AI in Drug Discovery and Development

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

Week-09
Lecture-43

Welcome to the course "AI in Drug Discovery and Development." In an earlier session, we talked about many generative models that can be used for generating new molecules, which is called de novo drug design. In today's session, we will discuss benchmarking generative models for drug design. So, by the end of this lecture, you will be able to understand the importance of benchmarking in generative drug design and key evaluation metrics. Explain the role of benchmarking platforms like MOSES, GuacaMol, and MOLESCORE, as well as analyze the strengths and limitations of different benchmarking tools. So, in the early session when we were talking about those generative models, we saw that most of those models suffer from some limitations: the molecules are not novel, they are not valid, and the generated molecules are similar to the training set.

So, how do we actually know that? We need to benchmark those molecules and models to see how they are performing. So, benchmarking is a process of evaluating and comparing the performance of different models or algorithms using a set of standardized tasks and metrics. So, in the context of denatured design, benchmarking involves testing generative models on their ability to produce novel, valid, and relevant molecules, as well as their capacity to optimize for specific chemical properties and objectives. So why do we need to benchmark these de novo generative models? Because the chemical space relevant to drug discovery is vast, estimated to be between 10^24 and 10^60 possible structures, it is impossible to explore exhaustively.

We saw in the virtual screening that we cannot screen all the huge chemical space because it is so big, and we are limited to only a set of maybe billions of molecules we can screen. So, to check how these generative models are performing in terms of diversity and validity, we need to use benchmarking tools. The numerous generative models, including deep learning and classical algorithms, have been proposed, but they have often been evaluated inconsistently, making it difficult to compare their strengths and weaknesses. So, the standardized benchmarking will lead to fair comparison, it will provide a common ground to objectively assess and compare different models revealing which approaches are most effective for specific tasks. Identify strengths and weaknesses for each model.

It can help researchers understand where models excel, such as generating diverse molecules, and where they fall short, such as producing realistic or synthesizable

compounds. And then it will also drive progress, as seen in other ML fields like computer vision and image processing; that benchmark fosters rapid innovation by setting clear performance targets and highlighting areas for improvement. And it also guides the model development, where insights from benchmarking inform the design of future models and optimization strategies, ensuring that new methods address real-world drug discovery challenges. Without benchmarking, it is difficult to understand whether the new generative models offer any advantage over the existing methods or random sampling, and whether they produce molecules that are both novel and practical for synthesis and testing. So, coming to the key evaluation metrics, we have a lot of matrices that can be used to evaluate or benchmark the generative models.

So, they can be classified among similar distribution learning matrices. So, we are actually looking for the validity, which is the fraction of generated molecules with the correct SMILE syntax and valency. Uniqueness is another parameter, you know, which is the proportion of distinct molecules among the generated samples. Novelty is another parameter, which is the fraction of molecules not present in the training set. And then FCD, Fractured ChemNet Distance, measures the similarity between the generated and the training distribution via the ChemNet.

Then KL Divergence compares the physicochemical descriptor distribution, such as low pure TPSA. And then we have a goal-directed matrix as well, where we can use, for example, the Tanimoto similarity, which gives us the structural similarity to the target molecule using fingerprints, like the ECFP fingerprint. And then the rediscovery rate determines the success rate in generating an exact target molecule, like celecoxib. And then, isomer generation, which is the ability to produce molecules matching a specific molecular formula, and the median molecules, which are a multi-objective optimization of similarity to multiple targets. And then MPO, which is multi-parameter optimization, evaluates the combined property targets using the geometric arithmetic means such as logP or TPSA.

And then top case scoring is another parameter that gives us, you know, the aggregated performance across the top 1, 10, or 100 generated molecules. And then we have an extended distribution matrix as well, where we can say, for example, the internal diversity, which is the chemical variety within the generated set, using the Tanimoto similarity and the similarity to the nearest neighbor, which is the maximum Tanimoto similarity between the generated molecules and the training set. The fragment or scaffold similarity, which compares the Brics fragments or Bemis-Murcko scaffold to the training data, allows us to have the extended goal-directed matrix as well, along with synthesizability SA, synthetic excess, and synthesizability excess. Score, assess score can be one of the most important parameters that predict the ease of synthesis using the RDKits assess score, the retro tree,

or other models as well. And then the docking score, which evaluates binding affinity prediction, can be done with AutoDock, Vina, or other tools.

