0.0,0.0,0.0
1.0,1.0,1.0,1.0,1.0,1.0
0.9997584800343495,0.4334406755402891,0.5999865786224298,0.30769230769230765,0.6666666666666666,0.2
0.9998389866895664,0.6667471733218835,0.6666666666666666,0.5,1.0,0.3333333333333333
0.9997584800343495,0.5001207599828252,0.5,0.0,0.0,0.0
0.9999194933447831,0.8333735866609417,0.8333333333333333,0.8,1.0,0.6666666666666666
0.9998389866895664,0.5715090780837884,0.5714285714285714,0.25,1.0,0.14285714285714285
0.9998121511378274,0.5000939244310864,0.5,0.0,0.0,0.0
0.9998926577930443,0.6667203377701446,0.6666666666666666,0.5,1.0,0.3333333333333333
0.9999194933447831,0.7000402533276084,0.7,0.5714285714285715,1.0,0.4
0.999973164448261,0.9000134177758695,0.9,0.888888888888889,1.0,0.8
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
```

Figure 4.5: Each and every event evaluation measures values using RNN’s

ROC graph of RNN’s model are given below

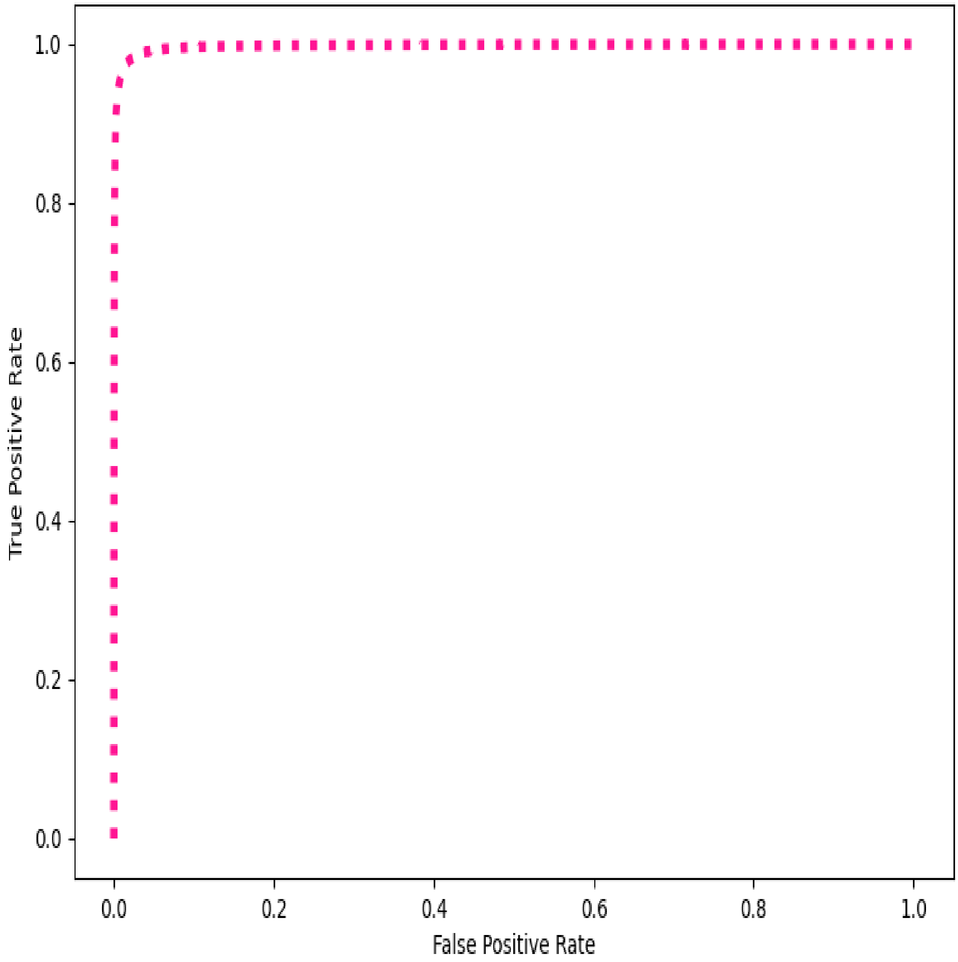


Figure 4.6: RNN’s ROC graph

4.7.5 LSTM's model Results

For training and testing we have followed the same procedure as for CNN model which has been discuss in section 4.5.1 below tables have results for LSTM's model with different experiments of layers, activation functions, features set and jacard measures.

