ISSN: 2306-6091

International Journal of Pharmaceuticals and Health care Research (IJPHR)

IJPHR |Vol.14 | Issue 2 | Apr - Jun -2026

www.ijphr.com

DOI: https://doi.org/10.61096/ijphr.v14.iss2.2026.201-218

Evaluation of Artificial Intelligence and Algorithms in Drug Discovery

[1]Dr. Ebenezer David*, [2]Dr. Mohamed Halith, [3]Dr. Balramchowbay, [4]Dr. Ram Mohan,

K. Abdullah, R. Abinaya, R. Abinisha, K. Abishek, B. Adhithyan

[1]Professor and Head, Department of Pharmacology, Dhanalakshmi Srinivasan College of Pharmacy, Perambalur, Tamil Nadu, India.

[2]Professor and Principal Department of Pharmaceutics, Dhanalakshmi Srinivasan College of Pharmacy, Perambalur, Tamil Nadu, India.

[3]Professor and Director, Department of clinical pharmacology, National cancer center, Singapore.

[4]Professor and Principal, Department of Biotechnology and Pharmacogenomics, ABN AND PRR college of science Kovvur, Rajamudhry, Andhra Pradesh.

[5]Student, Department of Pharmacology, Dhanalakshmi Srinivasan College of Pharmacy, Perambalur, Tamil Nadu, India.

Corresponding author: Dr. Ebenezer David
Email: ed_pharmacologist@aol.com

Published on: 13.04.2026

Published by: Futuristic Publications

Abstract: The integration of Artificial Intelligence (AI) and advanced algorithms has significantly transformed the drug discovery process by enabling faster, cost-effective, and data-driven approaches. Traditional drug development is time-consuming, expensive, and characterized by high failure rates, necessitating innovative solutions. AI techniques, including machine learning, deep learning, natural language processing, and reinforcement learning, play a crucial role in various stages of drug discovery such as target identification, virtual screening, lead optimization, and clinical development. Algorithms like Naïve Bayes, Support Vector Machines, Decision Trees, Random Forest, and boosting methods enhance predictive accuracy and decision-making capabilities. Additionally, deep learning models such as Convolutional Neural Networks, Recurrent Neural Networks, and Graph Neural Networks facilitate the analysis of complex biological and chemical data. Despite these advancements, challenges such as data quality, model interpretability, and integration into existing pharmaceutical frameworks remain significant barriers. Overall, AI-driven methodologies hold great promise in improving efficiency, reducing costs, and accelerating the development of safe and effective therapeutic agents.

Keywords: Artificial Intelligence; Drug Discovery; Machine Learning; Deep Learning; Algorithms; Virtual Screening; Target Identification; Molecular Fingerprinting; Reinforcement Learning; Clinical Development

INTRODUCTION:

Artificial intelligence (AI) and algorithms play a significant role in modern drug discovery by enabling faster and more efficient analysis of large biological and chemical datasets. AI techniques like machine learning and deep learning help in finding targets, screening virtual compounds, improving drug candidates, and predicting how safe and effective a drug might be. By reducing time, cost, and failure rates, AI-based algorithms have transformed traditional drug discovery into a data-driven and intelligent process.

DRUG DISCOVERY:

Discovering a new drug is a difficult and costly process, and many attempts end in failure. Creating a new medicine usually costs more than $2.5 billion and often takes more than ten years to finish. Only a small number of drug candidates enter clinical trials and receive regulatory approval. Even though many efforts are made, only about 2.01% of drug development projects end up creating a drug that can be sold in the market. These challenges show how important it is to find new ways that can speed up and increase the chances of success in discovering new drugs.

Traditional methods of finding and creating new drugs face major challenges that lead to high expenses, long time periods, and many unsuccessful attempts. These challenges encompass the laborious and time-consuming process of identifying potential drug targets; the resource-intensive nature of high-throughput screening for lead compounds; the iterative and expensive process of optimizing lead compounds to improve efficacy, selectivity, and safety; and the difficulties in designing and conducting efficient clinical trials, such as patient recruitment, data collection, and analysis. These bottlenecks significantly affect the efficiency and success rate of drug development, hindering the timely delivery of innovative therapies to patients.

Figure: 1

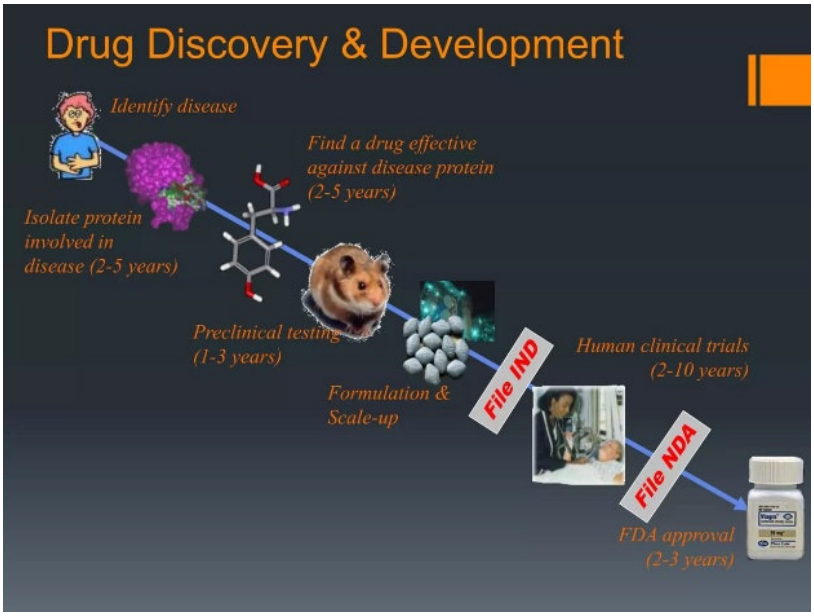

Figure: 2

ARTIFICIAL INTELLIGENCE:

Artificial Intelligence, or AI, is a type of technology that helps machines and computers do things that usually need human intelligence. It allows systems to learn from data, spot patterns, and make decisions to tackle difficult problems. It is utilized in healthcare, finance, e-commerce, and transportation, providing personalized recommendations and facilitating self-driving cars.