And then the multi-objective Pareto front assesses the trade-offs using the hypervolume indicators, and we have the three specific metrics. Which are, you know, like the geometric conformity where we can see whether a molecule is good at reproducing the 3D conformer of the molecule. So, we try to see whether the RMSD between the generated and the reference 3D structure and the pharmacophore match indicates that the model is able to generate a molecule with similar pharmacophoric features compared to the reference molecule, which represents the alignment with the target binding site features. And then practical feasibility metrics, such as paints or MCF filters, which are the filters for the pan-assay interference compounds or the medicinal chemistry risks, are considered. And retrosynthesis complexity, which predicts the synthetic steps via tools like AIZynth finder.

And then you have the drug-likeness score as well, which is, you know, the compliance with the quantitative estimate of the drug-likeness score or the Lipinski Rule of 5. So, these are the key evaluation metrics, an extended list of key evaluation metrics that are extensively used for benchmarking these models, and now we will see them one by one in detail. So, our validity is one of the first things. So, it measures the percentage of generated molecules that are chemically correct, meaning they obey the atomic valency bonding rules and are perceivable structures. So you can see here in this table, for example, that you have the valid spiles, and these are the invalid spiles.

So, while generating these valid smiles, the models also generate invalid smiles, especially in the case of those chemical language models, which are producing a lot of invalid smiles as well. Therefore, we need to see how good the model is at generating valid smiles. So, what it does is highlight the importance of the validity parameter. It is a kind of first quality check, where it tells us about the invalid molecules that cannot proceed to synthesis or evaluation. And then it is used for model assessment; high validity means the model understands the chemical syntax and is better at producing reliable and realistic molecules.

And then it saves resources by preventing waste in downstream screening or optimization as well. How do we do that? We use cheminformatics tools to parse generated molecules, such as RDKit. And then how do we calculate whether the percentage validity is equal to the number of valid molecules divided by the total number of molecules, multiplied by 100? And then there are tools like RDKit, ChemMol from Smile, OpenBabel, MolVS, and MOSES; we can use all of these tools to determine the validity of the generated molecules. Uniqueness is another parameter that measures the proportion of non-duplicate, valid molecules generated by a model; it indicates the model's ability to produce a diverse set of distinct chemical structures. So, the importance of uniqueness is that it prevents

redundancy in the molecular library and also ensures efficient exploration of the chemical space; a model that generates many duplicates is less useful for lead discovery and optimization.

Because the idea is that the de novo generative models can generate novel molecules, which can be assets for the IP of the company or the group that is working on the project. So, if that model is generating duplicate molecules or molecules that are very similar to the input molecules, then those tools are actually of no use. So, uniqueness is another important criterion that is used for evaluating or benchmarking deep generative models. So, how is it assessed? So, we compute the ratio of unique valid molecules to the total number of valid molecules generated, expressed as the percentage of unique valid molecules divided by the total number of valid molecules, multiplied by 100. So, molecules are usually compared using canonical SMILES or InChI formats.

There are several tools like RDKIT, MOSES, DEEPCHEM, CHEMBENCH, and GUACAMOL. So all of these can be used to determine the uniqueness of the generated molecules to benchmark the generative model. Another important parameter is Novelty. So, novelty measures the proportion of generated molecules that are not present in the training dataset, indicating the model's ability to explore new chemical spaces. So, the importance of novelty is that it is essential for discovering previously unknown compounds to have a high chance of obtaining IP.

High novelty helps avoid the rediscovery of known molecules and supports innovation in early-stage drug design. So, we tried to assess it by comparing the generated molecules using canonical SMILES or molecular fingerprints to those in the training set, and the percentage novelty is equal to the number of novel molecules divided by the total number of generated molecules multiplied by 100. And we can again assess it using RDKIT, MOSES, or, you know, Tanimoto similarity, which is used to confirm the novelty thresholds, okay. And then we have the Fréchet ChemNet distance (FCD). It measures the similarity between the distributions of generated molecules and the reference dataset by comparing the means and variances of molecular descriptors in both sets.