layers	Optimizer	Similarity Matrices	ACC	F1	AUC	AUPR	Precision	Recall
2	adam	jacard	0.7932	0.6024	0.9758	0.8486	0.7262	0.6321
3	adam	jacard	0.8623	0.6461	0.9968	0.9165	0.7952	0.6912
4	adam	jacard	0.8902	0.7792	0.9978	0.9407	0.8527	0.7530
4	adadelta	jacard	0.8851	0.7104	0.9977	0.9381	0.8161	0.6671

Table 4.9: LSTM results on different numbers of layers and jacard similarity measures

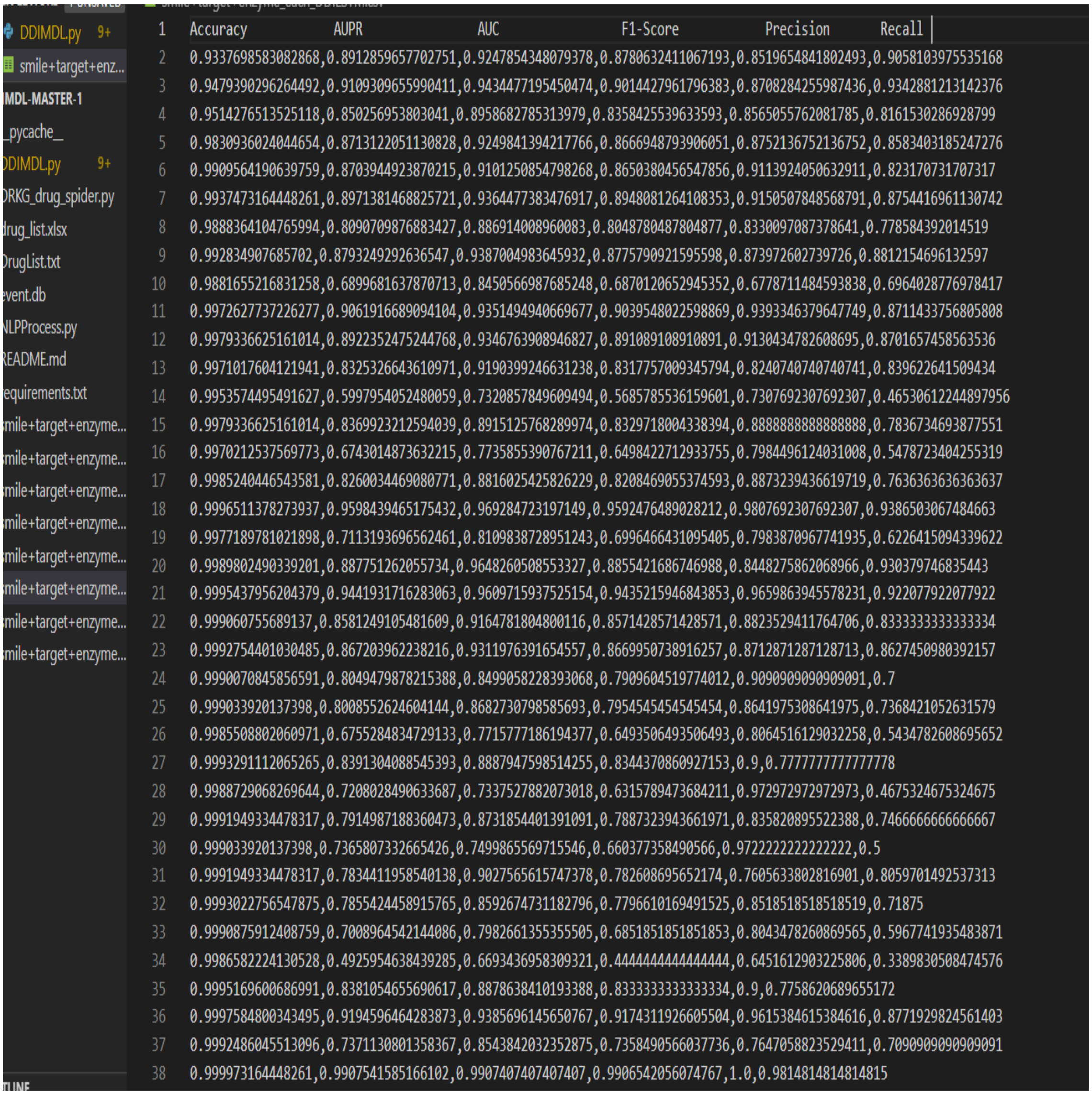
Set of Features	Similarity Matrices	ACC	F1	AUC	AUPR	Precision	Recall
Smiles	jacard	0.8645	0.7592	0.9977	0.9326	0.7922	0.7384
Target	jacard	0.8384	0.7389	0.9970	0.9160	0.7565	0.7318
Enzyme	jacard	0.6682	0.4164	0.9906	0.7497	0.5072	0.3769
Pathway	jacard	0.8273	0.7193	0.9968	0.9090	0.7475	0.7027
Smiles + Target	jacard	0.8855	0.7941	0.9982	0.9457	0.8270	0.7783
Smiles + Enzyme	jacard	0.8736	0.7495	0.9971	0.9165	0.7949	0.7250
Smiles + Pathway	jacard	0.8865	0.7915	0.9982	0.9451	0.8424	0.7702
Target + Pathway	jacard	0.8429	0.7573	0.9975	0.9218	0.7879	0.7379
Target + Enzyme	jacard	0.8593	0.7280	0.9969	0.9092	0.7810	0.6989
Pathway + Enzyme	jacard	0.8547	0.7131	0.9968	0.9058	0.7721	0.6838
Smiles + Pathway + Enzyme	jacard	0.8876	0.7792	0.9978	0.9383	0.8548	0.7441
Smiles + Target + Enzyme	jacard	0.8902	0.7792	0.9978	0.9407	0.8527	0.7430
Smiles + Target + Pathway	jacard	0.8752	0.7830	0.9982	0.9449	0.8265	0.7561
Target + Pathway + Enzyme	jacard	0.8590	0.7481	0.9973	0.9227	0.8062	0.7178
Smiles + Target + Pathway + Enzyme	jacard	0.8844	0.7851	0.9980	0.9431	0.8444	0.7527

Table 4.10: LSTM results different features set and Jacard Similarity measures

4.7.6 LSTM Each Event Results

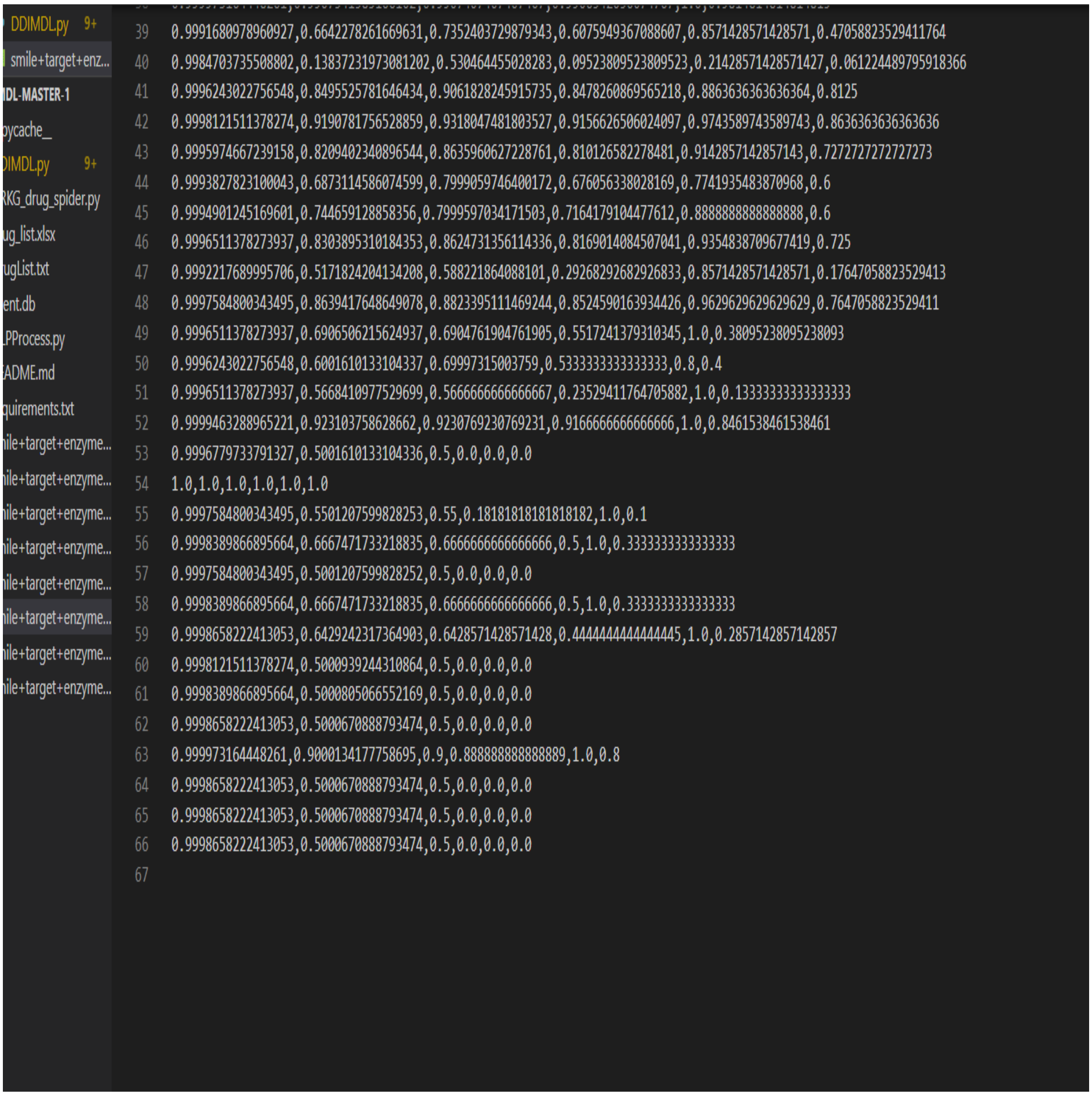
As we know that we have 65 types of different drug-associated events which are we predicting in our work. There fore after the results of drug features further we investigate the performance of our model for each of the events and the matrices score is calculated for each independently by using real values and predicted scores. It is likely that the events with higher frequency can gain better performances. Our models gives AUPR scores greater than 0.5 for the events numbered from 1 to 46 as you can see in the below screenshots except 39.

Below are the screenshots of each and every event's predictions results of the LSTM's model.



```
Accuracy        AUPR            AUC             F1-Score        Precision       Recall
0.9337698583082868,0.8912859657702751,0.9247854348079378,0.8780632411067193,0.8519654841802493,0.9058103975535168
0.947939029626442,0.9109309655990411,0.9434477195450474,0.9014427961796383,0.8708284255987436,0.9342881213142376
0.9514276513525118,0.850256953803041,0.8958682785313979,0.8358425539633593,0.8565055762081785,0.8161530286928799
0.9830936024044654,0.8713122051130828,0.9249841394217766,0.8666948793906051,0.8752136752136752,0.858340318524727
0.9909564190639759,0.8703944923870215,0.9101250854798268,0.8650380456547856,0.9113924050632911,0.823170731707317
0.9937473164448261,0.8971381468825721,0.9364477383476917,0.8948081264108353,0.9150507848568791,0.8754416961130742
0.9888364104765994,0.8090709876883427,0.886914008960083,0.8048780487804877,0.8330097087378641,0.778584392014519
0.992834907685702,0.8793249292636547,0.9387004983645932,0.8775790921595598,0.873972602739726,0.881215469613259
0.9881655216831258,0.6899681637870713,0.8450566987685248,0.6870120652945352,0.6778711484593838,0.6964028776978417
0.9972627737226277,0.9061916689094104,0.9351494940669677,0.9039548022598869,0.9393346379647749,0.8711433756805808
0.9979336625161014,0.8922352475244768,0.9346763908946827,0.891089108910891,0.9130434782608695,0.870165745856353