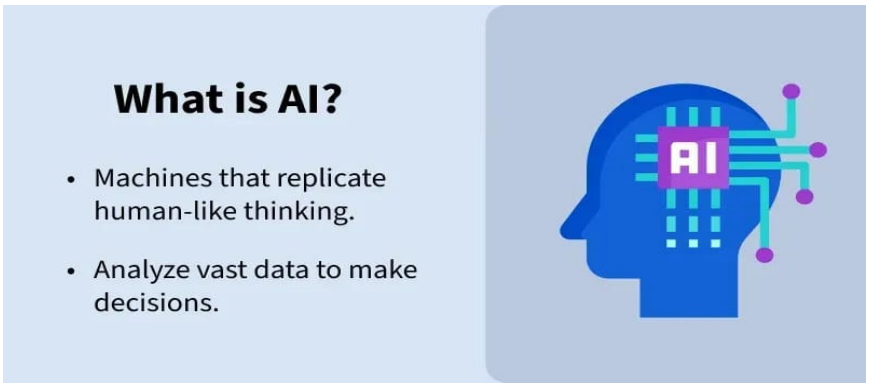

Figure: 3

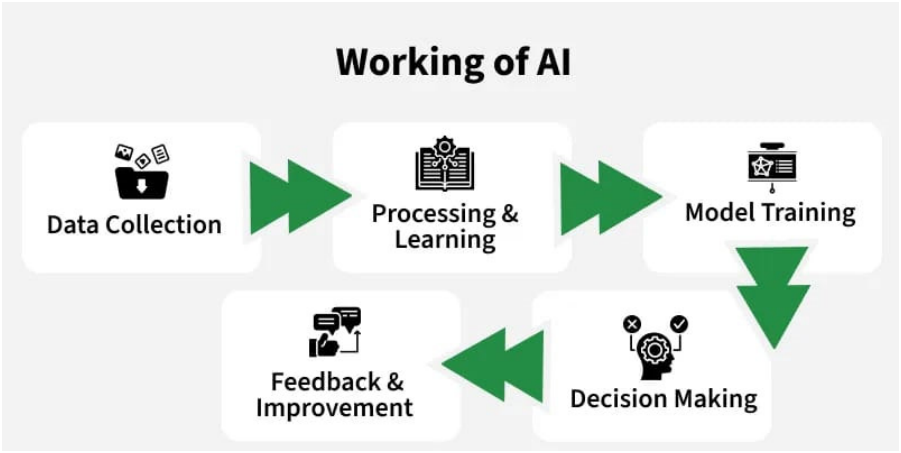

Figure: 4

IMPORTANCE OF ARTIFICIAL INTELLIGENCE:

- Artificial Intelligence provides many benefits that help increase productivity, make better decisions, and improve overall experience for users in various fields.
- AI helps make tasks faster by doing repetitive work automatically, makes processes run more smoothly, and helps avoid mistakes that people might make. This allows us to save time and focus on more strategic and innovative tasks.
- AI can handle and look at a lot of data, which helps people and companies make better choices based on real information. This helps in areas such as healthcare, finance, and retail. Personalization helps tailor experiences by analyzing user preferences and customizing recommendations. This improves how happy users are, as you can see on websites like Netflix, Amazon, and social media platforms.
- AI is always available, 24 hours a day, 7 days a week. Unlike people who need to rest, AI doesn't get tired, so it's perfect for jobs that need to run all the time, like helping customers, watching over security, and gathering data. It is good at looking at a lot of data and finding patterns that people might miss. This ability is useful in fields such as healthcare for diagnosing illnesses, spotting fraud, and understanding market trends.

ALGORITHMS:

Algorithms are a set of finite, well-defined steps or instructions designed to solve a problem or perform a computation. It can also be defined as a procedure for solving a mathematical or computational problem in a finite number of steps, often involving repetitive or recursive operations.

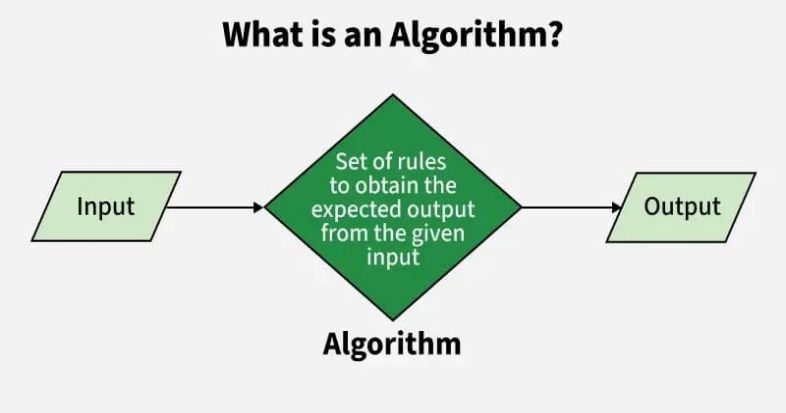

Figure: 5

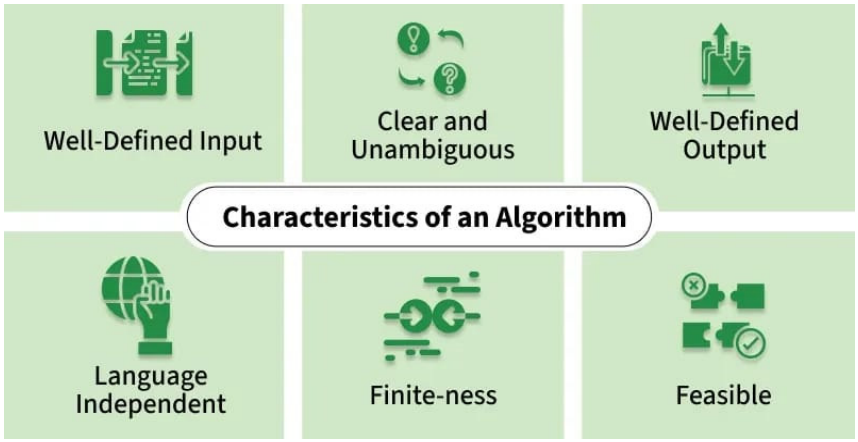

Figure: 6

CLASSIFICATION OF ALGORITHMS:

1. Machine Learning Algorithm,
2. Deep Learning Algorithm,
3. Natural Language Processing,
4. Molecular Fingerprinting,
5. Reinforcement Learning and Evolutionary Algorithms.

MACHINE LEARNING ALGORITHMS:

Machine learning (ML), a subset of artificial intelligence (AI), enables systems to learn from data without requiring explicit programming. It has played a significant role in recent innovations across various fields. For example, machine learning algorithms have been used to help doctors identify illnesses and forecast how patients might fare based on their past medical records and daily habits. Using machine learning and artificial intelligence in drug discovery has changed the pharmaceutical industry in a big way, making the process of creating new treatments faster and more effective. Machine learning algorithms have been used in different areas of drug discovery, like studying genes, proteins, and RNA, to find important pathways and markers linked to various diseases. This has facilitated the prioritization and validation of promising drug targets.

TYPES:

1. Supervised learning algorithms
2. Unsupervised learning algorithms

SUPERVISED LEARNING ALGORITHMS:

This means the model is trained using data that has been marked or labeled. Once the model is trained, it can be used to classify or predict new examples that it hasn't seen before. SL can be further divided into classification and regression. Classification uses special tools called algorithms to guess which category something belongs to, like if a patient is ill or not. The algorithm gets some data that has been labeled, and it uses that to learn how to sort new data into the right group from the ones it already knows about. For example, given a dataset containing patients records, a classifier can learn to distinguish between sick and healthy classes and accurately assign them to the respective categories. Meanwhile, regression uses an algorithm to learn how to predict a continuous number, like how much a house might cost. The model uses a set of examples that have been marked with the correct answers to learn how to guess the right number for new examples it hasn't seen before.

