International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I8P214

Volume 70 Issue 8, 140-148, August 2022

Original Article

Drug Side-effects Prediction using Hierarchical Fuzzy Deep Learning for Diagnosing Specific Disease

Nithya B[1], Anitha G[2]

[1]*Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, India.*
[2]*Department of Computer Applications, Karpagam Academy of Higher Education, Coimbatore, India.*

[1] nithyabaalu87@gmail.com

Received: 08 June 2022 Revised: 27 July 2022 Accepted: 06 August 2022 Published: 22 August 2022

Abstract *- In drug discovery, the foremost challenging task is predicting drug-disease correlation using drugs' various indications and side effects on specific diseases for proper diagnosis. To combat this issue, Wasserstein Auto-Encoder with Convolutional Neural Network (WAE-CNN) model was developed, which uses the side effects constraints along with the drugs and patient attributes from the large-scale databases to predict drugs for specific diseases. But, the correlation variability between several drug-disease side effect categories is quite unfair. Few categories are more complex to predict than others. Therefore, this article presents a Hierarchical Fuzzy Deep CNN (HFDCNN)model to predict and recommend drugs for particular diseases considering side effects. First, the database is created by collecting data about patients, diseases, drugs and their side effects. Then, such data are fed to the HFDCNN for prediction. In the HFDCNN model, FDCNN is embedded into the attribute hierarchy. It segregates simple classes using a coarse classifier, whereas fine classifiers differentiate complex classes. In the learning phase, an element-wise pre-learning is supported by global fine-tuning with a multinomial logistic loss normalized by a coarse coherence factor. Also, conditional executions of fine classifiers and layer variable reduction make this HFDCNN more robust for many disease data associated with the drugs and their side effects. Finally, the test results exhibit that the HFDCNN model achieves 95.3%, 97.1% and 98.55% of accuracies in predicting the drugs for Chronic Kidney Disease (CKD), diabetes and heart diseases, correspondingly compared to the classical models.*

Keywords *- Drug-disease correlation, Side effects, WAE-CNN, Attribute hierarchy, Fuzzy DCNN, Multinomial logistic loss, Conditional execution.*

1. Introduction

Data mining is used to find crucial hidden patterns in massive datasets. Machine learning, soft computing, data visualization, classifications, and regression approaches are used to handle some of the difficulties in data mining. This area of study is well-known because it produces superior outcomes. However, it is all centered on a few illness analysis, diagnosis, or prediction approaches employing diverse instruments and methodologies. Few studies have been undertaken on illness drug prediction [1-3]. There are two sorts of medications used in the medical field: generic and brand-name pharmaceuticals. Generic medications are comparable to brand-name pharmaceuticals in terms of dose, intended use, dosage forms, side effects, safeguards, and weight. Alternatively, they have similar pharmacological effects as their brand-name counterparts. Patients will be given a specific medicine based on the severity of their diseases [4-5].

Over the past decades, many investigations have been done using data mining algorithms to predict the appropriate drugs for specific diseases [6-7]. This viewpoint suggests the drug prediction for diabetes, asthma and chronic heart diseases [8]. For this purpose, many drugs and patient characteristics were initially retrieved. Then, Hidden Markov Models (HMM) were utilized to uncover hidden correlations between features. After that, feature statistical values were computed and merged.

Moreover, Artificial Neural Network (ANN) was used to process the detected characteristics. To predict drugs for illnesses with fewer side effects, the ANN employed higher weight values assigned to each node in the hidden layers during the training stage. The ANN-based classifier suffers from excessive computational complexity when faced with many features.

Physical traits and illness side effects of patients alter throughout time. As a result, attribute values change over time. A strategy [9] was carried out for selecting more relevant features in optimally split time intervals. The Krill Herd (KH) method [10] was used to calculate the temporal interval for feature selection. The KH algorithm was developed to choose the most discriminative features from a

collection of medication and patient parameters such as age, weight, height, and so on. This KH algorithm is composed of many krill, and their time-dependent location is influenced by three key factors: movement driven by the presence of other individuals, foraging activity and random diffusion. The best attributes from the best time segments were used in ANN to predict medications for various disorders. But, the KH method performs poorly on huge datasets.

As a result, the deep learning technique with WAE [11] was presented to improve prediction accuracy with fine-tuned drug and patient data. WAE fine-tunes drug attributes such as adverse effects and biological, chemical and phenotypic qualities. WAE employs AE [12] and Feed-Forward Neural Network (FFNN) as learning algorithms for recreating input features. Also, WAE was learned to encode input into a feature space using a random set of weights. The attributes were then recreated using a Generative Weights (GW) set. First, the GWs are largely derived from the encoder's unfolded weights and later modified.

The encoder reduces the dimensionality of the data by translating it to a hidden representation, and the decoder sequentially transfers the reduced features to the input. Because no labelled features were required in the training procedure, the method was unsupervised. WAE then employed the loss function to increase the stability of the regenerated values. Finally, the regenerated attributes were sent into CNN, which predicts drugs for various illnesses. However, the heterogeneity in correlation across multiple drug-disease side effect categories was quite unfair. Also, few categories were more difficult to predict than others.