0.9971017604121941,0.8325326643610971,0.9190399246631238,0.8317757009345794,0.8240740740740741,0.839622641509434
0.9953574495491627,0.5997954052480059,0.7320857849609494,0.5685785536159601,0.7307692307692307,0.46530612244897956
0.9979336625161014,0.8369923212594039,0.8915125768289974,0.8329718004338394,0.8888888888888888,0.7836734693877551
0.997021253756977,0.6743014873632215,0.7735855390767211,0.6498422712933755,0.7984496124031008,0.5478723404255319
0.9985240446543581,0.8260034469080771,0.8816025425826229,0.8208469055374593,0.8873239436619719,0.7636363636363637
0.9996511378273937,0.9598439465175432,0.969284723197149,0.9592476489028212,0.9807692307692307,0.938650306748466
0.9977189781021898,0.7113193696562461,0.8109838728951243,0.6996466431095405,0.7983870967741935,0.6226415094339622
0.9989802490339201,0.887751262055734,0.9648260508553327,0.8855421686746988,0.8448275862068966,0.930379746835443
0.9995437956204379,0.9441931716283063,0.9609715937525154,0.9435215946843853,0.9659863945578231,0.922077922077922
0.99906075568913,0.8581249105481609,0.9164781804800116,0.8571428571428571,0.8823529411764706,0.8333333333333334
0.9992754401030485,0.867203962238216,0.9311976391654557,0.8669950738916257,0.8712871287128713,0.862745098039215
0.9990070845856591,0.8049479878215388,0.8499058228393068,0.7909604519774012,0.9090909090909091,0.7
0.99903392013739,0.8008552624604144,0.8682730798585693,0.7954545454545454,0.864197530864197,0.736842105263157
0.9985508802060971,0.6755284834729133,0.7715777186194377,0.6493506493506493,0.8064516129032258,0.5434782608695652
0.9993291112065265,0.8391304088545393,0.8887947598514255,0.8344370860927153,0.9,0.777777777777778
0.9988729068269644,0.7208028490633687,0.7337527882073018,0.6315789473684211,0.972972972972973,0.467532467532467
0.9991949334478317,0.7914987188360473,0.8731854401391091,0.7887323943661971,0.835820895522388,0.746666666666666
0.99903392013739,0.7365807332665426,0.749986556971546,0.660377358490566,0.9722222222222222,0.5
0.9991949334478317,0.7834411958540138,0.9027565615747378,0.782608695652174,0.7605633802816901,0.805970149253731
0.9993022756547875,0.7855424458915765,0.8592674731182796,0.7796610169491525,0.8518518518518519,0.71875
0.9990875912408759,0.7008964542144086,0.7982661355355505,0.6851851851851853,0.8043478260869565,0.5967741935483871
0.9986582224130528,0.4925954638439285,0.6693436958309321,0.4444444444444444,0.6451612903225806,0.3389830508474576
0.9995169600686991,0.8381054655690617,0.8878638410193388,0.8333333333333334,0.9,0.7758620689655172
0.9997584800343495,0.9194596464283873,0.9385696145650767,0.9174311926605504,0.9615384615384616,0.8771929824561403
0.9992486045513096,0.7371130801358367,0.8543842032352875,0.7358490566037736,0.7647058823529411,0.7090909090909091
0.999973164448261,0.9907541585166102,0.9907407407407407,0.9906542056074767,1.0,0.9814814814814815
```

Figure 4.7: Each and every event evaluation measures values using LSTM's



```
0.9991680978960927,0.6642278261669631,0.7352403729879343,0.6075949367088607,0.8571428571428571,0.47058823529411764
0.9984703735508802,0.13837231973081202,0.530464455028283,0.09523809523809523,0.21428571428571427,0.061224489795918366
0.9996243022756548,0.8495525781646434,0.9061828245915735,0.8478260869565218,0.8863636363636364,0.8125
0.9998121511378274,0.9190781756528859,0.9318047481803527,0.9156626506024097,0.9743589743589743,0.8636363636363636
0.9995974667239158,0.8209402340896544,0.8635960627228761,0.810126582278481,0.9142857142857143,0.7272727272727273
0.9993827823100043,0.6873114586074599,0.7999059746400172,0.676056338028169,0.7741935483870968,0.6
0.9994901245169601,0.744659128858356,0.7999597034171503,0.7164179104477612,0.8888888888888888,0.6