So, what if we wanted to generate molecules that need to be close to the structures of existing molecules, like our target—like we wanted to discover or generate molecules for acetylcholinesterase inhibitors, right? And then we have a data set that consists of maybe 50,000 acetylcholinesterase inhibitors, and comparing the structurally generated molecules with this data set involves something called the Fréchet distance. So, what it does is provide a measure of how well the generated molecules match the chemical diversity of real known compounds, and a lower FCD value indicates that the generated set is chemically similar to the target distribution, which is essential for drug-like properties. And we assess it using,

you know, the FCD to compute the Fréchet distance between descriptor distributions, like molecular fingerprint embeddings from the neural networks. And a lower value indicates high distribution similarity between the generated and reference molecules. So, we have several tools that can be used to evaluate this parameter, such as MOSES, which includes FCD as part of its evaluation matrix for generating models.

With ChemNet, we can use it to calculate FCD directly to compare generated molecules to real-world chemical space. A deep chem or RD kit can also be adapted to calculate the FCD. We are, you know, basically comparing the molecular descriptors of the reference dataset and the generated molecules. And then another important parameter is KL divergence, which is Kullback-Leibler divergence and measures the difference between two probability distributions: the distribution of generated molecules and the target or reference distribution. So, it quantifies how much information is lost when the generated distribution is used to approximate the target distribution.

So, the lower KL divergence value indicates that the distribution of generated molecules is closer to the target distribution, which is vital for molecular diversity and realism and helps assess the alignment of generative models with real chemical data. So, it is computed using this formula, where p is the distribution of real molecules and q is the distribution of generated molecules. So, again we have, you know, a MOSES, which includes scale divergence as one of its key evaluation metrics, and then we can use DeepChem as well as Psi4 for, you know, calculating this parameter. Coming to the Tanimoto similarity, it quantifies structural similarity between the generated molecules and a target molecule using a molecular fingerprint; typically, we use ECFP_4. So, the importance of the Tanimoto similarity is that it indicates how closely generated molecules resemble known bioactive compounds, and it is useful for guiding models toward desired chemical spaces or scaffolds.

So, we determine it by using this formula, where A and B are the binary fingerprint vectors for the generated molecules and the reference molecule. And then we can use different tools like RDKit, DeepChem, and even MOSES to determine the tiny motor similarity. Then the rediscovery rate measures the success rate of generating known target molecules from a benchmark dataset. For example, we will discuss GuacaMol later on. So, Celecoxib is one of the reference molecules in GuacaMol, and you can try to rediscover Celecoxib by using your generative model.

And then this will tell you whether your model is able to identify or generate the molecule that is already a well-established inhibitor or ligand for that specific target. So, it demonstrates the model's ability to recall or reproduce non-active compounds, and that is how it is useful for validating the model's performance on known benchmarks. So, we

assess it using the percentage of exact matches between the generated molecules and the target structures in a predefined list. So, we have the guacamole, and Moses also has it. You can use these two tools, and you can also use RDKit.

Of course, you have to write your own custom scripts for that, but it can also be done with the RDKit. Okay, another parameter is isomer generation, which assesses a model's ability to generate diverse structural isomers with the same molecular formula. So what it does is encourage chemically plausible diversity under strict constraints, and it is important for scaffold hopping and exploring the local chemical space as well. So we can count the unique valid molecules generated that share the same molecular formula but differ in structure. So, we can use RDKit, MolVS, or OpenBabel to evaluate the model based on this criterion.

Then another parameter is medium molecules, which evaluates how similar the generated molecules are to multiple reference structures, typically using a similarity matrix such as the Tanimoto coefficient. So, it helps balance the multi objective generation if you wanted to generate a dual target inhibitors. So, then this this you know parameter can be of help. So, it also encourages chemical solutions that are not biased towards a single target as well. So, how do we assess it? We try to evaluate the medium of similarity scores across multiple reference molecules.