Hence in this manuscript, the HFDCNN model is proposed to predict and recommend drugs for particular diseases with the consideration of side effects. Initially, the database is built by gathering information about patients, diseases, treatments and their side effects. The data is then sent into the HFDCNN for prediction. FDCNN is incorporated into the attribute hierarchy of the HFDCNN model. It uses a coarse classifier to separate basic classes, while fine classifiers discriminate complicated classifications. During the learning phase, global fine-tuning with a multinomial logistic loss normalized by a coarse coherence factor is used to facilitate element-wise pre-learning. Further, conditional executions of fine classifiers and layer variable reduction make this HFDCNN more resilient for many illness data connected with medications and side effects. Thus, the appropriate drugs for specific diseases without side effects are predicted efficiently.

The remaining sections of this paper are organized as follows: Section II presents the studies related to drug-disease prediction. Section III discusses the HFDCNN model, and Section IV illustrates its efficiency for predicting drugs. Section V summarizes the entire study and provides the future scope.

2. Literature Survey

Hunta et al. [13] presented an enzyme and transporter protein Integrated Action Crossing (IAC) technique to predict non-communicable disease's drug-drug correlation depending on pharmacokinetic strategy. In this technique, the drug-drug correlation data were collected from the web. Then, the novel characteristics were created and applied to the different machine learning algorithms to generate the prediction model. But this technique was not suitable for large-scale databases.

Ibrahim & Thangamani [14] designed a new enhanced Singular Value Decomposition (SVD) scheme to reduce the dimensionality of drug-disease data. First, an integrated model for Hepatocellular Carcinoma (HCC) subordinate was developed using the Multi-source Bat Algorithm-based Random Walk (MBARW) to differentiate novel drugs and diseases. Then, a drug-drug similarity grid and disease-drug similarity chain were created based on the multi-source random stroll in gene-gene weighted correlation order. Moreover, all drugs in the drug-drug similarity chain and disease-drug bipartite graph chain were ranked by considering the known drugs for HCC. But, it was not suitable for the database containing many genes.

Jiang et al. [26] developed a Sigmoid Kernel and CNN (SKCNN) to train new attributes efficiently, defining drug-disease correlations using its hidden layer. Initially, the similarity metric of drugs was created by sigmoid drug similarity, drug structural similarity and that of disease utilizing disease sigmoid similarity and disease semantic similarity. According to the fused similarities of drugs and diseases, the SKCNN was utilized for training hidden interpretations for all drug-disease pairs whose tags were predicted by the random forest categorizer. But, the training data was comparatively inadequate, influencing the prediction accuracy.

Peng et al. [16] developed a learning-based technique depending on feature interpretation training and deep learning called DTI-CNN to predict drug-target correlations. Initially, the Jaccard similarity coefficient extracted the related attributes of drugs and proteins from heterogeneous networks and restarted the random walk scheme. After that, a denoising AE was used to minimize the dimension and detect the important attributes. According to these attributes, the CNN was created to predict the correlation between drugs and proteins. But, it needs more related data and network design to contain more sophisticated input networks.

Yang et al. [17] designed a Multiple Kernel-based Dual Graph RLS (MKDGRLS) to predict possible drug-disease correlations. Initially, the multiple kernels of drugs and disease spaces were determined correspondingly. After that, multiple kernels and associated Laplacian normalization terms were used to build MKDGRLS. Moreover, the

objective factor of MKDGRLS was solved by the alternated least squares to predict drug-disease correlations. But, it has many matrix functions and a high running time. Also, its efficiency was poor for small databases.

Jarada et al. [18] developed a Similarity Network Fusion and Collective Variational AE (SNF-CVAE) model to predict new drug-disease correlations using drug-associated similarity data and known drug-disease correlations. This model integrated similarity measures, similarity choice, SNF and CVAE to perform a non-linear analysis and enhance the drug-disease correlation prediction. But, it has a high training period due to the huge amount of mixtures of learnable hyperparameters.

Ding et al. [19-20] designed a Multi-view Graph Regularized Link Propagation (MvGRLP) framework to predict novel drug-target correlations. This framework merged complementary data among various views in drug and target space. Also, an iterative scheme was applied to resolve the objective function. But, it needs to define structural correlations among entities to increase the prediction efficiency.

A.G.Hari Narayanan[21] recognises the signs and symptoms of various skin diseases. After a large no of Classifiers, the voting classifier makes predictions about the image's content.

Sunil Pandey[22] comprises CNN of stacks of different layers which perform feature engineering and training or classification computations on the inputs. The design of CNN algorithms for high-performance distributed and parallel computing architectures assume significance.

Khaled Mohamad Alumstafa[23] K-nearest neighbor, Decision tree and SVM classifiers show the performance of the selected classification to classify the best or predict the heart disease cases.

C K Gomathy[24] Naïve Bayes Supervised Machine Learning algorithm predicts the disease. The probability of the disease is calculated using the Naïve Bayes algorithm.

Rayan Alanazi[25] Convolutional Neural Network(CNN) is for the prediction of the disease, and K-nearest neighbor (KNN) is used for calculating the distance to find the match that is generated in the data set for the prediction of diseases.

3. Proposed Methodology

This part briefly describes the HFDCNN model for drug side-effects prediction. First, several data regarding patients and drugs are gathered from the different websites, which involves many attributes like drug name, ingredients, size, shape, capsule shell, weight, surface region, dissolution period, predisposition of medema, side effect, drug mode (i.e., tablet, injection, capsules), drug price, drug category, dosage, brand name, drug feedback, patient age, sex, weight and height. Such details are gathered at multiple time slots, e.g., every week, 15 days, or month. Then, those attributes are fed to the HFDCNN as input to predict drugs and their side effects for certain diseases. The training and testing process of HFDCNN for drug prediction is briefly explained below.

