AI in Drug Discovery and Development

Prof. Rajnish Kumar
Dept. of Pharmaceutical Engineering and Technology
IIT-(BHU), Varanasi

Week-03
Lecture-13

Welcome to the course "AI in Drug Discovery and Development." In today's session, we will talk about feature engineering and data preprocessing. So, by the end of this lecture, you will be able to understand the importance of data preprocessing in the ML pipeline and its impact on model performance. Identify and apply common data cleaning techniques, including handling missing values, duplicates, and outliers. Explain and perform data transformation tasks such as normalization, standardization, and scaling. Differentiate among various encoding strategies and apply them appropriately based on the data types.

As well as evaluating pre-processing strategies using metrics and visualization tools to ensure data readiness for modeling. In earlier sessions, we have talked about building machine learning models as well as artificial neural network models. So, today we will talk about what data pre-processing is, what feature engineering is, and why it is important. The main purpose of the data pre-processing process is to ensure data integrity because the biological data set often has missing values.

It is mostly noisy as well because it is coming from diverse sources, and we do not have control over that. Most of the time, the data we use in building those predictive models comes from public resources. And the data coming from those public resources has a lot of, you know, problems, actually, or it may also consist of inconsistent values. So, the steps to get rid of all these problems are to standardize the molecular descriptors like molecular weight or LogP as well as normalize the assay conditions. To normalize the assay results, the values in the IC_{50} or EC_{50} should be used, and the structure also needs to be uniformly assigned, such as through SMILES canonicalization and outlier removal.

And so, this part is called pre-processing, and then the feature engineering step is to translate those complex chemical or biological data into a machine-readable format. So, many times we convert those molecules, like those structures, into fingerprints or physical and chemical descriptors, like TPSA or rotatable bonds, or into a protein-ligand interaction matrix. So, it enables integration of multi omics pharmacological data for richer models as well. So, that multi-omics data can be in the form of, you know, numerical values or in the form of text as well. So, it has, you know, an overall impact on the model quality.

So, it improves predictive accuracy, such as bioactivity, ADMET, or drug likeness, as well as enhances interpretability and reduces dimensionality. And it is essential to avoid false

predictions in critical pipelines. And it is said that poor preparation increases the risk of false positives and negatives, leading to costly validation errors and delays in lead optimization. So when we look at the different types of data, as I said, we are getting multiple kinds of data when we talk about drug discovery and development. So the data can be divided into basically four parts.

One is called clinical data, where the examples could be patient records, clinical trial outcomes, adverse events, or electronic health records. And it is used for safety profiling, drug repurposing, and real-world evidence. And the source of this data could be from ClinicalTrials.gov, the FDA database, or the hospital electronic health records. So, this data is mostly in the form of, you know, text, and sometimes it is numerical data as well.

And then another type of data is the chemical data; mostly, it is relevant to the cheminformatics pipeline where we have the structure of those molecules, which might be represented in strings like in the SMILES or as the InChI. You know representation, or they can be represented as features as well, like molecular fingerprints or descriptors. And this is used mainly for, you know, doing structure-based virtual screening, like finding compound similarity or doing quantitative structure-activity relationship modeling, or doing, you know, structure-based virtual screening as well. And then the source of this kind of data largely comes from ChEMBL, PubChem, ZINC, or the Drug Bank. And then we have the omics data, which is, you know, coming from the genomics studies or transcriptomics, proteomics, metabolomics, and epigenomics.

So, all those sorts of data are coming from those omics techniques, and by which we can understand the disease mechanism. It can help us in patient stratification while designing the clinical trial, as well as in biomarker discovery. And then the source of the data is coming from, like the databases GEO, TCGA, PRIDE, or MetaboLights. And then we have the biological data. For example, the biological data includes protein sequences, protein structures, binding sites, and bioassay results.

And then it is used for target identification, binding affinity prediction, and molecular docking as well. So, this data can come from the PDB, which is a protein data bank that consists of the deposited 3D structures of proteins. And then you have UniProt and BindingDB. So, there are many such databases that consist of this kind of data. Talking specifically about the chemical structure data.