1) PROBABILISTIC MODELS:

a) NAÏVE BAYES:

NB is a probabilistic algorithm that calculates the likelihood that a given input will belong to a specific class based on prior probabilities and conditional probabilities. This algorithm works under the idea that all the features are separate from each other. Gaussian, Bernoulli, and Multinomial Naive Bayes are three different kinds of Naive Bayes classifiers. The Gaussian Naive Bayes assumes that the input features follow a Gaussian distribution. The Bernoulli Naive Bayes classifier is used when the input features are binary, like whether a certain attribute is present or not, and the Multinomial Naive Bayes classifier is used when the input features are counts of different categories. NB is most helpful when there are a lot of input features and the dataset has many missing values. The learning process involves training the classifier using a labelled dataset, where the classifier learns the prior probabilities and conditional probabilities from the training data.

Bayes Theorem can be represented as,

$$P(Y|x_1, x_2, \dots, x_n) = \frac{P(Y)P(x_1,x_2,\dots x_n|Y)}{P(x_1,x_2,\dots,x_n)} \quad (1)$$

where Y represents the class variable, $x_1, x_2, \dots, x_n$are the independent variables, $P\ (Y\ )$, $P\ (x_i|Y\ )$, $P\ (x_1, x_2, \dots, x_n|\ Y\ )$, and $P\ (x_1, x_2, \dots, x_n)$ represents the prior probability of class Y , the conditional probability of feature x_i, the joint probability of all the features, and the probability of all the features occurring together, respectively.

The following formula is employed for predicting the class:

$$\hat{y} = \underset{Y_k}{\text{argmax}}\ \text{P}\ (Y_k) \prod_{i=1}^{n} P(x_i\ |Y_k) \quad (2)$$

Algorithm 1 Naïve Bayes Algorithm

```
procedure NaiveBayesClassifier(X, Y)
    Input: X, Y
    Output: Predicted class labels for a given sample
    Compute prior probabilities P(Y_k) for each class Y_k
    for each feature x_i in X do
        Compute conditional probabilities P(x_i|Y_k)
    end for
    for each input sample x = {x_1, x_2, ..., x_n} in X do
        Initialize maxProb ← 0
        Initialize predictedClass ← null
        for each class Y_k do
            prob ← P(Y_k)
            for each feature x_i in x do
                prob ← prob × P(x_i|Y_k)
            end for
            if prob > maxProb then
                maxProb ← prob
                predictedClass ← Y_k
            end if
        end for
        Output predictedClass for x
    end for
end procedure
```

Figure: 7

b) BAYESIAN NETWORK:

The Bayesian network, sometimes called the Bayes network, is a type of model that uses probability to help make decisions and reason about uncertain situations. A Bayesian network shows how different random variables are connected and depend on each other through a diagram that has arrows and no loops. Bayesian networks are particularly effective for modeling complex systems with many variables and complex interactions among them. Bayesian networks are good at making accurate predictions even when the data is not complete, which is one of their key strengths. Bayes networks are easy to update when new data comes in, which lets them change and adapt to new situations over time. Bayesian networks are valuable classifiers in various fields because they can handle incomplete data and still produce useful predictions.

To perform probabilistic inference in a Bayesian network, we can use Bayes' rule, which can be written as follows:

$$P(X_i|evidence) = \frac{P(evidence|X_i)P(X_i)}{\sum_{X_i} P(evidence|X_i)P(X_i|parents(X_i))} \quad (3)$$

where $P(evidence|X_i)$ is the likelihood of the evidence given X_i , $P(X_i)$ is the prior probability of X_i , and the denominator is the normalization constant.

2) LINEAR CLASSIFIERS:

Linear classifiers are a type of SL algorithm that creates a linear boundary between classes to classify the input data.

a) LOGISTIC REGRESSION:

LR is a statistical method used for predicting binary outcomes. This is a kind of statistical method used to predict outcomes that can only be one of two possible options, like yes or no, which are represented as 0 or 1. This tool helps predict the chance of a specific event happening based on a group of input factors. Furthermore, it involves calculating the gradient of the cost function, which measures the prediction error across the input data, and adjusting the parameters in the direction that minimizes this error. The learning rate alpha determines how big the changes to the parameters are, and this process is carried out for a set number of times.

Algorithm 2 Logistic Regression

procedure LOGISTICREGRESSIONTRAINING(X, Y, α, iterations)
Input:
X - feature set (input variables)
Y - target class labels (output variable)
α - learning rate
iterations - number of training iterations
Output: Model parameters Θ
Initialize model parameters Θ with zeros
for $i \leftarrow 1$ to iterations **do**
Compute the hypothesis $h_\Theta(x) = \frac{1}{1+e^{-\Theta^T X}}$
Compute the cost $J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)} \log(h_\Theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\Theta(x^{(i)}))]$
Compute the gradient $\nabla J(\Theta) = \frac{1}{m} X^T(h_\Theta(X) - Y)$
Update parameters $\Theta \leftarrow \Theta - \alpha \nabla J(\Theta)$
end for
return Θ
end procedure

Figure: 8

b) LINEAR DISCRIMINANT ANALYSIS:

Linear Discriminant Analysis (LDA) is often used to reduce the number of features in data and to classify things into different groups, especially when the groups are clearly separate and the conditions required for LDA are satisfied. The purpose of LDA is to reduce the data to a smaller number of dimensions while keeping as much information that helps distinguish between different classes as possible. If X is the $N \times D$ matrix representing N samples with D features, and y be the corresponding class labels. The class means μk and within-class scatter matrices S_w are calculated as follows:

$$mu_k = \frac{1}{N_k}\sum_{i=1}^{N} x_i, \tag{4}$$

$$S_\omega = \sum_{k=1}^{K}\sum_{i=1}^{N_k}(x_i - \mu_k)(x_i - \mu_k)^T \tag{5}$$

where N_K is the number of samples in class k.

The between-class scatter matrix S_b is given by:

$$S_b = \sum_{k=1}^{K} N_k\,(\mu_k - \mu)(\mu_k - \mu)^T \tag{6}$$

where μ is the overall mean of all classes. Furthermore, the LDA aims to find the projection matrix W that maximizes the ratio of between-class scatter to within-class scatter, given by:

$$max_w \frac{tr(W^T S_b W)}{tr(W^T S_\omega W)} \tag{7}$$

where tr (·) denotes the trace of a matrix. LDA assumes that the covariance matrix and that the classes are linearly separable. When these assumptions hold, LDA provides a simple and effective classification method.

3) NONLINEAR CLASSIFIERS:

a) SUPPORT VECTOR MACHINES:

Support Vector Machines can be used for both straight-line and curved classification problems, which makes them very useful for many different real-world situations. The algorithm uses the idea of a hyperplane, which acts as a line or surface that divides data points into separate groups. The objective of SVMs is to identify the hyperplane that maximizes the margin. This method of maximizing the margin helps SVMs perform well with new data and deal with information that isn't easily separated by a straight line.