The notations considered in this article are the following: a database comprises $\{x_i, y_i\}$, where x_i refers to the input data and y_i refers to the label. There are C fine attributes of data in the database $\{S_j^f\}_{j=1}^{C}$. An attribute hierarchy is learned with K coarse attributes (disease-related characteristics) $\{S_k^c\}_{k=1}^{K}$.

3.1. HFDCNN Structure

The HFDCNN is developed to replicate the architecture of attribute hierarchy, where fine attributes are split into coarse attributes. It executes end-to-end classification as depicted in Fig. 1. It has 4 major units: shared layers, a single coarse attribute layer$B$, multiple fine attribute layers $\{F^k\}_{k=1}^{K}$ and a single probabilistic averaging unit.

The shared layers are utilized to obtain the most relevant attributes. The configuration of shared layers is assigned to be similar to the prior layers in the fundamental block net.

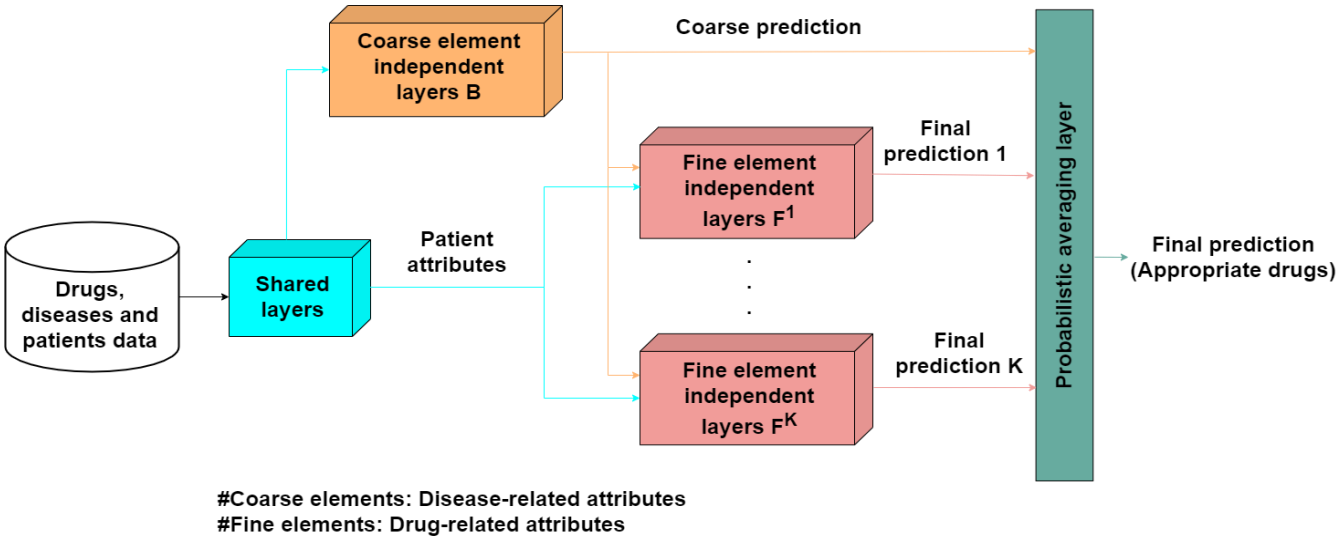

Fig. 1 Structure of HFDCNN for Drug Prediction based on 2-stage Attribute Hierarchy

The top of the structure in Figure 1 has independent layers of coarse attribute layer$B$, which reutilizes the setting of rear layers from the fundamental block FDCNN and creates a fine prediction $\{B_{ij}^{f}\}_{j=1}^{C}$ for a data x_i. To generate a prediction $\{B_{ik}\}_{k=1}^{K}$ over coarse attributes, a fine-to-coarse aggregation layer is added, which combines fine predictions into coarse ones while mapping from fine attributes to coarse ones$P$: $[1, C] \mapsto [1, K]$ is provided. The coarse attribute probabilities are useful for 2 reasons: (i) they are utilized as a weight to merge the predictions created by fine attribute layers, and (ii) if threshold, then they allow conditional executions of fine attribute layers whose related coarse probabilities are adequately high.

The bottom of the structure in Figure 1 has independent layers of a group of fine categorizers $\{F^k\}_{k=1}^{K}$, all of which create fine attribute predictions. Because all fine layers succeed only in categorizing a tiny group of classes, they generate a fine prediction over a limited group of classes. The probabilities of another fine attribute missing in the limited group are perfectly assigned to 0. The layer settings are essentially replicated from the fundamental block FDCNN, except that the number of filters in the last categorization layer is assigned to be the number of the limited group rather than the entire classes.

Layers in both the coarse and fine attribute layers are shared. The reasons for this are 3-fold: (1). It is demonstrated that prior layers in FDCNN respond to class-agnostic attributes like patient's age, gender, etc., whereas rear layers capture more class-specific attributes like drug size, drug shape, disease name, side effects, etc. Because patient attributes are relevant for coarse and fine prediction processes, the prior layers are allowed to share with the coarse and fine layers;(2). It minimizes both the overall floating-point functions and the memory usage of system implementation; and (3). It minimizes the quantity of HFDCNN variables, which is essential for effective learning of HFDCNN.