So, because most of the drug discovery, at least the drug discovery part, is, you know, related to small molecules, we try to identify molecules that can engage the target and give us the desired effect. So, it is essential for model input and learning from the chemical space, and as I said, it can be represented in multiple ways. So, SMILES is one way in

which the structure of the molecule is represented in a 1D string, and then you have the InChI or InChI key, and then you have the SELFIES, molecular graphs, deep learning embeddings, and fingerprints like ECFP, MAX, or SMARTS. And we will discuss all these representations step by step, and then you will have the biological data. So, that biological data can come from gene expression data, which captures changes in gene activity under different conditions, like disease versus healthy.

And then it can be used for, you know, mechanism action studies, biomarker discovery, or drug response prediction, and it can come from GEO, LINCS, or TCGA. So, these are some of the databases where this gene expression data can be utilized. And then you have the protein sequence and structure data, which can consist of either primary information, primary structure, or 1D structure, which is known as the amino acid sequence. The 3D structural information can be used for target identification, structure-based drug design, and binding site prediction. and then sources can be UniProt, PDB, AlphaFoldDB, etc.

And then we have the functional and interaction data. So, this could be in the form of protein-protein interaction networks or signaling pathways, and it is being used in systems biology, target prioritization, and polypharmacology analysis. So, the sources for this are String, BioGRID, or IntAct. And then you have the target annotation data, which is the information about known drug targets, biological pathways, and disease associations. So, it includes gene protein function and pathway involvement, which pathways those genes are involved in, such as KEGG or Reactome, and the disease linkage DisGeNET, with sources like DrugBank, ChEMBL, or TTD.

Okay, and then talking about the assay and STS data, which is part of, you know, biological data. So, whenever a compound is tested or evaluated against a drug target. So, we measure its activity in some, you know, some quantitative measurement. So, the assay data is an experimental measurement of compounds' activity on biological targets, and it can be collected via biochemical, cell-based, or phenotypic assays. So, usually, it is obtained classically through high-throughput screening, which is an automated testing of large compound libraries that generates large-scale activity data for hit identification.

So, some of the key activity matrices include IC_{50}, which is the inhibition constant 50 and indicates the potency of the inhibitor. So, a lower IC_{50} means that the compound is highly potent; in fact, its potency is high. And then you have the EC_{50}, which is the effective concentration 50, the concentration of a compound for 50 percent of the maximum effect. If a molecule can inhibit the enzyme activity of any enzyme by 50 percent at 10 micromolar, then 10 micromolar will be the EC_{50} of that compound to inhibit that specific enzyme. And then you have the Kd or Ki, so these indicate the binding affinity, like how strongly a molecule can bind to its target.

And then this is essential for SAR and lead optimization. This is a very important metric which is, you know, used extensively for evaluating the SAR and optimizing the lead compounds. And this data can be in different formats. So, it can be raw or curated like we obtain the data in IC_{50} and then we convert it into pIC_{50}, which is a negative logarithm of the IC_{50} value. And then it can come from different databases, like ChEMBL, which is one of the databases that curates the structures of the molecules and their associated biological activities.

And then you have the BindingDB, which is also similar to ChEMBL. And then you have the PubChem Bioassays and GtoPDB; these are all the databases that consist of biological activity data. And then, how is it relevant to the AI models? Because it is used as labels in regression and classification tasks. So, the data that we use, and this is the dependent variable: the biological activity. So, it is being used to build a model, and it is being used as training data to build those models, as well as for quality and standardization.

So the units of the data replicate, and filtering these is critical. And then we have the ADMET and PK/PD data in drug discovery. So, where ADMET refers to the absorption, distribution, metabolism, excretion, and toxicity properties of a drug, which collectively determine its behavior, safety, and effectiveness in the body. ADMET is the pharmacokinetic term that describes what our body does to the drug; those processes are known as pharmacokinetic processes. So, it is a quantitative analysis of drug concentrations over time.