0.9996511378273937,0.8303895310184353,0.8624731356114336,0.8169014084507041,0.9354838709677419,0.725
0.9992217689995706,0.5171824204134208,0.588221864088101,0.29268292682926833,0.8571428571428571,0.17647058823529413
0.9997584800343495,0.8639417648649078,0.8823395111469244,0.8524590163934426,0.9629629629629629,0.7647058823529411
0.9996511378273937,0.6906506215624937,0.6904761904761905,0.5517241379310345,1.0,0.38095238095238093
0.9996243022756548,0.6001610133104337,0.69997315003759,0.5333333333333333,0.8,0.4
0.9996511378273937,0.5668410977529699,0.5666666666666667,0.23529411764705882,1.0,0.13333333333333333
0.9999463288965221,0.923103758628662,0.9230769230769231,0.9166666666666666,1.0,0.8461538461538461
0.9996779733791327,0.5001610133104336,0.5,0.0,0.0,0.0
1.0,1.0,1.0,1.0,1.0,1.0
0.9997584800343495,0.5501207599828253,0.55,0.18181818181818182,1.0,0.1
0.9998389866895664,0.6667471733218835,0.6666666666666666,0.5,1.0,0.3333333333333333
0.9997584800343495,0.5001207599828252,0.5,0.0,0.0,0.0
0.9998389866895664,0.6667471733218835,0.6666666666666666,0.5,1.0,0.3333333333333333
0.9998658222413053,0.6429242317364903,0.6428571428571428,0.4444444444444445,1.0,0.2857142857142857
0.9998121511378274,0.5000939244310864,0.5,0.0,0.0,0.0
0.9998389866895664,0.5000805066552169,0.5,0.0,0.0,0.0
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
0.999973164448261,0.9000134177758695,0.9,0.888888888888889,1.0,0.8
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
0.9998658222413053,0.5000670888793474,0.5,0.0,0.0,0.0
```

Figure 4.8: Each and every event evaluation measures values using LSTM's

ROC graph of LSTM's model are given below

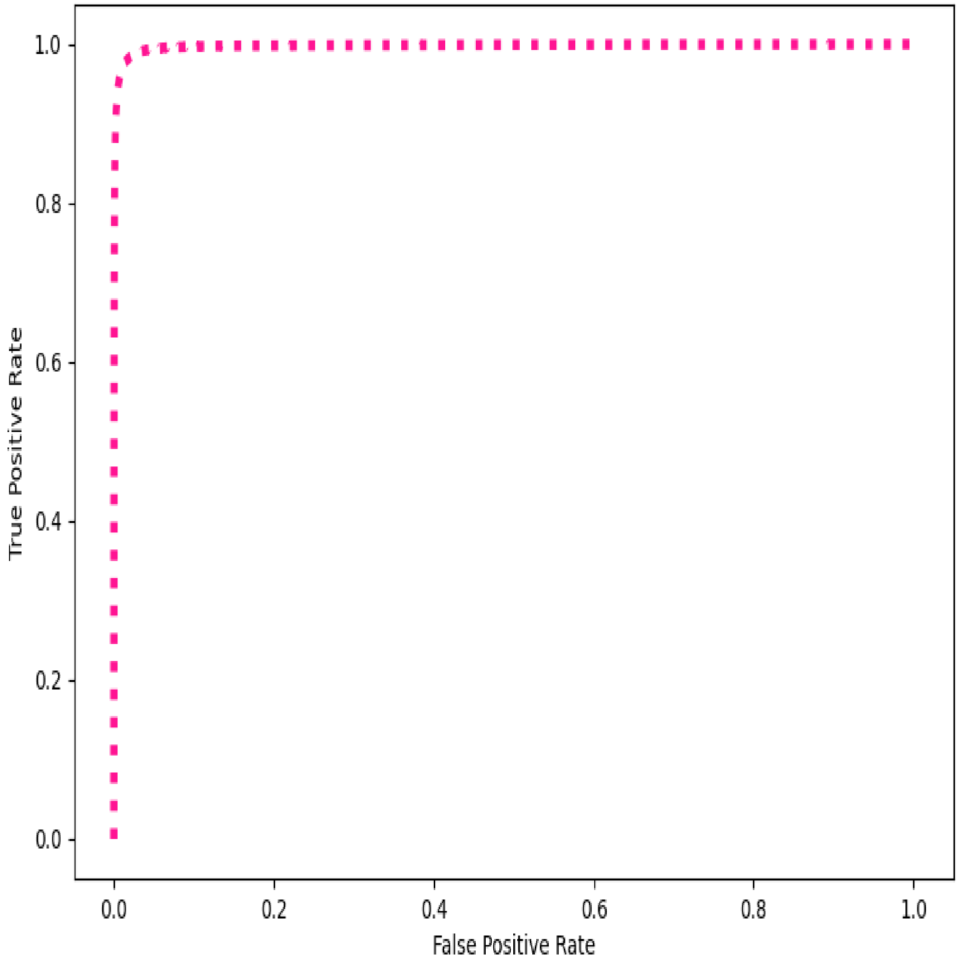


Figure 4.9: LSTM's ROC graph

4.8 Comparing Results

In this section, we have compared our results with previously proposed methods of deep learning and machine learning. To show the robustness of our model MMCNNDDI, MLSTMDDI, and MRNNDDI we compare our model with the following models which are DDIMDL[7], DeepDDI[18] and CNN-DDI[39] and also consider some of the popular classification methods which are random forest (RF), K-nearest neighbor (KNN), logistic regression (LR) and build sub-models as our models like (MMCNNDDI, MLSTMDDI, and MRNNDDI). All the above methods are using the same drug features like targets, enzymes, pathways, and chemical substructure except CNN-DDI and Deep-DDI for the random forest we set the decision tree values to 100 and for KNN we set the neighbor value to 4. As we have seen in the literature review of our work that most of the studies focus on whether two drug interacts with each other or not or some of the studies have only focused on the drugs which are used for specific disease like the authors of [8] predict drug interaction for cancer drugs, and some of the studies used only one feature of drugs like [18] used only chemical substructures for the prediction of DDIs and some of the studies like CNN-DDI [39] used different multiple features from our set of features and used simple CNN for the prediction of DDI associated events we can notice in table 4.11 that our two models MCNNDDI and MLSTMDDI perform better then CNN-DDI in all evaluation matrices and MRNNDDI results are approximately equal to CNN-DDI in-terms of Accuracy and AUC, and AUPR and F1-score of our RNN models are better then CNN-DDI. The results show that using multiple drug features and using sub-models-based deep neural networks perform well than using one feature or using simple structure-based models for the prediction of DDIs associated events. The experiment results of all the methods are shown in table 4.11.

Method	**ACC**	**F1**	**AUC**	**AUPR**
MCNNDDI	0.9000	0.8286	0.9981	0.9478