The right side of the structure in Figure 1 is the probabilistic averaging layer, which gets both the fine and coarse attribute predictions, as well as generates a weighted mean as the final prediction:

$$p(x_i) = \frac{\sum_{k=1}^{K} B_{ik} p_k(x_i)}{\sum_{k=1}^{K} B_{ik}} \quad (1)$$

In Eq (1), B_{ik} refers to the probability of coarse attribute$k$ for the data x_i predicted by B and $p_k(x_i)$ refers to the fine attribute prediction created by F^k. It emphasized that coarse and fine attribute layers reutilize the layer settings from the fundamental block HFDCNN. This adaptable structure enables the selection of the appropriate FDCNN structure as the fundamental block according to the processes.

3.2. Attribute hierarchy Training

The main objective of creating an attribute hierarchy is to aggregate similar fine attributes into similar coarse attributes, allowing a specific fine attribute categorizer to be learned. A top-down scheme is applied to train the hierarchy from the learning data.

A held-out data group is randomly sampled with balanced class distribution from the learning data. The remaining learning data is utilized for training a fundamental block net. A confusion matrix F is obtained by analyzing the net on the held-out group. A distance matrix D is determined as $D = 1 - F$, and its diagonal values are assigned 0. Then, D is converted as$D = 0.5 * (D + D^T)$. The input D_{ij} evaluates how simple it is to distinguish classes i and j.

Spectral grouping is applied on D to group fine attributes into Kcoarse attributes. The outcome is a 2-stage attribute hierarchy defining a multiple-to-single mapping $P^d: [1, C] \mapsto [1, K]$ from fine to coarse attributes. In this study, the coarse attributes are time-dependent.

3.2.1. Overlapping coarse attributes

When there are time-dependent coarse attributes, the entire categorization strongly relies on the coarse attribute categorizer. When the data is fed to an imperfect fine attribute categorizer, the error cannot be resolved since the probability of the labelled data is automatically assigned to 0. By eliminating the correlation variability limit among coarse attributes, the HFDCNN can become less reliant on the coarse attribute categorizer.

So, additional fine attributes are added to the coarse attributes. For a specific fine categorizer F^k, it is preferred to include such fine attributes $\{j\}$. That is expected to be miscategorized into the coarse attribute$k$. So, the likelihood $u^k(j)$ is estimated that the data in the fine attribute$j$ is miscategorized into a coarse attribute$k$ on the held-out group as:

$$u^k(j) = \frac{1}{|S_j^f|} \sum_{i \in S_j^f} B_{ik}^d \quad (2)$$

In Eq. (2), B_{ik}^d denotes the coarse attribute probability, which is attained by combining fine attribute probabilities $\{B_{ij}^f\}_j$ based on the mapping $P^d: B_{ik}^d = \sum_{j|P^d(j)=k} B_{ij}^f$. The likelihood $u^k(j)$ is thresholded by the parametric factor $u_t = (\gamma K)^{-1}$ and the limited group $S_k^c$is included in each fine attribute$\{j\}$ such that $u^k(j) \geq u_t$. Observe that all branching layer provides a complete group prediction if $u_t = 0$ and a time-dependent group prediction if $u_t = 1$. With overlapping coarse attributes, the attribute hierarchy mapping P^d is expanded to be a multiple-to-multiple mapping $P^o$and so the coarse attribute predictions are modified as $B_{ik}^o = \sum_{j|k \in P^o(j)} B_{ij}$. Observe the total of $\{B_{ik}^o\}_{k=1}^K$ exceeds 1, and so L_1 regularization is performed.

3.3. HFDCNN Learning

The quantity of variables in rear layers rises relative to the number of coarse attributes if fine attribute layers are included in the HFDCNN. It raises the learning difficulty and the possibility of over-fitting for an equal quantity of learning data. However, the learning data (attributes) in the stochastic gradient descent mini-batch are probabilistically given to distinct fine attribute layers. It needs a greater mini-batch to guarantee that a huge amount of learning data calculates variable gradients in the fine attribute layers. A huge learning mini-batch raises the learning memory usage and reduces the learning speed. As a result, the HFDCNN learning is decomposed into many stages rather than learning the entire HFDCNN from scratch, as summarized in *Algorithm 1*.

Algorithm 1: HFDCNN learning
Function HFDCNN LEARNING
Pre-learn HFDCNN;
Initialize coarse attribute layer;
Pre-learn fine attribute layers;
Adjust the entire HFDCNN
End Function

3.3.1. Pre-learning HFDCNN

Both the coarse and fine attribute layers are pre-learned simultaneously.

Initialization of the coarse attribute layer

Initially, a fundamental block of HFDCNN (F^p) is pre-learned by utilizing the learning data. Because both the prior and rear layers in the coarse attribute layer are similar to the layers in the fundamental block HFDCNN, the weights of F^p are replicated into the coarse attribute layer for the initialization task.

Pre-learning the rear layers of fine attribute layers

Fine attribute layers $\{F^k\}_k$ are separately pre-learned. All F^k must concentrate on categorizing fine attributes within the coarse attributeS_k^c. So, the pre-learning of all F^k utilizes only data $\{x_i | i \in S_k^c\}$ from the coarse attributeS_k^c. The shared prior layers are previously initialized and kept constant in this phase. For all F^k, each rear layer is initialized, excluding the final convolutional layer by replicating the trained variables from the pre-learned net F^p.