So, the parameters can include C_{max}, the maximum concentration, T_{max}, the time at which the concentration is maximum, and then the area under the curve, $T_{1/2}$, bioavailability, etcetera. So, it informs the dosing regimen and the exposure profile of a molecule. Then we have the pharmacodynamics data, which is a relationship between drug concentration and the biological effect. Then we have different models like the E_{max} model or the Hill equation, which are used for determining the pharmacodynamics of these molecules, and they describe efficacy, potency, and the therapeutic index. So why this data is important in AI models is because ADMET prediction is the key to reducing late-stage failure, and the lack of safety and efficacy is one of the major reasons why many drugs fail in late-stage clinical trials.

So, if we can predict those ADMET properties, so we can reduce the chances of failure in the late stage clinical trials. So, this and then AI can model the relationship between the chemical structures and the ADMET PK/PD outcomes. and this data is being used in QSAR modeling for toxicity, PK/PB modeling and multi objective optimization where we are trying to optimize the efficacy as well as the safety. Okay, so now we have talked about

different sorts of data. So now the next thing is data pre-processing.

And the first thing that is in the data pre-processing is the data cleaning. And the data cleaning means we need to handle the missing values and the outliers. So why is data cleaning important? Because it ensures the accuracy and reliability of models, as well as being essential for generating high-quality datasets for predictive modeling, like our ADMET or HTS data. And it improves the performance of ML models by providing cleaner and more consistent inputs. So, how do we handle the missing values? So, the common cause of missing values is, as I said many times, whenever we are curating the biological data.

So, it is coming from multiple sources, okay. So, it can consist of the incomplete assays, dropout during the experiments, or loss of data in the clinical trials. So, we need to take care of those missing values, and how do we do that? We do the imputation. By filling the gaps using the mean, median, and mode or model-based approaches like KNN or regression modeling, we can predict those missing values and utilize them. Or what we can do is delete those values.

So, we can either remove rows and columns with excessive missing data or predict them using ML models and then use them in the training data set. So, some of the tools that can be used are Pandas, scikit-learn, or R. And then another thing is outliers. So, outliers are the data points that significantly deviate from the general trend, like extreme IC_{50} values or unrealistic ADMET predictions. So, what they can do is distort the statistical analysis and predictive models, which may indicate data errors or experimental anomalies.

So, how to handle those outliers is one method known as capping, where we can replace outliers with the boundary values, or what we can do is transform them. So, applying a log or square root transformation to normalize the data and get rid of those outliers. Or we can remove them, excluding data points deemed as errors or true outliers based on domain knowledge. However, this is a very serious concern because we will talk about the activity cliff as well, where a small change in a molecular structure can lead to a big change in the bioactivity. So, if we consider that as an outlier, we might miss the, you know, the relation between the structure and the activity.

So, we need to handle them very carefully. So, the best practice to handle this is that we need to have the domain knowledge; we use the scientific context of known drug toxicity thresholds to guide the cleaning. As well as doing the visualization, we detect the missing values and utilize plots like histograms, box plots, or heat maps. Okay, another thing is standardization versus normalization. So, the standardization known as Z-score scaling involves centering the data by subtracting the mean and scaling by the standard deviation.

So, what it ensured is, it ensured that all features contribute equally avoiding the scale bias.

So, it is used in cases where features have different units or scales, like IC_{50} values or molecular descriptors, and it is required for models assuming a normal distribution. So, like if we are using a modeling technique that assumes the data is normally distributed and the data is not actually normally distributed. So, we try to standardize it to obtain a normal distribution, and then we use techniques like SVM or linear regression to model the prediction. And then another technique is normalization, where what we try to do is min-max scaling. We rescale the data to a fixed range, typically from 0 to 1.

So it prevents large-value features from dominating the learning. So it is used when neural networks or models that are sensitive to the feature scale are being used. And the features vary widely, such as molecular weight and LogP. For example, molecular weight can be in the range of 100 to 800, while LogP will be in the range of 0 to 5. So, the scale of the two parameters' values is very different.