MLSTMDDI	0.8902	0.7792	0.9978	0.9407
MRNNDDI	0.8866	0.7779	0.9979	0.9395
CNN-DDI	0.8871	0.7496	0.9980	0.9251
DDIMDL	0.8852	0.7585	0.9976	0.9208
DeepDDI	0.8371	0.6848	0.9961	0.8899
RF	0.7775	0.5936	0.9956	0.8349
KNN	0.7214	0.4831	0.9813	0.7716
LR	0.7920	0.5948	0.9960	0.8400

Table 4.11: Different Method Performance

4.9 Applying Drug-Drug Interaction Data set to Drug Target Interaction Existing Method

The drug-drug Interaction data-set used in our study can not be used for drug target interaction purpose because for drug target interaction we need a positive and negative interaction of drug target and protein targets which are not available in the data set we used in our study. And this data set contain only interactions for the Drug to Drug interactions.

Chapter 5

Conclusion

5.1 Conclusion

Recently deep learning techniques are used for the prediction of drug-drug interaction but generally, these studies concentrate on one feature of drugs or whether one drug interacts with another or not and have greatly contributed to understanding the DDIs better. However, it has been divulged that drugs taken together may interact with each other, and unexpected drug-drug interactions (DDIs) may lead to unexpected adverse drug events. Therefore the more DDIs we know, the better we can take effective measures to stop such events In this research study, we use deep learning multi-modal techniques on drug bank database which was created by DDIMDL the following DL models are used CNN's, LSTM's and RNN's for the prediction of drug-drug interaction events. The data set has 572 drugs and their diverse features like chemical substructures (SMILES), enzymes, pathways, and targets, 74528 interactions, and 65 types of drug-drug interaction events. The CNN model achieved an accuracy of 90.00%, LSTM's model achieved an accuracy of 89.02% and the RNN model achieved an accuracy of 88.66%. The CNN model has a 1D CNN input layer with a filter size of 1 and 5 kernel size, 3 dense layers of 1024,512,256 neurons and an output layer of 65 neurons, 1 Flatten layer and have BachNorm and Dropout layer with 0.3 value and use a combination of four drug features for the input of the model. The LSTM's and RNN models have one input layer, three dense layers of 1024, 512, 256 neurons, three BachNorm layers and three dropout layers of value 0.3, and one output layer of 65 neurons. we use four CNN, LSTM's, and RNN sub-models for each and every feature of drugs and then in the last, we combine these sub-models for drug-drug interactions associated events prediction.

5.2 Future Directions

For future work to improve the DDI associated events prediction we will consider using more features of drugs for the prediction of DDI associated events, second enlarge the event data set, and third we will use other deep learning techniques like transformers, berth for improving the performance for the prediction of DDIs associated events.

References

[1] D. M. Qato, J. Wilder, L. P. Schumm, V. Gillet, and G. C. Alexander, “Changes in prescription and over-the-counter medication and dietary supplement use among older adults in the united states, 2005 vs 2011,” *JAMA internal medicine*, vol. 176, no. 4, pp. 473–482, 2016. Cited on p. 1.

[2] K. Baxter and C. L. Preston, *Stockley’s drug interactions*, vol. 495. Pharmaceutical Press London, 2010. Cited on p. 1.

[3] M. Pirmohamed, S. James, S. Meakin, C. Green, A. K. Scott, T. J. Walley, K. Farrar, B. K. Park, and A. M. Breckenridge, “Adverse drug reactions as cause of admission to hospital: prospective analysis of 18 820 patients,” *Bmj*, vol. 329, no. 7456, pp. 15–19, 2004. Cited on p. 1.