Then we can use MOSES, GUACAMOLE, or the RDKit again to evaluate this parameter. And then multiparametric optimization, because many times when we are working on lead optimization, our objective is to optimize the molecule across multiple parameters, such as having a molecule that is soluble, metabolically stable, permeable, and has a high affinity for the target. So, these parameters combine multiple physicochemical or admitted properties into a single score to guide drug-likeness and property-based optimization. So, the importance of MPO is that it is essential for generating molecules that meet real-world drug design criteria, as well as avoiding the optimization of a single property at the expense of others. Because what happens many times is that if we try to improve the potency, the solubility or permeability decreases.

So, balancing all those properties at the same time is a big challenge and this parameter is good at evaluating that. So, how do we assess it? It can be computed using the geometric or arithmetic mean of the normalized values of all those parameters. And then we can maybe use ADMETLAB, the RDKit MPO scoring function, or DEEPCHEM to evaluate this parameter. And then coming to TOP-K scoring, which evaluates model performance based on the quality of its TOP-K generated molecules according to a predefined scoring function. So, the importance of top-case scoring is that it reflects practical use, as only top hits are pursued in real drug discovery campaigns.

It balances quantity with quality in generation because many times when we are generating molecules, we use graph invention, and it generates 10,000 molecules. So, we are not going to take up all those molecules that have been generated. Instead, we will try to rank them based on some parameters and use the top 10, top 100, or top 1,000 molecules. So, this parameter is going to evaluate that criterion. How do we assess it? So, we rank the generated molecules and report performance on the top 1, 10, or 100 based on a custom or predefined scoring matrix, and then there are tools like GuacaMol and MOSES, or you can also use the custom scoring frameworks to evaluate this criterion.

And then coming to another parameter, which is internal diversity: internal diversity measures the chemical variety within the generated set by calculating pairwise Tanimoto similarity between the molecules in the set. So, the importance of internal diversity is that it evaluates how structurally diverse the generated molecules are and helps avoid further collapse, where many similar molecules are generated. And we have seen that it was, you know, one of the challenges or disadvantages of most generative models. And it also encourages broad exploration of the chemical space. So, how do we assess it? So, it is calculated as one minus the average pairwise Tanimoto similarity among all the generated molecules.

And then we can use tools like RDKIT, MOSES, or DEEPCHEM to determine the internal diversity. Okay, another parameter is the similarity to the nearest neighbor. So, it measures the maximum similarity of each generated molecule to its closest match in the training set using Tanimoto similarity on ECFP_4 fingerprints. So, the importance of, you know, SNN is that a lower SNN indicates mobility and the model's ability to generate molecules that are distinct from the training data, while a high SNN value suggests overfitting or limited generalization. So, how do we assess it? So, for each generated molecule, we compute the Tanimoto similarity to all training molecules, take the maximum or the nearest neighbor, and then average these maximum similarities over the generated set.

And then we can use it; we can do it by using the RDKit process or the GuacaMol tools. Then, fragment or scaffold similarity compares the distribution of substructures in generated molecules, either Briggs fragments or the Bemis-Murcko scaffold, to those in the training set. So, we compare the fragment's similarity with the training set. And the importance of this is that it evaluates whether the model is capturing meaningful substructural patterns, and it also supports the understanding of chemical space coverage and synthetic realism as well. So, how do we assess it? So we extract the BRICS fragment or Bemis-Murko scaffold from the generated molecules, and then we compare the distribution overlap with those from the training set using the Jaccard or overlap matrix.

So we can use RDKit, which has the Briggs or Bemis-Murcko functions, and then we can use MOSES or the Chemdesk library as well. Synthesizability is another very important parameter. And as we discussed, if a model is generating new molecules that are not synthesizable, then they are actually of no use. So, synthesizability assesses how easily a molecule can be synthesized in a laboratory setting, typically using scoring functions such as the synthetic accessibility score. The importance of this assessment score is that it ensures the generated molecules are practically accessible, not just theoretically interesting.