The optimization task can be formulated as:

$$\min_{w,b} \frac{1}{2}||w||^2 \quad s.t.\, y_i(w^N x_i + b) \geq 1 \quad \forall i = 1, \ldots, n \tag{8}$$

where w represents the weight vector, b is the bias term, and $||w||$ is the Euclidean norm of w. Meanwhile, w and b can be computed using:

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i \quad \text{and} \quad b = y_k - \sum_{i=1}^{n} \alpha_i y_i x_i^T x_k \tag{9}$$

where k is any support vector with $\alpha k > 0$.

b) k – NEAREST NEIGHBOURS:

k-NN is a straightforward and easy-to-understand classification method that does not assume any specific form for the data. The primary concept of k-NN is to categorize a new data point by determining the majority class among its closest neighbours in the feature space. The distance metric is important for finding the neighbours. While Euclidean distance is often used, k-NN can also use other types of distance measures that help understand the data's structure more effectively.

Assuming x_i represents a data point in the feature space, and x_j denotes its nearest neighbour. The non-Euclidean distance $d\ (x_i, x_j)$ between x_i and x_j can be calculated using various metrics such as Manhattan distance, Minkowski distance, or Mahala Nobis distance, depending on the characteristics of the data. For example, the Manhattan distance between two data points x_i and x_j in n-dimensional space is defined as:

$$d\ (x_i, x_j) = \sum_{k=1}^{n} |x_{ik} - x_{jk}| \tag{10}$$

where x_{ik} and x_{jk} are the k-th features of x_i and x_j , respectively. This distance metric is particularly useful when dealing with high dimensional data.

4) DECISION TREES:

DT is a popular machine learning method that builds a tree structure showing different decisions and the results they can lead to. The tree consists of nodes, branches, and leaves that symbolize decisions, potential results of those decisions, and the ultimate outcome of the decision-making process. DT models are known for being easy to understand because their structure is like a tree, which lets researchers see and explain how decisions are made. This is very important when checking models to get approval from regulators. DT models are also flexible and can work with both numbers and categories, which makes them good for jobs like predicting how drugs interact with targets, sorting chemicals by their biological effects, and finding important parts of molecules that help drugs work better.

a) CLASSIFICATION AND REGRESSION TREES:

The classification and regression tree (CART) algorithm is used for both classification and regression tasks. The algorithm selects the input variable that provides the best split. The best split is defined as the one that maximizes the difference between the parent node's impurity and the weighted impurity of the child nodes. The impurity of a node shows how mixed up the class or target values are inside that node. The impurity measure used in classification tasks is usually Gini impurity.

Algorithm 3 CART Algorithm

1: **Input:** X, Y, maximum depth of the tree (max_depth), minimum size of a node (min_size)
2: **Output:** CART-based DT model
3: **Procedure**
4: Initialize tree
5: Split the root node based on the best-split point
6: Recursively split child nodes
7: Stop if maximum depth or minimum node size is reached
8: Prune the tree
9: **return** DT model
10: **End Procedure**

Figure: 9

b) ITERATIVE DOCHOTOMISER 3:

It selects features to split the data based on the Information Gain (IG) criterion, with the goal of maximizing the reduction in entropy. The process continues recursively until all data is perfectly classified.

Algorithm 4 ID3 Algorithm

```
Input: X, Y
Output: ID3-based DT model
Procedure
if All samples are in the same class then
    return leaf node with class label
end if
if No features left to split then
    return leaf node with the most common class label
end if
Select feature with highest IG as node
Split dataset based on feature values
Recursively apply ID3 to each subset
return DT model
End Procedure
```

Figure: 10

c) C4.5

The C4.5 decision tree is a more advanced version of the ID3 algorithm. Some of the improvements include its ability to handle both continuous and categorical data in classification tasks. C4.5 uses the information gain ratio to pick the best feature at each step and also has ways to deal with missing data and prunes the tree to prevent it from becoming too complex. It deals with continuous attributes by automatically setting threshold values for splitting. In the C4.5 algorithm, the data is iteratively partitioned until each subset is pure or predefined stopping criteria are met, resulting in decision tree models that can effectively classify new instances.

Algorithm 5 C4.5 Algorithm

```
Input: X, Y, stopping criteria (thresholds)
Output: DT model
Procedure
if All examples are in the same class or other stopping criteria met then
    return leaf node with class label
end if
Select feature with highest information gain ratio
Split dataset based on feature values or threshold for continuous data
Handle missing values
Recursively apply the splitting and selection steps (steps 5-9) to each subset
Apply pruning to reduce tree size and complexity
return C4.5 model
End Procedure
```

Figure: 11

d) RANDOM FOREST:

RF is a type of algorithm that brings together several decision tree models to build a stronger and more accurate model. This algorithm is known for its ability to model complex data and provide reliable predictions.

The algorithm functions by making lots of DTs and combining their predictions. Each DT is trained using a random part of the input data and a random selection of the available features. This random approach reduces overfitting and enhances the final model's generalization performance. The algorithm uses the average of the predictions from the base models to make predictions for new, unseen data. The RF algorithm can handle both categorical and continuous data, and unlike a single decision tree model, the final prediction from N trees is calculated like this:

$$\text{H(N(x))} = \underset{j}{\text{argmax}} \sum_{k=1}^{K} 1(h_k(x) = j), for\ j = 1, \dots, C \quad (11)$$

5) BOOSTING:

Boosting is an ensemble technique that iteratively improves the performance of individual models by giving more weight to misclassified instances in subsequent iterations, ensuring the models learn from their mistakes and make better predictions.

a) XG BOOST:

The XG Boost is a version of gradient boosting that has been made faster and more efficient. The XG Boost algorithm creates several base models one after another, and each new model tries to fix the mistakes made by the previous models. The final prediction is then calculated by summing up all the base model's predictions. The XG Boost's objective function is represented as:

$$\mathcal{L}(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y_i}) + \sum_{k=1}^{K} \Omega(f_k) \quad (12)$$

where θ denotes the model parameters, n is the number of instances, y_i and $\hat{y_i}$ represents the actual label and predicted label of the i the instance,$l(y_i, \hat{y_i})$ is the loss function, K is the number of trees, and $\Omega(f_k)$ is the regularization term. At every iteration, XG Boost fits a new model to correct the errors of the previous model, which is achieved by minimizing the objective function using the gradient descent algorithm.

b) ADA BOOST:

AdaBoost is a type of boosting method that, like other boosting techniques, works by combining several weak classifiers into a single strong classifier. It focuses on misclassified data points and iteratively trains the model to enhance its performance. The algorithm begins by assigning equal weights to each training sample. Let D_1 be the weight vector for the first round of training, where $D_{1,i} = 1\backslash n$, at every iteration, a weak classifier $h_j(x)$ is trained using the weighted training data. The weak classifier is trained to minimize the weighted error rate E_j:

$$E_j = \sum_{i=1}^{n} D_{J,I} I(y_i \neq h_j(x_i)) \quad (13)$$

where y_i and x_i are the true label and feature vector of the instance i, and I is the indicator function. The weight α_j of the weak classifier is then computed as

$$\alpha_j = \frac{1}{2} \ln \frac{1-E_J}{E_j} \quad (14)$$

The weights of the training samples are updated based on the performance of the weak classifier. The weight of data point i in the $(j+1)\ th$ round, $D_{j+1,i}$ is computed as

$$D_{j+1,i} = \frac{D_{j,i} \exp(-\alpha_j y_i h_j(x_i))}{Z_J} \quad (15)$$

where Z_j is the normalization factor and $Z_j = \sum_{i=1}^{n} D_{j,i} \exp(-\alpha_j\ y_i\ h_j(x_i))$. The purpose of the weight update is to give more weight to the misclassified instances. Assuming we have N total number of weak classifiers, the final classification model is the weighted combination of the base models:

$$\text{H(x)} = \sin(\sum_{j=1}^{N} \alpha_j\ h_j(x)) \quad (16)$$

AdaBoost is helpful in finding new medicines because it brings together many weak predictors to create a stronger one, which makes the predictions more accurate. Furthermore, AdaBoost works well when the main problem is having too few examples of one class, which is often the case in drug discovery datasets where there are usually more inactive compounds than active ones. By paying more attention to the examples that were classified incorrectly, AdaBoost improves the model's ability to identify instances from the minority class, like rare but possibly very effective compounds. This feature makes AdaBoost very useful in the early stages of drug development, where finding new active compounds is very important.

c) CAT BOOST:

Categorical Boosting (CAT BOOST) is a widely used ensemble learning technique that combines multiple weak learners to create a strong ensemble classifier. Cat Boost is famous for dealing with categorical features without requiring one-hot encoding. It can also deal with missing information and has tools for working with text data, which makes it good for processing natural language. It uses a symmetric tree structure for its DTs, which helps to reduce overfitting. Cat Boost can deal with imbalanced data by using a special type of objective function that considers how the different classes are distributed in the data. The objective function is defined as follows:

$$F = \sum_{i=1}^{n} L(y_i, \hat{y}_i) + \sum_{i=1}^{N} \Omega(f_i) \quad (17)$$

$L(y_i, \hat{y}_i)$ is the loss function, f_i is the i-th tree, and $\Omega(f_i)$ denotes the regularization term that penalizes complex trees. This algorithm uses the logarithmic loss function, and its regularization term is the L2 regularization, which is represented as follows:

$$L(y_i, \hat{y}_i) = -(y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)) \quad (18)$$

$$\Omega(f_i) = \frac{1}{2}\lambda ||w||^2 \quad (19)$$

where w represents the vector of weights and λ is the regularization parameter.

UNSUPERVISED LEARNING ALGORITHMS:

In this ML type, the algorithm learns and identifies patterns in data without prior knowledge of the outcomes. It is used when the data is not labelled. It can also be split into two types: clustering and association. Clustering involves grouping ships based on their features in the data. Both methods assist in finding patterns and understanding information from data that isn't organized in a standard way.

DEEP LEARNING ALGORITHMS:

Deep learning methods employ multiple interconnected layers of nodes to examine intricate non-linear relationships between predictor and response variables. Supervised deep learning has been very successful and is used in many real-world situations. This section talks about some of the methods used in supervised learning. However, before delving into deep learning architectures, it is necessary to introduce the main building block and core foundation of deep learning, i.e., the ANN. The ANN works similarly, to how biological nervous systems, like the brain, handle information. It is made up of many connected processing units called neurons that work together to solve a specific problem. You can picture it as a graph where neurons are connected by arrows, and each arrow has a number that shows how the connection is strong.

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \quad (20)$$

where y represents the output matrix and f denotes the activation function applied to each element. Furthermore, w, x, and b represent the weight matrix, input matrix, and bias vector.

1) CONVOLUTIONAL NEURAL NETWORK:

Convolutional Neural Network (CNN) is a specialized class of neural networks designed to process data using three fundamental building blocks or layers.

The CNN output y can be computed as,

$$y = \sigma(W.X + b) \quad (21)$$

where n represents the number of neurons in the FC layer, σ is the activation function, X indicates the input to the FC layer, W is the weight matrix, and b is the bias vector.

CNNs are particularly effective in processing spatial data, making them ideal for analysing-content screening images and predicting molecular properties from chemical structures. Their ability to automatically extract hierarchical features from input data makes CNNs are robust algorithm for identifying biologically relevant patterns without extensive manual feature engineering.

2) RECURRENT NEURAL NETWORK:

A recurrent neural network, or RNN, is a kind of artificial neural network designed to work with data that comes in sequences. It is characterized by its ability to retain information from previous inputs and use it to make predictions. The RNN structure is made up of linked nodes arranged in a cycle that goes in one direction, which lets information move around in a loop. Moreover, Simple RNN and more advanced versions such as LSTM (Long Short-term Memory) and GRU (Gated Recurrent Unit) work well with sequential data, which is often used in models for pharmacokinetics and pharmacodynamics. These models utilize historical data, making them

valuable for modelling drug response over time and optimizing dosage regimens. The ability of RNNs to remember information is important for understanding and predicting how drugs interact with the body over time.

3) GRAPH NEURAL NETWORKS:

Graph neural networks are specifically designed to process data with complex dependencies and relationships represented as graphs, where nodes are connected by edges. Graph Neural Networks (GNNs) aim to learn representations for each node in the graph by incorporating information from neighbouring nodes and their relationships. GNNs are uniquely suited to modelling data that can be represented as graphs, such as molecular structures and protein–protein interaction networks. By collecting data from linked nodes, GNNs are able to understand the complex connections and relationships in the data. This capability is especially useful for predicting molecular activity, identifying potential drug targets, and examining interactions within biological pathways. GNNs can work with complicated data that shows relationships, which is why they are very important for analyzing molecular and genetic information during drug discovery.

$$m_v = \sum_{u \in N(v)} f(h_u, e_{vu}) \quad (22)$$

where m_v is the aggregated message for node v, obtained by combining information from its neighbouring nodes $N(v)$ using the aggregation function f. The aggregation function can consider both the hidden representations of the neighbouring nodes h_u and the edge features e_{vu}.

NATURAL LANGUAGE PROCESSING:

NLP techniques are essential for extracting valuable insights from vast volumes of unstructured text data in scientific literature, patents, and electronic health records (EHRs), enabling the identification of new drug candidates and biomarker targets.

Natural language processing algorithms can find important information from scientific papers, like experiment findings, chemical designs, and biological processes, to discover possible targets for new drugs and effective treatment methods. By analysing patents, NLP can identify novel chemical entities, understand intellectual property landscapes, and identify potential collaborators or licensing opportunities. Natural language processing can look at electronic health records to find connections and trends between patient features, how diseases develop, and the results of different treatments. This can help find new biomarkers, predict how patients will respond to certain treatments, and create tailored treatment plans.

MOLECULAR FINGERPRINTING:

Molecular fingerprinting is a computational method that converts chemical structures into concise numerical forms, such as bit vectors or feature arrays, to represent essential structural and physicochemical characteristics. These fingerprints help compare things quickly, find similar items, and work with machine learning, which makes them very useful in finding new drugs and studying materials. Common types include circular fingerprints such as Extended Connectivity Fingerprints (ECFP) that show substructure patterns, path-based fingerprints that represent bond sequences, and 3D fingerprints that capture molecular shapes. Molecular fingerprinting focuses on the chemical features, such as functional groups and bonds, rather than linguistic meaning. Tools like RD Kit and Chem Des are commonly used to calculate fingerprints quickly, helping to connect chemistry with artificial intelligence.