Fine-tuning HFDCNN

Once both coarse and fine attribute layers are efficiently pre-learned, the entire HFDCNN is adjusted. After learning the attribute hierarchy and the related mapping P^oEvery fine attribute layer concentrates on categorizing a predetermined subgroup of fine attributes.

During fine-tuning, the ambiguity of coarse attribute predicted by the coarse attribute layer must be kept reliable with those related to the fine attribute layers. Therefore, a coarse attribute reliability term is added to normalize the classical multinomial logistic error.

Coarse attribute reliability

The trained fine-to-coarse attribute mapping $P: [1, C] \mapsto [1, K]$ gives a method to characterize the desired coarse attribute distribution $\{t_k\}$. Particularly, t_k is assigned to be the percentage of each learning data within the coarse attributeS_k^c under the hypothesis, the distribution over coarse attributes across the learning set is nearby within the learning mini-batch.

$$t_k = \frac{\sum_{j|k \in P(j)} |S_j|}{\sum_{k'=1}^K \sum_{j|k' \in P(j)} |S_j|}, \forall k \in [1, K] \quad (3)$$

The final error function utilized for adjusting the HFDCNN is defined as:

$$E = -\frac{1}{n}\sum_{i=1}^{n}\log\left(p_{y_i}\right) + \frac{\lambda}{2}\sum_{k=1}^{K}\left(t_k - \frac{1}{n}\sum_{i=1}^{n}B_{ik}\right)^2 \quad (4)$$

In Eq. (4), n denotes the size of the learning mini-batch, λ denotes the normalization constant and is assigned to 20.

3.4. HFDCNN Testing

Because fine attribute layers are added to the HFDCNN, the number of variables, memory usage and execution period in rear layers, each scale linearly in the number of coarse attributes. To guarantee HFDCNN is adaptable for large-scale drug-disease prediction, conditional execution and layer variable reduction methods are developed.

3.4.1. Conditional execution

During testing, for considered data, it is not essential to analyze each fine attribute categorizer because most of them contain irrelevant weights B_{ik} as in Eq. (1). Their contributions to the final prediction are omitted. Conditional executions of the top-weighted fine layers will speed up the HFDCNN categorization. So, B_{ik} is thresholded by a parametric factor $B_t = (\beta K)^{-1}$ and B_{ik} is reassigned to 0 if $B_{ik} < B_t$. Such fine attribute categorizers with $B_{ik} = 0$ are not analyzed.

3.4.2. Variable reduction

In HFDCNN, the number of variables in rear layers of fine attribute categorizers raises linearly in the number of coarse attributes. So, the layer variables are reduced during the test phase to minimize memory usage.

Particularly, the product quantization scheme [20] is adopted to reduce the variable matrix $W \in R^{m\times n}$ by initially splitting it horizontally into sections of size s, i.e. $W = \left[W^1, \ldots, W^{(n/s)}\right]$. After that, k-means clustering is utilized to group the rows in $W^i, \forall i \in [1, (n/s)]$. By only accumulating the closest group indices in an 8-bit integer matrix $I \in R^{m\times(n/s)}$ and group midpoints in a single-accuracy floating number matrix$C \in R^{k\times n}$, a reduction factor $\frac{32mn}{32kn+8m^{n}/_{s}}$ is achieved. The hyperparameters for variable reduction are (s, k).

Thus, this HFDCNN is efficiently trained and tested for predicting the drugs for a specific disease without any side effects.

4. Experimental Analysis

In this section, the efficiency of the HFDCNN is analyzed by implementing it in Java 8 software tools. For this experiment, different details of patients and drugs are gathered from various websites, hospitals, etc. Also, the sample dataset containing the brand-name and generic drugs utilized for CKD, diabetes and heart diseases are collected according to their drug name, class, dosage, side effects and the ingredients present in the drugs. The efficiency of the HFDCNN is compared with the different existing models: KH-ANN [10], WAE-CNN [11], SKCNN [26], DTI-CNN [16], MKDGRLS [17], SNF-CVAE [18] and MvGRLP [19] regarding the following metrics:

- **Accuracy:** It is the fraction of accurate prediction of drugs for specific diseases over the total number of data tested.
- $$Acc = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

  TP (True Positive) is a result of the given positive classes being categorized as themselves. TN (True Negative) results in the negative classes being categorized as themselves. FP (False Positive) is a result of inexactly classifying positive classes as negative. FN (False Negative) results in the negative classes being inexactly classified as positive.
- **Precision is the ratio of** predicted drugs and diseases at TP and FP rates.

$$Recall = \frac{TP}{TP+FP} \quad (6)$$

- **Recall:** It is the ratio of exactly predicted drugs and diseases at TP and FN rates.

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

Table 1 provides the accuracy values for the existing and proposed drug prediction models executed on various disease databases.