[4] R. Miikkulainen, J. Liang, E. Meyerson, A. Rawal, D. Fink, O. Francon, B. Raju, H. Shahrzad, A. Navruzyan, N. Duffy, *et al.*, “Evolving deep neural networks,” in *Artificial intelligence in the age of neural networks and brain computing*, pp. 293–312, Elsevier, 2019. Cited on p. 6.

[5] Y. Bengio, I. Goodfellow, and A. Courville, *Deep learning*, vol. 1. MIT press Massachusetts, USA:, 2017. Cited on p. 6.

[6] X. Hou, J. You, and P. Hu, “Predicting drug-drug interactions using deep neural network,” in *Proceedings of the 2019 11th International Conference on Machine Learning and Computing*, pp. 168–172, 2019. Cited on pp. 6 and 17.

[7] 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*, 2020. Cited on pp. 6, 16, 17, and 54.

[8] S. Suthaharan, “Support vector machine,” in *Machine learning models and algorithms for big data classification*, pp. 207–235, Springer, 2016. Cited on pp. 7, 16, and 55.

[9] Y. Zheng, H. Peng, X. Zhang, Z. Zhao, X. Gao, and J. Li, “Ddi-pulearn: a positive-unlabeled learning method for large-scale prediction of drug-drug interactions,” *BMC bioinformatics*, vol. 20, no. 19, pp. 1–12, 2019. Cited on pp. 7 and 18.

[10] D. Song, Y. Chen, Q. Min, Q. Sun, K. Ye, C. Zhou, S. Yuan, Z. Sun, and J. Liao, “Similarity-based machine learning support vector machine predictor of drug-drug interactions with improved accuracies,” *Journal of clinical pharmacy and therapeutics*, vol. 44, no. 2, pp. 268–275, 2019. Cited on pp. 7 and 18.

[11] Y.-H. Feng, S.-W. Zhang, and J.-Y. Shi, "Dpddi: a deep predictor for drug-drug interactions," *BMC bioinformatics*, vol. 21, no. 1, pp. 1–15, 2020. Cited on pp. 7 and 17.

[12] X. Chen, X. Liu, and J. Wu, "Drug-drug interaction prediction with graph representation learning," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 354–361, IEEE, 2019. Cited on pp. 8 and 18.

[13] C. Sun, P. Xuan, T. Zhang, and Y. Ye, "Graph convolutional autoencoder and generative adversarial network-based method for predicting drug-target interactions," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2020. Cited on p. 8.

[14] H. E. Manoochehri, A. Pillai, and M. Nourani, "Graph convolutional networks for predicting drug-protein interactions," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 1223–1225, IEEE, 2019. Cited on p. 9.

[15] B. Yegnanarayana, *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009. Cited on p. 9.

[16] N. Rohani and C. Eslahchi, "Drug-drug interaction predicting by neural network using integrated similarity," *Scientific reports*, vol. 9, no. 1, pp. 1–11, 2019. Cited on pp. 10 and 17.

[17] N. R. Monteiro, B. Ribeiro, and J. Arrais, "Drug-target interaction prediction: end-to-end deep learning approach," *IEEE/ACM transactions on computational biology and bioinformatics*, 2020. Cited on p. 10.

[18] J. Y. Ryu, H. U. Kim, and S. Y. Lee, "Deep learning improves prediction of drug–drug and drug–food interactions," *Proceedings of the National Academy of Sciences*, vol. 115, no. 18, pp. E4304–E4311, 2018. Cited on pp. 10, 16, 54, and 55.

[19] M. A. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE journal*, vol. 37, no. 2, pp. 233–243, 1991. Cited on p. 11.

[20] G. Lee, C. Park, and J. Ahn, "Novel deep learning model for more accurate prediction of drug-drug interaction effects," *BMC bioinformatics*, vol. 20, no. 1, p. 415, 2019. Cited on pp. 11 and 17.

[21] Y. Dai, C. Guo, W. Guo, and C. Eickhoff, "Drug–drug interaction prediction with wasserstein adversarial autoencoder-based knowledge graph embeddings," *Briefings in Bioinformatics*, 2020. Cited on pp. 11 and 18.

[22] S. Liu, Z. Huang, Y. Qiu, Y.-P. P. Chen, and W. Zhang, "Structural network embedding using multi-modal deep auto-encoders for predicting drug-drug interactions," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 445–450, IEEE, 2019. Cited on pp. 12 and 17.

[23] P. Kim, "Convolutional neural network," in *MATLAB deep learning*, pp. 121–147, Springer, 2017. Cited on pp. 12 and 22.

[24] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, pp. 396–404, 1990. Cited on pp. 12 and 22.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012. Cited on pp. 12 and 22.

[26] P. K. Shukla, P. K. Shukla, P. Sharma, P. Rawat, J. Samar, R. Moriwal, and M. Kaur, "Efficient prediction of drug–drug interaction using deep learning models," *IET Systems Biology*, 2020. Cited on pp. 12 and 17.

[27] M. R. Karim, M. Cochez, J. B. Jares, M. Uddin, O. Beyan, and S. Decker, "Drug-drug interaction prediction based on knowledge graph embeddings and convolutional-lstm network," in *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pp. 113–123, 2019. Cited on pp. 12 and 17.

[28] S. Purkayastha, I. Mondal, S. Sarkar, P. Goyal, and J. K. Pillai, "Drug-drug interactions prediction based on drug embedding and graph auto-encoder," in *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 547–552, IEEE, 2019. Cited on p. 13.

[29] D. Singh Dhami, S. Yan, G. Kunapuli, D. Page, and S. Natarajan, "Predicting drug-drug interactions from heterogeneous data: An embedding approach," *arXiv e-prints*, pp. arXiv–2103, 2021. Cited on p. 13.

[30] L. LiuQiao, L. DuanHong, *et al.*, "Knowledge graph construction techniques," *Journal of computer research and development*, vol. 53, no. 3, p. 582, 2016. Cited on pp. 14 and 23.

[31] 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 (International Joint Conferences on Artificial Intelligence Organization)*, pp. 2739–2745, 2020. Cited on p. 14.

[32] Y. Chen, T. Ma, X. Yang, J. Wang, B. Song, and X. Zeng, "Muffin: multi-scale feature fusion for drug–drug interaction prediction," *Bioinformatics*, 2021. Cited on p. 14.