REINFORCEMENT LEARNING AND EVOLUTIONARY ALGORITHMS:

Reinforcement learning (RL) offers a powerful approach to drug optimization by framing the process as a decision-making problem. In this framework, an agent (the RL algorithm) learns to take actions, such as modifying the molecular structure and adjusting dosage, to maximize a reward signal like drug efficacy, while minimizing toxicity. Machine learning methods can find the best solutions that might not be obvious to people doing research.

Algorithms based on how nature evolves help in finding the best ways to design molecules. These algorithms copy how evolution works, like mutations and choosing the best traits, to create and improve molecular shapes that have the right features. Through iterative modification of molecular structures and assessment of their fitness according to specific criteria such as binding affinity and drug-likeness, evolutionary algorithms can effectively navigate chemical space to identify new drug candidates with improved properties.

APPLICATIONS OF ALGORITHMS IN DRUG DISCOVERY:

1) Target Identification and Validation:

AI makes it much easier to predict which molecules could be good targets for drugs by looking at a wide range of biological information. AI algorithms can find new targets better than old methods by combining data from genomics, proteomics, and other sources. For example, AI can analyse genomic data to identify genetic

variations linked to diseases and pinpoint genes and their encoded proteins as potential targets. Similarly, AI can analyse proteomic data, such as protein structures and interactions, to identify proteins involved in disease pathways and assess their draggability. Furthermore, AI can integrate multiple data sources, such as Drug Bank, PubChem, Antibiotic Combination Database (ACDB), Antibiotic Adjuvant Database (AADB), as well as clinical trial data and electronic health records, to identify potential targets and predict their therapeutic potential. Machine learning algorithms, like deep learning and natural language processing, are important for studying complicated data sets and finding patterns and connections that humans might not easily notice. Machine learning models have become strong tools for understanding how genes relate to diseases and for finding new biomarkers. These models can analyse complex datasets such as gene expression profiles, single-nucleotide polymorphisms (SNPs), and protein–protein interaction networks, to identify patterns and relationships that traditional statistical methods might overlook. For example, algorithms like SVMs and random forests can be trained using data that includes gene expression information and whether a person has a disease. This helps in predicting the chance of developing a disease and finding which genes are connected to the risk of getting the disease. Unsupervised learning methods, like clustering and dimensionality reduction, can help find groups of genes that show similar expression patterns and discover new subtypes of diseases. Besides that, deep learning models like recurrent neural networks (RNN) and convolutional neural networks (CNN) can look at complicated data from genes and proteins to find detailed patterns and make accurate predictions about disease results. For example, datasets containing 10,000–15,000 entries have been used for target proteins such as Mpro (the main protease of SARS-CoV-2) in antiviral drug development and hERG (human Ether-a-go-go-Related Gene) in evaluating cardiotoxic effects.

2) Drug Screening and Lead Discovery:

AI-driven virtual screening and computer-based methods have changed how scientists find promising drug candidates during the drug discovery process. These methods use computer tools to quickly check large collections of chemicals, making the process much faster and cheaper than older methods that involve testing many chemicals one by one. ML algorithms are essential for these methods. For example, they can help build models that predict how active a compound is in the body based on its chemical makeup. These models can then be utilized to screen extensive chemical libraries and prioritize compounds that have the highest likelihood of binding to the target of interest. These AI methods can really speed up the process of finding good lead compounds and help make drug development more successful in the end.

3) Drug Optimization and Design:

AI-driven techniques are transforming drug development by enhancing crucial properties like solubility, stability, and bioavailability. Machine learning algorithms can look at big sets of chemical structures along with their properties to guess important values very accurately. In QSAR predictions, around 1000–5000 data points were utilized for predicting water solubility, while DL models are capable of predicting drug stability under various conditions. For the task of predicting protein functions, researchers can use two open databases, the UniProt Consortium and the Protein Data Bank (PDB), to collect protein sequence data from different species. This data can then be used to train prediction models by going through steps such as downloading it in batches, cleaning it up, and preparing it for use. These predictive models allow researchers to quickly identify and optimize drug candidates with enhanced physicochemical properties, thus boosting their likelihood of successful clinical translation. Moreover, deep learning algorithms, like generative adversarial networks, can create new chemical structures that have the desired properties, helping to explore a wider range of chemicals in the drug development process.

4) Preclinical and Clinical Development:

AI has transformed clinical trial design, patient recruitment, and data analysis, resulting in more efficient and effective studies. AI algorithms can look at past trial information to help make the study better. They can figure out the best number of people needed, choose the right results to measure, and find the best group of patients to include. AI-powered platforms can greatly help in finding and involving patients by using focused ads and tailored approaches to reach the right people. AI plays a crucial role in real-time data monitoring and analysis. Machine learning algorithms can keep looking at data from clinical trials to find possible safety issues, spot unusual side effects, and check how well treatments work right away. This allows researchers to make smart choices about changing their studies, like adjusting how much medicine is given or adding new treatment options, which helps speed up and improve the clinical trials.

CHALLENGES AND LIMITATIONS OF USING ALGORITHMS IN DRUG DISCOVERY:

1) Data Quality and Availability:

A big problem in using AI for finding new drugs is getting enough good-quality data that's been properly labelled to teach the models. Data from different sources, like chemical structures, biological tests, and clinical

studies, varies a lot, which makes it a big challenge. Combining and making these different data sources work together in one consistent format for training AI can be difficult and take a lot of time. Moreover, unfair tendencies in the training data can greatly impact how well the model works and how trustworthy its results are. For example, if a dataset mostly includes people from a certain group or has a particular illness, the model created from that data might have biases. These biases can make the model less effective and less accurate when used in real-life situations. Addressing these challenges requires careful data curation, robust data preprocessing techniques, and the development of methods to mitigate bias and ensure data representativeness.

2) Interpretability and Transparency:

A big problem preventing many people from using AI systems widely is that they are complicated and hard to understand. Many AI models, particularly deep neural networks, function as "black boxes," making it difficult to interpret the reasoning behind their decisions. The absence of clear explanations and openness causes worries about trust, responsibility, and the risk of hidden bias. For example, in healthcare, it's important for doctors to understand why an AI system gives a certain diagnosis so they can make good choices and keep patients safe.

3) Integration Into Existing Drug Development:

The incorporation of AI tools into current drug development processes poses substantial challenges. Traditional pharmaceutical processes usually follow strict rules and focus a lot on proven methods. Adding AI might need big changes to the current systems, steps, and skills used in these processes. Moreover, worries about keeping data private, protecting intellectual property, and how AI might affect jobs in the pharmaceutical industry can make it harder for companies to accept and use these technologies.