So, to normalize them, we bring both of them onto the same scale from 0 to 1, and that is the normalization. Okay, another thing is encoding the categorical features. So, if we have the categorical data, you know, as categorical features for our molecules or for our biological data. So, what we need to do is encode them into numbers because machines cannot, and ML algorithms require numerical inputs, okay. So, the categorical data can include drug class, target type, or disease category that needs to be converted into numerical representation.

And proper encoding ensures model performance and interpretability. So, there are multiple methods. So, one of the methods is known as one-hot encoding, where we create binary columns for each category, like we have drug A, B, C, okay. So, we have drugs A, B, and C. And then it is useful when no ordinal relationship exists between the categories, such as the target classes.

So, we just give the 1-node encoding as if the a is there. So, then it is 1, and then if b is there, 1, and then if c is. So, in both cases, c is not there. So, it is represented by 0. And then we can use another technique, which is label encoding, where we convert the categories into integer labels, like drug A is 0, drug B is 1, and drug C is 2.

So, it is suitable for you to know the ordinal relationships ranked or ordered, for example, for the efficacy labels. So, this is for the label; you know the label encoding. And then we have the binary encoding, which is efficient for high cardinality categorical variables like a large number of drug categories, and we convert categories into binary digits that are more compact than one-hot encoding. So, like this is for drug A and B; like bin 1 B is there,

bin 2 A is there, and bin 3 B is there. Okay, and then what we need to consider in this case is dimensionality; for example, one-hot encoding can increase feature space size, leading to sparse matrices, and the model choice, as some models can handle label encoding while others, like neural networks, may require one-hot encoding.

Okay, the next thing is handling the imbalanced data sets. So, the class imbalance occurs when one class, for example, active versus inactive compounds, is significantly underrepresented in the data set. We have a data set of 1,000 compounds, and only 50 of them are active while 950 are inactive. So, it means that the active compounds are underrepresented, which indicates they have a very low number of those compounds. So, it leads to biased model predictions that favor the majority class, predicting the inactive more often, and it is common in drug discovery, especially for example, in identifying rare disease targets or predicting toxicity.

So, the techniques to handle the imbalances, like S-MOTE (Synthetic Minority Oversampling Technique), generate synthetic samples for the minority class by interpolating between the existing instances, helping to balance the dataset without losing information. And then you can use undersampling, which reduces the number of samples in the majority class to balance the dataset. And it may lead to a loss of valuable data, but it is useful when data is abundant in the majority class. And then class weight, where we adjust the weight of each class in the model's loss function to penalize the misclassification of the minority class more. And it is suitable for models that support class weight adjustments, like SVM, random forest, or neural nets.

So, this is something we have already talked about, like when we need to use it and what the pros and cons are. Like the SMOTE process, it generates synthetic data and avoids data loss. It can actually introduce noisy or overfitted data. And then, under sampling, the advantage is that it is simpler and reduces the computation time. However, it can potentially lead to the loss of valuable data in the major class.

And then the problems with the class weight; the advantages of the class weight method are that it is effective without changing the data set size. However, it may not work well in the highly imbalanced datasets. Okay, another technique is splitting the data set. So, it is essential for model evaluation and generalization, and it helps ensure that models are not overfitting and can perform well on unseen data. So, the validation and test sets provide a reliable measure of model performance across different data distributions, like drug efficacy in different conditions.

So, some of the common dataset splitting techniques. So, the training, validation, and test split is where we use the training set to train the model, the validation set is used for

hyperparameter tuning and model selection, and the test set is used to evaluate model performance on unseen data. So, typically we use a 60-20-20 split, where 60 percent of the molecules are in the training set, 20 percent in the validation set, and 20 percent in the test set, but it may vary depending on the data size. And then another technique is stratified k-fold cross-validation, where the dataset is divided into k equal subsets and the model is trained and validated k times, each time using a different fold for validation and the remaining for training. So, it is stratified because the stratification ensures that each fold maintains the same class distribution as the original data set, which is crucial for imbalanced data sets whenever we are talking about building a drug efficacy model or a toxicity model. And when do we use it? So, it is used for robust performance evaluation in smaller data sets where the model is validated multiple times.