[33] M. Abdel-Basset, H. Hawash, M. Elhoseny, R. K. Chakrabortty, and M. Ryan, "Deeph-dta: Deep learning for predicting drug-target interactions: A case study of covid-19 drug repurposing," *IEEE Access*, vol. 8, pp. 170433–170451, 2020. Cited on p. 15.

[34] "Drugbank history," 2021. Cited on p. 21.

[35] L. R. Medsker and L. Jain, "Recurrent neural networks," *Design and Applications*, vol. 5, 2001. Cited on p. 24.

[36] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, “A tutorial on the cross-entropy method,” *Annals of operations research*, vol. 134, no. 1, pp. 19–67, 2005. Cited on p. 32.

[37] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014. Cited on p. 32.

[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014. Cited on p. 32.

[39] C. Zhang and T. Zang, “Cnn-ddi: A novel deep learning method for predicting drug-drug interactions,” in *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 1708–1713, IEEE, 2020. Cited on pp. 54 and 55.

Thesis

ORIGINALITY REPORT

14%	9%	9%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	en.wikipedia.org Internet Source	2%
2	doctorpenguin.com Internet Source	1%
3	academic.oup.com Internet Source	1%
4	"Intelligent Computing Theories and Application", Springer Science and Business Media LLC, 2021 Publication	1%
5	Sukannya Purkayastha, Ishani Mondal, Sudeshna Sarkar, Pawan Goyal, Jitesh K Pillai. "Drug-Drug Interactions Prediction Based on Drug Embedding and Graph Auto-Encoder", 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE), 2019 Publication	<1%
6	www.ijcai.org Internet Source	<1%

7	Submitted to Liverpool John Moores University Student Paper	<1%
8	Submitted to The University of Manchester Student Paper	<1%
9	www.narubian.com Internet Source	<1%
10	Arunaben Prahladbhai Gurjar, Shitalben Bhagubhai Patel. "chapter 3 Fundamental Categories of Artificial Neural Networks", IGI Global, 2021 Publication	<1%
11	"Drug Interactions in Infectious Diseases", Springer Science and Business Media LLC, 2011 Publication	<1%
12	S. Shajun Nisha, M. Mohamed Sathik, M. Nagoor Meeral. "Application, algorithm, tools directly related to deep learning", Elsevier BV, 2021 Publication	<1%
13	"Intelligent Computing Theories and Application", Springer Science and Business Media LLC, 2019 Publication	<1%
14	Mohamed Abdel-Basset, Hossam Hawash, Mohamed Elhoseny, Ripon K. Chakrabortty,	<1%

	Michael Ryan. "DeepH-DTA: Deep Learning for Predicting Drug-Target Interactions: A Case Study of COVID-19 Drug Repurposing", IEEE Access, 2020 Publication	
15	link.springer.com Internet Source	<1 %
16	Submitted to Cranfield University Student Paper	<1 %
17	Submitted to General Sir John Kotelawala Defence University Student Paper	<1 %
18	bmcbioinformatics.biomedcentral.com Internet Source	<1 %
19	www.nuventra.com Internet Source	<1 %
20	Submitted to Indian Institute of Technology, Madras Student Paper	<1 %
21	www.iit.upcomillas.es Internet Source	<1 %
22	Prashant Kumar Shukla, Piyush Kumar Shukla, Poonam Sharma, Paresh Rawat, Jashwant Samar, Rahul Moriwal, Manjit Kaur. "Efficient Prediction of Drug-drug interaction using	<1 %

	Deep Learning Models", IET Systems Biology, 2020 Publication	
23	analytixon.com Internet Source	<1 %
24	Submitted to University of Stellenbosch, South Africa Student Paper	<1 %
25	R Naren, J Subhashini. "Comparison of deep learning models for predictive maintenance", IOP Conference Series: Materials Science and Engineering, 2020 Publication	<1 %
26	Submitted to 2642 Student Paper	<1 %
27	Chengcheng Zhang, Tianyi Zang. "CNN-DDI: A novel deep learning method for predicting drug-drug interactions", 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2020 Publication	<1 %
28	Submitted to University of Wollongong Student Paper	<1 %
29	onlinelibrary.wiley.com Internet Source	<1 %
30	www.nature.com Internet Source	<1 %

31	Tao Wang, Changhua Lu, Mei Yang, Feng Hong, Chun Liu. "A hybrid method for heartbeat classification via convolutional neural networks, multilayer perceptrons and focal loss", PeerJ Computer Science, 2020 Publication	<1 %
32	Submitted to University of Ulster Student Paper	<1 %
33	publica.fraunhofer.de Internet Source	<1 %
34	"Artificial Neural Networks and Machine Learning – ICANN 2017", Springer Science and Business Media LLC, 2017 Publication	<1 %
35	Mostafa Elsaadouny, Jan Barowski, Ilona Rolfes. "Extracting the Features of the Shallowly Buried Objects using LeNet Convolutional Network", 2020 14th European Conference on Antennas and Propagation (EuCAP), 2020 Publication	<1 %
36	Submitted to University Computing Centre (SRCE) Croatia Student Paper	<1 %
37	trepo.tuni.fi Internet Source	<1 %

38	www.geeksforgeeks.org Internet Source	<1 %
39	Submitted to Hong Kong Baptist University Student Paper	<1 %
40	umpir.ump.edu.my Internet Source	<1 %
41	Submitted to Laureate Higher Education Group Student Paper	<1 %
42	Mina Samizadeh, Behrouz Minaei-Bidgoli. "Drug-target Interaction Prediction by Metapath2vec Node Embedding in Heterogeneous Network of Interactions", International Journal on Artificial Intelligence Tools, 2020 Publication	<1 %
43	Yang Zhang, Yang Qiu, Yuxin Cui, Shichao Liu, Wen Zhang. "Predicting Drug-drug Interactions using Multi-modal Deep Auto-encoders based Network Embedding and Positive-unlabeled Learning", Methods, 2020 Publication	<1 %
44	Submitted to National College of Ireland Student Paper	<1 %
45	Submitted to Texas A&M University - Commerce Student Paper	<1 %