CONCLUSION:

In conclusion, artificial intelligence and advanced algorithms have significantly transformed the field of drug discovery by making the process faster, more efficient, and data driven. Techniques such as machine learning, deep learning, and various algorithmic models enable accurate target identification, virtual screening, drug design, and clinical analysis, thereby reducing time, cost, and failure rates associated with traditional methods. Despite these advantages, challenges such as data quality, model interpretability, and integration into existing pharmaceutical workflows remain important concerns. Addressing these limitations is essential to fully realize the potential of AI in drug discovery. Overall, the continued development and responsible implementation of AI and algorithms hold great promise for accelerating innovation and improving the success rate of developing new and effective medicines.

BIBILIOGRAPHY:

1. DiMasi JA, Grabowski HG, Hansen RW. Innovation in the pharmaceutical industry: new estimates of R&D costs. J Health Econ. 2016; 47:20–33.
2. Hay M, et al. Clinical development success rates for investigational drugs. Nat Biotech Nol. 2014;32(1):40–51.
3. Xue H, et al. Review of drug repositioning approaches and resources. Int J Biol Sci. 2018;14(10):1232.
4. Low ZY, Farouk IA, Lal SK. Drug repositioning: new approaches and future prospects for life-debilitating diseases and the COVID-19 pandemic outbreak. Viruses. 2020; 12(9):1058.
5. https://www.geeksforgeeks.org/artificial-intelligence/what-is-artificial-intelligence-ai/
6. https://www.geeksforgeeks.org/dsa/introduction-to-algorithms/
7. Rustam, F., Reshi, A. A., Mehmood, A., Ullah, S., On, B. W., Aslam, W., et al. (2020). COVID-19 future forecasting using supervised machine learning models. IEEE Access, 8, 101489–101499.
8. Dalal, K. R. (2020). Analysing the role of supervised and unsupervised machine learning in iot. In 2020 international conference on electronics and sustainable communication systems (pp. 75–79). IEEE.
9. Aruleba, R. T., Adekiya, T. A., Ayawei, N., Obaido, G., Aruleba, K., Mienye, I. D., et al. (2022). COVID-19 diagnosis: A review of rapid antigen, RT-PCR and artificial intelligence methods. Bioengineering, 9(4), 153.
10. Baştanlar, Y., & Ozuysal, M. (2014). Introduction to machine learning. miRNomics: MicroRNA Biology and Computational Analysis, 105–128.
11. Arar, O. F., & Ayan, K. (2017). A feature dependent naive Bayes approach and its application to the software defect prediction problem. Applied Soft Computing, 59,197–209.

12. Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. Journal of Information Science, 44(1), 48–59.
13. Singh, G., Kumar, B., Gaur, L., & Tyagi, A. (2019). Comparison between multinomials and Bernoulli naive Bayes for text classification. In 2019 international conference on automation, computational and technology management (pp. 593–596). IEEE.
14. Kelly, A., & Johnson, M. A. (2021). Investigating the statistical assumptions of Naive Bayes classifiers. In 2021 55th annual conference on information sciences and systems (pp. 1–6). IEEE.
15. Seixas, F. L., Zadrozny, B., Laks, J., Conci, A., & Saade, D. C. M. (2014). A Bayesian network decision model for supporting the diagnosis of dementia, Alzheimer's disease and mild cognitive impairment. Computers in Biology and Medicine, 51,140–158.
16. Kyrimi, E., McLachlan, S., Dube, K., Neves, M. R., Fahmi, A., & Fenton, N. (2021). A comprehensive scoping review of Bayesian networks in healthcare: Past, present and future. Artificial Intelligence in Medicine, 117, Article 102108.
17. Harris, J. K. (2021). Primer on binary logistic regression. Family Medicine and Community Health, 9(Suppl 1).
18. Ekins, S., Puhl, A. C., Zorn, K. M., Lane, T. R., Russo, D. P., Klein, J. J., et al. (2019). Exploiting machine learning for end-to-end drug discovery and development. Nature Materials, 18(5), 435–441.
19. Seng, J. K. P., & Ang, K. L. M. (2017). Big feature data analytics: Split and combine linear discriminant analysis (SC-LDA) for integration towards decision making analytics. IEEE Access, 5, 14056–14065.
20. Thomaz, C. E., Kitani, E. C., & Gillies, D. F. (2006). A maximum uncertainty LDA-based approach for limited sample size problems — with application to face recognition. Journal of the Brazilian Computer Society, 12(2), 7–18.
21. Gholami, R., & Fakhari, N. (2017). Support vector machine: principles, parameters, and applications. In Handbook of neural computation (pp. 515–535).
22. Suthaharan, S., & Suthaharan, S. (2016). Support vector machine. In Machine learning models and algorithms for big data classification: thinking with examples for effective learning (pp. 207–235).
23. Anava, O., & Levy, K. (2016). K-nearest neighbours: From global to local. In Advances in neural information processing systems: vol. 29.
24. Gavankar, S. S., & Sawarkar, S. D. (2017). Eager decision tree. In 2017 2nd international conference for convergence in technology (pp. 837–840). IEEE.
25. Mienye, I. D., & Jere, N. (2024). A survey of decision trees: Concepts, algorithms, and applications. IEEE Access.
26. Breiman, L. (2017). Classification and regression trees.
27. Mienye, I. D., Sun, Y., & Wang, Z. (2019). Prediction performance of improved decision tree-based algorithms: A review. Procedia Manufacturing, 35, 698–703.
28. Zou, R. Y., & Schonlau, M. (2018). The random forest algorithm for statistical learning with applications in stata. The Stata Journal, 20(3).
29. Mienye, I. D., & Sun, Y. (2022). A survey of ensemble learning: Concepts, algorithms, applications, and prospects. IEEE Access, 10, 99129–99149.
30. He, J., Hao, Y., & Wang, X. (2021). An interpretable aid decision-making model for flag state control ship detention based on SMOTE and XG Boost. Journal of Marine Science and Engineering, 9(2), 156.
31. Dhaliwal, S. S., Nahid, A.-A., & Abbas, R. (2018). Effective intrusion detection system using XG Boost. Information, 9(7), 149.
32. Li, Y., & Chen, W. (2020). A comparative performance assessment of ensemble learning for credit scoring. Mathematics, 8(10), 1756.
33. Mienye, I. D., Obaido, G., Aruleba, K., & Dada, O. A. (2021). Enhanced prediction of chronic kidney disease using feature selection and boosted classifiers. In International conference on intelligent systems design and applications (pp. 527–537).