Table 1. Comparison of Accuracy

Disease categories	MvGRLP	MKDGRLS	KH-ANN	SKCNN	DTI-CNN	SNF-CVAE	WAE-CNN	HFDCNN
	Accuracy (%)							
CKD	83.3	85.9	87	89	91.6	92.8	94	95.3
Diabetes	87.7	90.5	92	92.6	93.8	95	96	97.1
Heart diseases	89.5	91	93	93.8	94.7	95.6	97	98.5

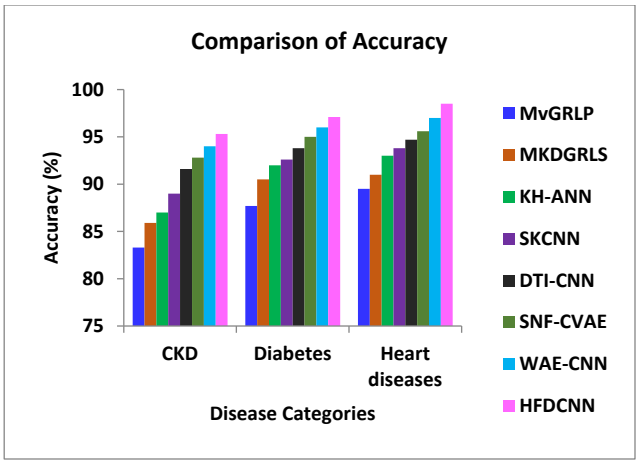

Fig. 2 Accuracy vs Different Disease Categories

Figure 2 depicts the graphical representation of accuracy achieved for various drug prediction models using the different drugs and patient data related to CKD, diabetes and heart diseases. It addresses that the HFDCNN attains the maximum accuracy to predict appropriate drugs for patients and diagnose the diseases properly with less or no side effects.

For the CKD, the accuracy of predicting drugs using HFDCNN is 14.41% greater than the MvGRLP, 10.94% greater than the MKDGRLS, 9.54% greater than the KH-ANN, 7.08% greater than the SKCNN, 4.04% greater than the DTI-CNN, 2.69% greater than the SNF-CVAE and 1.38% greater than the WAE-CNN. For diabetes disease, the accuracy of predicting drugs by the HFDCNN is 10.72% better than the MvGRLP, 7.29% better than the MKDGRLS, 5.54% better than the KH-ANN, 4.86% better than the SKCNN, 3.52% better than the DTI-CNN, 2.21% better than the SNF-CVAE and 1.15% better than the WAE-CNN. For heart diseases, the accuracy of predicting drugs using HFDCNN is 10.06% larger than the MvGRLP, 8.24% larger than the MKDGRLS, 5.91% larger than the KH-ANN, 5.01% larger than the SKCNN, 4.01% larger than the DTI-CNN, 3.03% larger than the SNF-CVAE and 1.55% larger than the WAE-CNN. It is because of training the coarse and fine attributes hierarchically and independently.

Table 2 provides the precision values for the different existing and proposed drug prediction models executed on various disease databases.

Table 2. Comparison of Precision

Disease categories	MvGRLP	MKDGRLS	KH-ANN	SKCNN	DTI-CNN	SNF-CVAE	WAE-CNN	HFDCNN
	Precision							
CKD	0.810	0.832	0.854	0.877	0.906	0.911	0.924	0.945
Diabetes	0.850	0.869	0.887	0.900	0.915	0.930	0.941	0.959
Heart diseases	0.847	0.864	0.875	0.896	0.933	0.948	0.963	0.981

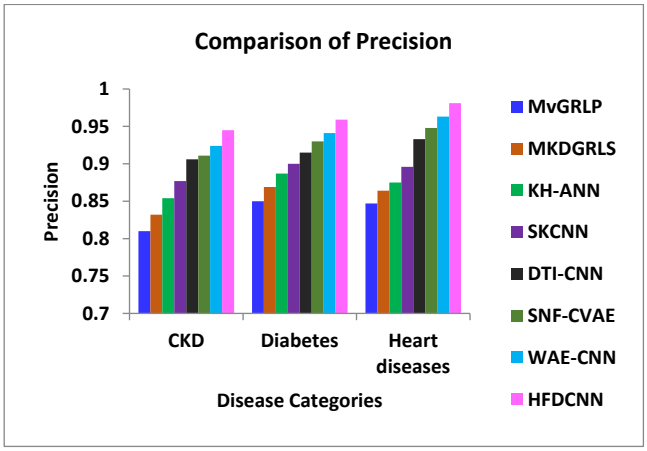

Fig. 3 Precision vs Different Disease Categories

Figure 3 portrays the graphical representation of precision values achieved for various drug prediction models using the different drugs and patients' data related to CKD, diabetes and heart diseases. It observes that the HFDCNN attains the maximum precision to predict suitable drugs for patients and diagnose the diseases accurately with less or no side effects. For the CKD, the precision of predicting drugs using HFDCNN is 16.67% greater than the MvGRLP, 13.58% greater than the MKDGRLS, 10.66% greater than the KH-ANN, 7.75% greater than the SKCNN, 4.3% greater than the DTI-CNN, 3.73% greater than the SNF-CVAE and 2.27% greater than the WAE-CNN. For diabetes disease, the precision of predicting drugs by the HFDCNN is 12.82% better than the MvGRLP, 10.36% better than the MKDGRLS, 8.12% better than the KH-ANN, 6.56% better than the SKCNN, 4.81% better than the DTI-CNN, 3.12% better than the SNF-CVAE and 1.91% better than the WAE-CNN. For heart diseases, the precision of predicting drugs using HFDCNN is 15.82% larger than the MvGRLP, 13.54% larger than the MKDGRLS, 12.11% larger than the KH-ANN, 9.49% larger than the SKCNN, 5.14% larger than the DTI-CNN, 3.48% larger than the SNF-CVAE and 1.87% larger than the WAE-CNN.