And then we can use the temporal split. It is a time-based splitting method that is used when data has a temporal dimension, such as clinical trial data or drug discovery time series data. So, training on past data and testing on future data helps simulate a real-world scenario where future data is not available during model training. And when do we use it? We use it in scenarios where the temporal trends are important, such as drug release kinetics, disease progression, or clinical trials. And some of the best practices for data splitting are that we shall use stratified K-fold, which is best for small or imbalanced data sets, where it is crucial to maintain the distribution of classes, like different drug classes or target types. Or the temporal split, which is best for time-sensitive data, where you need to predict future outcomes based on historical data.

Example patient outcomes, drug response over time, and ensuring that the randomization is known temporal data splits to avoid bias. Okay, so that was about, you know, the data pre-processing. So, what kind of data do we have, and how do we pre-process that data? So, the next thing is feature engineering. So, let us talk about feature engineering for cheminformatics, like building models for small molecules. And one of the major features for small molecules is the molecular descriptors.

So, it is a numerical value that represents molecular properties, which are used as features in QSAR or ML models. It is derived from chemical structures and encodes physicochemical, topological, or spatial characteristics. So, we have different descriptor types, like 0D, 1D, 2D, 3D, and 4D. And then the purpose is for 0D descriptors. So, these are like simple counts of atoms or bonds, such as the number of carbon atoms, the number of sulfur atoms, the number of nitrogen atoms, the number of oxygen atoms, or the molecular composition.

And then we have 1D descriptors. These are basic physicochemical descriptors and properties that do not require 2D or 3D structure information for their calculation. And then

you have the 2D descriptors, which capture the connectivity or the substructure information, like how those atoms are connected to each other. And then we have the 3D descriptors, which encode 3D conformation or features such as steric or electrostatic features, as well as time step in this data, or it could be the grid or conform methods. So, these are like the different types of 1D, 2D, and 3D descriptors, and then here you can see the structure, like how they are being calculated or, you know, what kind of information they consist of. So, in 0D, it consists of information about the composition, such as what the atoms in this molecule are or what those properties are.

And then 1D is like talking about substructures or the sequence of connectivity; in 2D, you get to know how those molecules are connected to each other. However, in 3D, you can get information about the 3D features, such as pharmacophoric features, which groups or parts of the molecule are important for bioactivity. And then you have the 4D, where you add additional, you know, information—a fourth dimension that is called 4D descriptors. Okay, another type of feature is the molecular fingerprint, which is very, very common. So, these are binary or hashed factors that encode structural information of the molecules, and they capture the presence or absence of substructures and are used for similarity searches, clustering, and ML modeling.

So, these are compact, efficient representations suitable for large datasets. So, some of the common types of fingerprints are like MACCS keys, which are a kind of predefined library. So, it is a 166-bit vector and codes specific structural keys like the aromatic rings. And then we have the Morgan ECFP extended connectivity fingerprint, which is circular and hashed. So, it captures local atomic environments, and it is widely used in QSAR modeling.

And then you have the PubChem fingerprint, which is again a binary vector. So, it is 881 bits and codes for substructure presence from the PubChem database. Another important representation is the molecular graph representation. So, where a molecule is represented as a graph, where nodes are, you know, atoms and the edges are the chemical bonds. So, each atom can be enriched with features such as the type of atom, whether it is carbon, oxygen, or nitrogen, atomic number, hybridization, formal charge, aromaticity, valence electrons, and atom chirality. The edges can include the bond type: single, double, triple, aromatic, bond stereochemistry, whether it is in a ring or not, or the conjugation status.