34. Zheng, H., Xiao, F., Sun, S., & Qin, Y. (2022). Brillouin frequency shift extraction based on AdaBoost algorithm. Sensors, 22(9), 3354.
35. Sevinç, E. (2022). An empowered AdaBoost algorithm implementation: A COVID-19 dataset study. Computers & Industrial Engineering, 165, Article 107912.
36. Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CAT Boost: unbiased boosting with categorical features. In Advances in neural information processing systems: vol. 31.
37. James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). Unsupervised learning. In An introduction to statistical learning: with applications in python (pp.503–556).
38. Jain, A. K., Mao, J., & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial. Computer, 29(3), 31–44.
39. Kattenborn, T., Leitloff, J., Schiefer, F., & Hinz, S. (2021). Review on convolutional neural networks (CNN) in vegetation remote sensing. ISPRS Journal of Photogrammetry and Remote Sensing, 173, 24–49.
40. Dernoncourt, F., Lee, J. Y., Uzuner, O., & Szolovits, P. (2017). De-identification of patient notes with recurrent neural networks. Journal of the American Medical Informatics Association, 24(3), 596–606.
41. Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2008). The graph neural network model. IEEE Transactions on Neural Networks, 20(1), 61–80.
42. Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2022). Graph neural networks in recommender systems: A survey. ACM Computing Surveys, 55(5), 1–37.
43. Agarwal P, Searls DB. Literature mining in support of drug discovery. Brief Bioinform. 2008;9(6):479–92.
44. Kosonocky CW, et al. Mining patents with large language models elucidates the chemical function landscape. Digital Discovery.2024;3(6):1150–9.
45. Cowie MR, et al. electronic health records to facilitate clinical research. Clin Res Cardiol. 2017;106(1):1–9.
46. Li Z, et al. Fingerprinting interactions between proteins and ligands for facilitating machine learning in drug discovery. Biomolecules. 2024;14(1):72.
47. Landrum G. Rdkit documentation. Release. 2013;1(1–79):4.
48. Dong J, et al. ChemDes: an integrated web-based platform for molecular descriptor and fingerprint computation. J Cheminform. 2015; 7:60.
49. Popova M, Isayev O, Tropsha A. Deep reinforcement learning for de novo drug design. Sci Adv. 2018;4(7): eaap7885.
50. Jayaraman P, et al. A primer on reinforcement learning in medicine for clinicians. NPJ Digit Med. 2024;7(1):337.
51. Lameijer E-W, et al. Evolutionary algorithms in drug design. Nat Comput. 2005;4(3):177–243.
52. Abou Hajal A, Al Meslamani AZ. Insights into artificial intelligence utilisation in drug discovery. J Med Econ. 2024;27(1):304–8.
53. Hopkins AL, Groom CR. The druggable genome. Nat Rev Drug Discov. 2002;1(9):727–30.
54. Lin Q, Tam PK-H, Tang CS-M. Artificial intelligence-based approaches for the detection and prioritization of genomic mutations in congenital surgical diseases. Front Pediatr. 2023; 11:1203289.
55. Overington JP, Al-Lazikani B, Hopkins AL. How many drug targets are there? Nat Rev Drug Discov. 2006;5(12):993–6.
56. Knox C, et al. Drug Bank 6.0: the Drug Bank Knowledgebase for 2024. Nucleic Acids Res. 2024;52(D1):1265-d1275.
57. Kim S, et al. PubChem 2025 update. Nucleic Acids Res. 2024;53(D1): D1516–25.
58. Lv J, et al. ACDB: an antibiotic combination database. Front Pharmacol. 2022; 13:869983. Qiu X, et al. Advances in AI for protein structure prediction: implications for cancer drug discovery and development. Biomolecules. 2024;14(3):339.

59. Serrano DR, et al. Artificial intelligence (AI) applications in drug discovery and drug delivery: revolutionizing personalized medicine. Pharmaceutics. 2024;16(10):1328.
60. Osama S, Shaban H, Ali AA. Gene reduction and machine learning algorithms for cancer classification based on microarray gene expression data: a comprehensive review. Expert System Appl. 2023; 213:118946.
61. Statnikov A, et al. A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics. 2005;21(5):631–43.
62. Friedman J. The elements of statistical learning: data mining, inference, and prediction. (No Title), 2009.
63. Ching T, et al. Opportunities and obstacles for deep learning in biology and medicine. J R Soc Interface. 2018;15(141):20170387.
64. Boby ML, et al. Open science discovery of potent noncovalent SARS-CoV-2 main protease inhibitors. Science. 2023;382(6671):201.
65. Shen M-Y, et al. A comprehensive support vector machine binary hERG classification model based on extensive but biased end point hERG data sets. Chem Res Toxicol. 2011;24(6):934–49.
66. Naithani U, Guleria V. Integrative computational approaches for discovery and evaluation of lead compound for drug design. Front Drug Discov. 2024. https://doi.org/10.3389/fddsv.2024.1362456.
67. Paul D, et al. Artificial intelligence in drug discovery and development. Drug Discov Today. 2021;26(1):80–93.
68. Niazi SK, Mariam Z. Recent advances in machine-learning-based chemoinformatics: a comprehensive review. Int J Mol Sci. 2023;24(14):11448.
69. Vidhya KS, et al. Artificial intelligence's impact on drug discovery and development from bench to bedside. Cureus. 2023;15(10): e47486.
70. Singh S, et al. Artificial intelligence and machine learning in pharmacological research: bridging the gap between data and drug discovery. Cureus. 2023;15(8): e44359.
71. Piir G, et al. Best practices for QSAR model reporting: physical and chemical properties, ecotoxicity, environmental fate, human health, and toxicokinetics endpoints. Environ Health Perspect. 2018;126(12):126001.
72. An F, et al. Machine learning model for prediction of drug solubility in supercritical solvent: modeling and experimental validation. J Mol Liq. 2022; 363:119901.
73. Consortium TU UniProt: the universal protein knowledgebase in 2025. Nucleic Acids Res. 2024;53(D1): D609–17.
74. Berman HM, Burley SK. Protein Data Bank (PDB): Fifty-three years young and having a transformative impact on science and society. Q Rev Biophys. 2025;58: e9
75. Chen J-Y, et al. Evaluating the advancements in protein language models for encoding strategies in protein function prediction: a comprehensive review. Front Bioeng Biotechnol. 2025; 13:506508.
76. Sousa T, et al. Generative deep learning for targeted compound design. J Chem Inf Model. 2021;61(11):5343–61.
77. Bretz F, et al. Adaptive designs for confirmatory clinical trials. Stat Med. 2009;28(8):1181–217.
78. Chow S-C, Liu J-P. Design and analysis of clinical trials: concepts and methodologies, vol. 507. Hoboken: Wiley; 2008.
79. Blanco-González A, et al. The role of AI in drug discovery: challenges, opportunities, and strategies. Pharmaceuticals (Basel). 2023;16(6):891.
80. Kokudeva M, et al. Artificial intelligence as a tool in drug discovery and development. World J Exp Med. 2024;14(3):96042.
81. Gichoya JW, et al. AI pitfalls and what not to do: mitigating bias in AI. Br J Radiol. 2023;96(1150):20230023.
82. Cheong BC. Transparency and accountability in AI systems: safeguarding wellbeing in the age of algorithmic decision-making. Front Hum Dynam. 2024. https://doi.org/10.3389/fhumd.2024.1421273.

83. Kiseleva A, Kotzinos D, De Hert P. Transparency of AI in healthcare as a multilayered system of accountabilities: between legal requirements and technical limitations. Front Artif Intell. 2022; 5:879603.

84. Dwivedi YK, et al. Artificial Intelligence (AI): multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. Int J Inf Manage. 2021; 57:101994.