Table 3 provides the recall values for the different existing and proposed drug prediction models executed on various disease databases.

Table 3. Comparison of Recall

Disease categories	MvGRLP	MKDGRLS	KH-ANN	SKCNN	DTI-CNN	SNF-CVAE	WAE-CNN	HFDCNN
	Recall							
CKD	0.809	0.82	0.84	0.87	0.918	0.926	0.94	0.953
Diabetes	0.857	0.868	0.89	0.91	0.925	0.94	0.95	0.961
Heart diseases	0.84	0.855	0.87	0.905	0.95	0.968	0.98	0.986

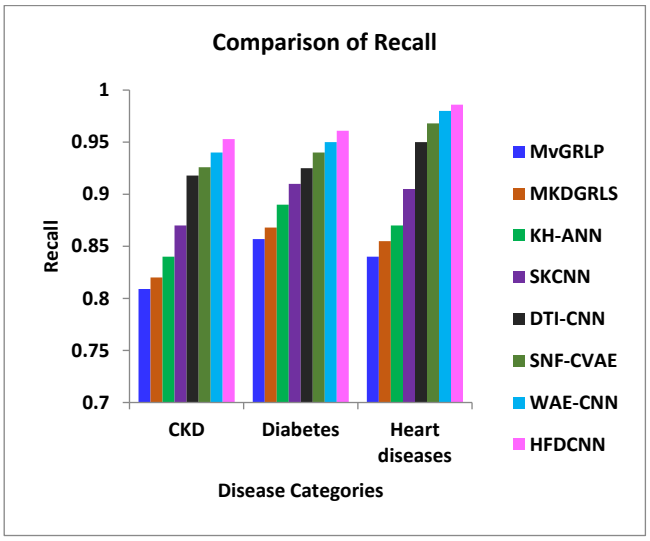

Fig. 4 Recall vs Different Disease Categories

Fig. 4 illustrates the graphical representation of recall values achieved for various drug prediction models using the different drugs and patient data related to CKD, diabetes and heart diseases. It observes that the HFDCNN attains the maximum recall to predict suitable drugs for patients and diagnose the diseases accurately with less or no side effects. For the CKD, the recall of predicting drugs using HFDCNN is 17.8% greater than the MvGRLP, 16.22% greater than the MKDGRLS, 13.45% greater than the KH-ANN, 9.54% greater than the SKCNN, 3.81% greater than the DTI-CNN, 2.92% greater than the SNF-CVAE and 1.38% greater than the WAE-CNN.

For diabetes disease, the recall of predicting drugs by the HFDCNN is 12.14% better than the MvGRLP, 10.71% better than the MKDGRLS, 7.98% better than the KH-ANN, 5.6% better than the SKCNN, 3.89% better than the DTI-CNN, 2.23% better than the SNF-CVAE and 1.16% better than the WAE-CNN. For heart diseases, the recall of predicting drugs using HFDCNN is 17.38% larger than the MvGRLP, 15.32% larger than the MKDGRLS, 13.33% larger than the KH-ANN, 8.95% larger than the SKCNN, 3.79% larger than the DTI-CNN, 1.86% larger than the SNF-CVAE and 0.61% larger than the WAE-CNN.

5. Conclusion

This study developed the HFDCNN model for drug prediction and recommendation for specified illnesses. Primarily, the database was prepared by gathering information regarding patients, diseases, drugs and their side effects. Afterward, the created database was provided to the HFDCNN for the prediction process. This HFDCNN model was designed by embedding the FDCNN into the attribute hierarchy, which splits uncomplicated classes by the coarse categorizer and complicated classes by the fine categorizers. During the HFDCNN learning, an element-wise pre-learning was adopted based on the global adjustment with a multinomial logistic error normalized through the coarse coherence factor.

Additionally, the HFDCNN was enhanced by the conditional executions of fine categorizers and the reduction of layer variables for huge databases. By learning the HFDCNN, the appropriate drugs for certain diseases were predicted according to the patient's characteristics. At last, the test outcomes proved that the HFDCNN model has 95.3%, 97.1% and 98.5% accuracy for predicting the drugs for CKD, diabetes and heart diseases, respectively, compared to the other existing models.

References

[1] Agatonovic- Kustrin S, & Morton D, "Data Mining in Drug Discovery and Design," In *Artificial Neural Network for Drug Design, Delivery and Disposition*, Academic Press, pp. 181-193, 2016.

[2] Chittora, A., &Mekala A. M, "Discovery of Drug and Medicine Using Data Mining Techniques," *Research Journal of Pharmacy and Technology*, vol. 10, no. 12, pp. 4147-4151, 2017.

[3] Yosipof, A., Guedes, R. C., & García-Sosa A. T, "Data Mining and Machine Learning Models for Predicting Drug Likeness and Their Disease or Organ Category," *Frontiers in Chemistry*, vol. 6, pp. 1-11, 2018.

[4] Dunne, S., Shannon, B., Dunne, C., & Cullen W, "A Review of the Differences and Similarities Between Generic Drugs and their Originator Counterparts, Including Economic Benefits Associated with Usage of Generic Medicines, using Ireland as a Case Study," *BMC Pharmacology and Toxicology*, vol. 14, no. 1, pp. 1-19, 2013.