So, it is using graph-based neural models like graph neural networks to capture the complex structural and relational information. And why do we need those graphical representations? Because it allows models to learn directly from the molecular structure. Because when we talk about a molecule, a bioactive molecule, its three-dimensional structure or conformation is responsible for binding to the binding pocket and giving the desired therapeutic effect. So, if we can get that information and code it. So, that will be

best for modeling those properties, and this is exactly why we need the graph representation where we can add that 3D structure as well as the bioactive conformation.

So, it captures non-linear and non-local interactions more effectively than fingerprints or descriptors and is crucial for probability prediction, toxicity classification, and drug target interaction modeling. So, this is an illustration of different molecular graph representations of aspirin molecules. So, this is your aspirin molecule, and then this is a pharmacophore graph; this is a functional group graph where the functional groups are represented and how those functional groups are connected. Ah, and then you have the atom-by-atom graph where you can see that all those atoms are represented as the nodes, and then bonds are represented as, you know, the edges. And then you have the junction tree graph, and there are many other sorts of graphs as well; people are developing new kinds of graphs every day to make it more valuable, actually.

And then we have the physicochemical features, like the quantitative molecular properties derived from the structure, which describe the molecule's solubility, permeability, and reactivity. And these are essential for QSAR modeling and filtering out drug-like molecules in ADMET prediction. So, some of the common physicochemical descriptors are LogP, which is an optimal water partition coefficient that indicates lipophilicity, which affects absorption and permeability. And then you have the molecular weight, which is the sum of atomic weights that affects the bioavailability, transport, and rules of use for the rule of 5 filtering. And then you have the topological polar surface area, which is the surface area coming from the polar atoms like nitrogen, oxygen, etc.

, and it is related to membrane permeability, especially BBB permeation. So, what role these physicochemical features can play in drug discovery is that they are critical for early filtering of the chemical libraries and are used in Lipinski's rule of five for oral bioavailability, and they can serve as features in the ADMET toxicity prediction models as well. So another kind of feature is actually the scaffold. So scaffold extraction and scaffold-based splitting. So the scaffold is the core structure of a molecule without its side chains, and it is commonly extracted using the Bemis-Murko scaffold method.

So, the Bemis-Murko scaffold method is a method to extract the scaffold. Usually, a molecule is made up of, you know, a core scaffold, and then it has, you know, various substitutions on that core scaffold. So it represents the chemotype or the core chemical framework. And why do we need to extract those scaffolds? So it can identify structural diversity within the data sets, as well as help detect the chemical bias and over-representation of these scaffolds. It's also useful for visual clustering, diversity analysis, and lead-hopping. And then what we can do is use a scaffold-based dataset splitting instead of random splitting; molecules can be grouped by scaffold, and it ensures that no scaffold

is shared across the training, validation, and test sets.

So, to make sure that the training, validation, and test sets are equally diverse in their composition of the molecules. So, the scaffold-based splitting can also be used. And it prevents data leakage and overestimation of the model's performance. The benefits of ML in drug discovery encourage learning that generalizes across chemotypes and provide a more realistic evaluation of model performance on novel scaffolds, which is especially important for QSAR and bioactivity prediction tasks.

And then we have the techniques called feature reduction and interpretability. So, many times when we talk about the molecular features, they have high-dimensional features, which can be thousands of descriptors. Now, out of those thousands of descriptors, which of those features are responsible for or, you know, correlating with our endpoints or biological activity So, that is one question, and that is why we need the feature reduction technique. So, it reduces redundancy, improves the training speed, and combats the overfitting. And it also enhances model interpretability and performance.

So, we use several techniques, and one of them is PCA, principal component analysis. So, it is an unsupervised technique for dimensionality reduction. It projects features into orthogonal components, capturing maximum variance, and helps visualize data clusters and assess the structural activity relationships. so the feature selection technique we use is we identify the relevant and informative features for the task like a variance thresholding, recursive feature elimination, mutual information, chi-square test, embedded method like lasso tree based importance So these are all different methods that can be used for feature selection, so why is interpretability important in drug discovery? Because understanding why a model predicts a compound as active is very crucial information, and that is known as explainability. So, if we cannot explain why those features or this model is working, then it is of no use. And it also enables mechanistic insights, supports regulatory acceptance, and aids in hypothesis generation.