46	Submitted to Higher Education Commission Pakistan Student Paper	<1%
47	Numan Shafi, Faisal Bukhari, Waheed Iqbal, Khaled Mohamad Almustafa, Muhammad Asif, Zubair Nawaz. "Cleft prediction before birth using deep neural network", Health Informatics Journal, 2020 Publication	<1%
48	Submitted to University College London Student Paper	<1%
49	Submitted to VIT University Student Paper	<1%
50	www.jmir.org Internet Source	<1%
51	Abdulhamit Subasi. "Machine learning techniques", Elsevier BV, 2020 Publication	<1%
52	Chang Sun, Ping Xuan, Tiangang Zhang, Yilin Ye. "Graph convolutional autoencoder and generative adversarial network-based method for predicting drug-target interactions", IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2020 Publication	<1%
53	Submitted to Griffith College Dublin Student Paper	

		<1%
54	aaltodoc.aalto.fi Internet Source	<1%
55	link.umsl.edu Internet Source	<1%
56	orcid.org Internet Source	<1%
57	pastel.archives-ouvertes.fr Internet Source	<1%
58	www.ijitee.org Internet Source	<1%
59	E.C. Malthouse. "Limitations of nonlinear PCA as performed with generic neural networks", IEEE Transactions on Neural Networks, 1998 Publication	<1%
60	Prashant Kumar Shukla, Piyush Kumar Shukla, Poonam Sharma, Paresh Rawat, Jashwant Samar, Rahul Moriwal, Manjit Kaur. "Efficient prediction of drug–drug interaction using deep learning models", IET Systems Biology, 2020 Publication	<1%
61	Qiang Tang, Fulei Nie, Juanjuan Kang, Wei Chen. "mRNALocater: Enhance the prediction accuracy of eukaryotic mRNA subcellular	<1%

	localization by using model fusion strategy", Molecular Therapy, 2021 Publication	
62	Yuanfei Dai, Chenhao Guo, Wenzhong Guo, Carsten Eickhoff. "Drug–drug interaction prediction with Wasserstein Adversarial Autoencoder-based knowledge graph embeddings", Briefings in Bioinformatics, 2020 Publication	<1 %
63	homepages.ecs.vuw.ac.nz Internet Source	<1 %
64	pdfs.semanticscholar.org Internet Source	<1 %
65	pubs.sciepub.com Internet Source	<1 %
66	tel.archives-ouvertes.fr Internet Source	<1 %
67	www.ns2.thinkmind.org Internet Source	<1 %
68	"Advances in Knowledge Discovery and Data Mining", Springer Science and Business Media LLC, 2020 Publication	<1 %
69	Alexander F. B. Carmichael, Deepayan Bhowmik, Johanna Baily, Andrew Brownlow,	<1 %

George J. Gunn, Aaron Reeves. "Ir-Man", Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2020
Publication

70 Ali Nasr, Sydney Marie Bell, Jiayuan He, Rachel l Whittaker, Clark R Dickerson, Ning Jiang, John McPhee. "MuscleNET: Mapping Electromyography to kinematic and dynamic biomechanical variables by machine learning", Cold Spring Harbor Laboratory, 2021
Publication
<1%

71 Feixiong Cheng, Yue Yu, Jie Shen, Lei Yang, Weihua Li, Guixia Liu, Philip W. Lee, Yun Tang. "Classification of Cytochrome P450 Inhibitors and Noninhibitors Using Combined Classifiers", Journal of Chemical Information and Modeling, 2011
Publication
<1%

72 Pathima Nusrath Hameed, Karin Verspoor, Snezana Kusljic, Saman Halgamuge. "Positive-Unlabeled Learning for inferring drug interactions based on heterogeneous attributes", BMC Bioinformatics, 2017
Publication
<1%

73 Shichao Liu, Ziyang Huang, Yang Qiu, Yi-Ping Phoebe Chen, Wen Zhang. "Structural Network Embedding using Multi-modal Deep
<1%

Auto-encoders for Predicting Drug-drug Interactions", 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2019
Publication

74 Xiaoqiang Xu, Ping Xuan, Tiangang Zhang, Bingxu Chen, Nan Sheng. "Inferring drug-target interactions based on random walk and convolutional neural network", IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2021
Publication
<1 %

75 Y. Dehouck, J. M. Kwasigroch, M. Rooman, D. Gilis. "BeAtMuSiC: prediction of changes in protein-protein binding affinity on mutations", Nucleic Acids Research, 2013
Publication
<1 %

76 Yang Qiu, Yang Zhang, Yifan Deng, Shichao Liu, Wen Zhang. "A Comprehensive Review of Computational Methods for Drug-drug Interaction Detection", IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2021
Publication
<1 %

77 depositonce.tu-berlin.de
Internet Source
<1 %

78 downloads.hindawi.com
Internet Source
<1 %

79	es.scribd.com Internet Source	<1 %
80	export.arxiv.org Internet Source	<1 %
81	ijisrt.com Internet Source	<1 %
82	techscience.com Internet Source	<1 %
83	web-tools.uts.edu.au Internet Source	<1 %
84	Heba Ibrahim, A. Abdo, Ahmed M. El Kerdawy, A. Sharaf Eldin. "Signal Detection in Pharmacovigilance: A Review of Informatics-driven Approaches for the Discovery of Drug-Drug Interaction Signals in Different Data Sources", Artificial Intelligence in the Life Sciences, 2021 Publication	<1 %
85	Liheng Gong, Jingjing Yang, Xiao Zhang. "Semi-supervised Breast Histological Image Classification by Node-attention Graph Transfer Network", IEEE Access, 2020 Publication	<1 %
86	Yifan Deng, Xinran Xu, Yang Qiu, Jingbo Xia, Wen Zhang, Shichao Liu. "A multimodal deep	<1 %

learning framework for predicting drug–drug interaction events", Bioinformatics, 2020
Publication

87 Integrated Series in Information Systems, 2016.
Publication
<1%

88 Sabyasachi Chakraborty, Satyabrata Aich, Moon-il Joo, Mangal Sain, Hee-Cheol Kim. "A Multichannel Convolutional Neural Network Architecture for the Detection of the State of Mind Using Physiological Signals from Wearable Devices", Journal of Healthcare Engineering, 2019
Publication
<1%

Exclude quotes	On	Exclude matches	< 5 words
Exclude bibliography	On		