[5] Tamargo, J., Le Heuzey, J. Y., & Mabo P, "Narrow Therapeutic Index Drugs: A Clinical Pharmacological Consideration to Flecainide," *European Journal of Clinical Pharmacology*, vol. 71, no. 5, pp. 549-567, 2015.

[6] Jiménez, R., Anupol, J., Cajal, B., & Gervilla E, "Data Mining Techniques for Drug Use Research," *Addictive Behaviors Reports*, vol. 8, no. 128-135, 2018.

[7] Bagherian, M., Sabeti, E., Wang, K., Sartor, M. A., Nikolovska-Coleska, Z., & Najarian K, "Machine Learning Approaches and Databases for Prediction of Drug–Target Interaction: A Survey Paper," *Briefings in Bioinformatics*, vol. 22, no. 1, pp. 247-269, 2021.

[8] Nithya, B., & Anitha G, "Prediction of Drugs for Diseases with Side Effect and Patient Physical Attributes".

[9] Nithya, B., & Anitha G, "The Optimal Time Slot Selection and Feature Selection for the Prediction of Drugs for Diseases," *International Journal of Nonlinear Analysis and Applications*, vol. 12, pp. 2137-2151, 2021.

[10] Gandomi, A. H., & Alavi A. H, "Krill Herd: A New Bio-Inspired Optimization Algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831-4845, 2012.

[11] Mondal, A. K., Asnani, H., Singla, P., & Prathosh A. P, "FlexAE: Flexibly Learning Latent Priors for Wasserstein Auto-Encoders," In *Uncertainty in Artificial Intelligence*, pp. 525-535, 2021.

[12] Yousefi-Azar, M., Varadharajan, V., Hamey, L., & Tupakula U, "Autoencoder-Based Feature Learning for Cyber Security Applications," In *IEEE International Joint Conference on Neural Networks*, pp. 3854-3861, 2017.

[13] Hunta, S., Yooyativong, T., & Aunsri N, "A Novel Integrated Action Crossing Method for Drug-Drug Interaction Prediction in Non-Communicable Diseases," *Computer Methods and Programs in Biomedicine*, vol. 163, pp. 183-193, 2018.

[14] Ibrahim, S. J. A., & Thangamani M, "Enhanced Singular Value Decomposition for Prediction of Drugs and Diseases with Hepatocellular Carcinoma Based on Multi-Source Bat Algorithm Based Random Walk," *Measurement*, vol. 141, 176-183, 2019.

[15] HemanthSomasekar and Dr. KavyaNaveen, "A System For Identifying Synthetic Images Using Lstm: A Deep Learning Approach," *International Journal of Computer Trends and Technology*, vol. 69, no. 2, pp. 64-67, 2021. *Crossref*, https://doi.org/10.14445/22312803/IJCTT-V69I2P110

[16] Peng, J., Li, J., & Shang X, "A Learning-Based Method for Drug-Target Interaction Prediction Based on Feature Representation Learning and Deep Neural Network," *BMC Bioinformatics*, vol. 21, no. 13, pp. 1-13, 2020.

[17] Yang, H., Ding, Y., Tang, J., & Guo F, "Drug–Disease Associations Prediction Via Multiple Kernel-Based Dual Graph Regularized Least Squares," *Applied Soft Computing*, vol. 112, pp. 1-14, 2021.

[18] Jarada, T. N., Rokne, J. G., & Alhajj R, "SNF–CVAE: Computational Method to Predict Drug–Disease Interactions using Similarity Network Fusion and Collective Variational Autoencoder," *Knowledge-Based Systems*, vol. 212, pp. 1-23, 2021.

[19] Ding, Y., Tang, J., & Guo F, "Identification of Drug-Target Interactions Via Multi-View Graph Regularized Link Propagation Model," *Neurocomputing*, vol. 461, pp. 618-631, 2021.

[20] Jegou, H., Douze, M., & Schmid C, "Product Quantization for Nearest Neighbor Search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 117-128, 2010.

[21] A.G.Hari Narayanan, Dr J Amar Pratap Singh, "Skin Disease Ensemble Classification using Transfer Learning and Voting Classifier," *International Journal of Engineering Trends and Technologies*, vol. 12, pp. 287-293, 2021.

[22] Sunil Pandey,Naresh Kumar Nagwani,ShrishVerma, "Analysis and Design of High Performance Deep Learning Algorithms: Convolutional Neural Networks," *International Journal of Engineering Trends and Technologies*, vol. 6, pp. 216-224, 2021.

[23] Khaled Mohamad Almustafa, "Prediction of Heart Disease and Classifiers' Sensitivity Analysis," *BMC Bioinformatics*, vol. 21, pp. 275, 2020.

[24] C K Gomathy, "The Prediction of Disease using Machine Learning," *International Journal of Scientific Research in Engineering and Management(IJSREM)*, vol. 5, no. 10, 2021.

[25] RayanAlanazi, "Identification and Prediction of Chronic Diseases using Machine Learning Approach," *Journal of Healthcare Engineering*, vol. 2022, 2022

[26] Jiang, H. J., You, Z. H., & Huang Y. A, "Predicting Drug−Disease Associations Via Sigmoid Kernel-Based Convolutional Neural Networks, *Journal of Translational Medicine*, vol. 17, no. 1, pp. 1-11, 2019.