So, let's talk about feature engineering for biological data. So, most of the time we have the gene expression vector data. So, which we need to normalize and filter that data. So, the gene expression vector is a quantitative profile of gene activity, similar to that obtained from RNA-seq or microarray, where each sample is represented by expression levels of thousands of genes, and it is used in drug response prediction, target identification, and disease classification. So, why do we need to normalize the gene expression data to ensure comparability across different samples, experimental batches, and platforms like microarray versus RNA-seq? And the normalization techniques we use are like TPM (transcripts per million), FPKM, or RPKM for RNA-seq. Quantile normalization for the microarray data and Z-score standardization, which are useful for the ML models.

And then another step is filtration for feature reduction, where the gene expression data is high dimensional. We reduce the noise by removing the low variance genes, selecting differentially expressed genes, and using the biologically relevant gene sets from the pathways or based on the targets. So this is an exemplary gene-filtering strategy for ML-guided biomarker discovery. So where you can see the raw RNA-seq data, the IBD data set consists of 434 data points, the sepsis data set consists of 18 data points, and the sepsis 2 data set consists of 45 data points.

So then we do a kind of between-sample normalization followed by gene filtering. and then followed by feature engineering, followed by ML gene selection, and followed by the performance evaluation. Okay, that was about the gene data. So, what if we actually have the protein data? So, protein sequence or protein structure data? So, a protein sequence embedding is one of the techniques, a featureization technique for the protein. So, why do we need to embed the protein sequences? Because protein sequences are usually in the FASTA format or raw strings of amino acids, we need to convert them into numerical representations for the ML models, and the embeddings capture structural, functional, and evolutionary information. So, it enables protein function prediction, drug target interaction modeling and enhances representation for protein ligand binding biomarker identification.

So, there are again multiple methods, such as one-hot encoding. K-Mer technique, where the frequencies of amino acid sub-sequences, like 3-Mer, such as ALA or GLY, are used. And then you have the ProtVec, which is a word-to-vec style embedding trained on k-mers; here it captures the local patterns. and then UniRep, which is our RNN-based learning representation capturing sequence-level biophysics information And then ESM, evolutionary scale modeling, which is a transformer-based embedding trained on millions of sequences. And then there is another kind of feature, which is the biological pathway feature.

So the pathway features represent gene proteins based on their role in biological processes. And then it adds biological interpretability and reduces the noise in high-dimensional omics data. Some of the common biological pathway sources are KEGG, the Kyoto Encyclopedia of Genes and Genomes, which consists of metabolic, signaling, and disease-related pathways. And then you have the Gene Ontology. And then you have Reactome and BioCarta, which are additional curated pathway databases. So, how do you know how to construct the features? So, feature construction approaches are like gene set enrichment, where we score the expression data against pathway gene sets, or the binary encoding gene or protein that is part of the pathway gives a value of 1 or 0 based on that.

And then pathway activity scores PCA or SSGSEA-based summarization of the gene sets.

And the embedding approaches we used are usually the graph-based representation of the pathways. And we use it for target identification based on pathway activity, predicting drug mechanisms of action, or for biomarker discovery and disease subtyping. And then we can also have these protein-protein interaction and gene-gene interaction networks as features. So, protein-protein interactions represent physical or functional associations between the proteins, and gene-gene interactions include co-expression, regulatory, or synthetic interactions. So, the network models are biological systems represented as a graph of nodes and edges, with the edges representing the interactions and the nodes representing the genes or proteins.

So, why shall we use them as features because it captures biological context, network influence, and the pathway-level effects. and it improves the ML model in target prediction, drug repurposing, and disease classification. So some of the feature engineering approaches that are being used in this case are the node-based features like degree, centrality, clustering coefficient, or page rank, and the embedding-based features like graph embeddings using node2vec, DeepWalk, or GraphSAGE. And the neighborhood aggregation, like the mean expression of connected genes, uses graph convolution for context-aware features. And it can be used, as you know, for predicting novel drug targets based on network centrality, inferring disease gene associations, and identifying essential genes or sub-networks for therapeutic targeting.

And then we can have advanced representation learning, as well. So, it is a process of learning feature representations directly from the raw data, which helps capture high-level patterns without manual feature engineering. So, and this is especially used in deep learning models and is important for complex tasks like drug discovery, target prediction, and activity modeling. These, you know, molecular embeddings are, so first, it is being embedded as a dense vector representation of a molecule or compound, which captures the chemical properties, functional groups, and molecular interactions. That can be used for predicting drug efficacy, toxicity, and target binding as well.

So, there are multiple techniques. So, two of the main techniques are autoencoders and graph neural nets. So, autoencoders are an unsupervised neural net for dimensionality reduction. The encoder learns a compact representation of the molecule, and the decoder reconstructs the original molecule in the form of SMILES; for example, its embedding is used for learning latent space and novel compound generation. And then you have the GNN, which treats molecules as graphs; we have talked a lot about this, and it captures the structural dependencies within the molecule. So, it enables node-level prediction and graph-level prediction, for example, the compound bioactivity.

Okay, coming to the end. So, there are some challenges and best practices; for example,

the problem of dimensionality is one of the challenges, as the number of features increases, the volume of the feature space expands, making it harder to find meaningful patterns. What is its impact, It leads to overfitting, poor generalization, and increases the computational cost. Particularly problematic in molecular datasets with hundreds or thousands of descriptors, fingerprints, or embeddings. So the solution is to reduce the dimensionality by using either PCA or t-SNE and feature selection techniques to identify the informative features. And then another issue is the information leakage; it occurs when data from outside the training set leaks into the model, leading to over-optimistic performance estimates.

So, the common cause of data leakage is using feature data for training; for example, in temporal drug discovery datasets, or using target-related information in features, such as including IC50 values as features for drug activity prediction. And then the impact of information leakage is that it leads to overfitting, poor model performance on unseen data, and inaccurate predictions in real-world applications. And the solution is to ensure proper dataset splitting, training-test validation, as well as cross-validation and stratified K-fold to avoid unintentional data leakage. Feature redundancy is another important issue.

So, it occurs when multiple features are highly correlated or provide the same information. So, the impact of feature redundancy can result in overfitting, reduce interpretability, and increase the computational cost as well. So, the solution is to use feature selection methods based on correlation thresholding or PCA for dimensional reduction, or regularization techniques like L1 and L2 to penalize redundant features. And then, you know, to ensure reproducibility in domain-specific feature choices. So, this is also important that we ensure reproducibility.

So, reproducibility is the ability to replicate the results of an experiment or model under the same conditions. So we already talked about certain challenges in drug discovery, such as data variability, complexity of models, and lack of standardization. So what we can do is the best practice is to do version control for data sets, models, and the code. And document preprocessing pipelines, hyperparameters, and training conditions seed random number generators for reproducible results. And the use of containerization, like Docker, to ensure consistent environments, open-source code, and data availability for transparency. While choosing the features, we need to use domain knowledge to identify features that reflect the biological mechanism of action.

Balance between the biological relevance and data availability, and be mindful of feature redundancy and the curse of dimensionality. So, coming to the summary, the data pre-processing ensures clean, consistent data sets by handling missing values, outliers, and scaling and normalizing the features. And feature engineering involves creating

meaningful input features, like the molecular descriptors or fingerprints, that enhance the model's performance. The techniques like PCA and feature selection help reduce dimensionality and remove redundant features. Proper data set splitting, like training, testing, and validation, ensures reliable model evaluation and prevents overfitting.

As well as domain-specific feature choices informed by expert knowledge, they are key to capturing relevant biological and chemical patterns for accurate predictions. I have some suggestions for further reading; you can go through these articles to get more information on this topic. And with that, thank you.