And it encourages models to generate drug-like and synthetically feasible molecules. How do we do that? So, we can get this SA score from RDKit as well, which combines molecular complexity and the fragment contributions. So, the lower score indicates an easier synthesis; scores can range from 1 to 10. Okay, and then we have a retrotrae or ASKCOS, which offers a more advanced retrosynthesis-based evaluation. And then we have the AIZynthfinder as well, which is, you know, from AstraZeneca.

Okay, and then the docking score, which is especially useful for generating those molecules, is used in structure-based de novo drug design. So, the docking score estimates the binding affinity of generated molecules to a biological target, typically through molecular docking simulations. The importance is that it links generation with biological relevance because if a molecule can fit into the binding pocket of the target or bind to the target with greater potency, then it will be of high utility. And it also helps identify functionally active compounds early in silico and is used in goal-directed generation to optimize molecules for high binding affinity. How is it assessed? We dock each generated molecule to a target protein using software like Autodock, Vina, Gnina, or Glide and evaluate them based on the binding free energy, where a more negative docking score means stronger predicted binding.

And then we can use, as I said, Autodock Vina, Gnina, Glide, Smina, or there are plenty of docking tools that can be used to assess whether the generated molecules can bind efficiently to the target of interest. And then a multi-objective Pareto front: Pareto front analysis evaluates how well the model balances multiple objectives like potency, solubility, and synthesizability by identifying non-dominated solutions. So, it is important because drug design is inherently multi-objective, and we have talked about it a lot. We need to optimize multiple properties during the load lead optimization process. So, it encourages the generation of balanced molecules rather than extreme values in a single property, and it also allows for trade-off analysis across competing properties.

How do we do it? So, we compute a matrix for each objective per molecule and use Pareto ranking or compute hypervolume indicators to assess the spread and balance of the solutions. So, the larger the hypervolume, the better the coverage of the optimal trade-offs.

So, we can use Deep, which is a Python library for Pareto optimization, or Pymoo, or GuacaMol, or we can even run custom scripts using RDKit and then evaluate these molecules based on these parameters. And then geometric conformity, which measures how closely the 3D geometry of a generated molecule matches an unknown or reference conformation, is typically measured using the RMSD. So, the importance is that we ensure three-dimensional structural accuracy.

which is critical for molecular recognition binding and it also validates that generated molecules are not only chemically correct, but they are especially realistic as well. So, how do we do it? So, we align the generated 3D conformation with the reference structure and calculate the RMSD between the atomic positions. So, the lower RMSD values of less than 2 Å indicate high structural similarity. Therefore, we can use RDKit again for conformer generation and RMSD calculation. Or we can use the open bevel, or we can use pie mode, and there are plenty of other tools as well that can be used to evaluate the geometric conformity of the molecules generated from the de novo generative models.

And then the pharmacophore matches another parameter related to the 3D generation. So, it assesses how well a generated molecule matches key pharmacophoric features, such as hydrogen bond donors, acceptors, and aromatic rings, of a known active compound or binding site. So, the importance of this is that it highlights functional similarity to bioactive molecules and is also essential for target-specific design in lead optimization. So how do we assess it? We define or extract pharmacophoric features from the targets or active molecules, and then we compare spatial alignment and presence in the generated structures. We evaluate the fit score or the feature overlap, and we have a lot of tools for that.

We can use LigandScout and Phase, which is a tool you know from Schrödinger, and then we can also use RDKit and farm it as well. So these are all the pharmacophore modeling tools that can be used to generate the pharmacophores for the reference molecule and the generated molecule, and to compare both: PAINS and MCF filters. So, these are quite important because many times we see molecules that are shown to be active, but they are not actually active in almost every assay. And why? Because they interact with those you know—assays, actually.

So, the PAINS filter: Pan Assay Interference Compounds filter. So, it can screen generated molecules against known toxicophores or problematic substructures, such as PAINS or the MCF medicinal chemistry filters. So, its importance is that it prevents the pursuit of false positives in screening, as well as improving the quality and safety of generated chemical libraries. So how do we assess it? So we match substructures using SMARTS patterns defined in the PAINS or MCF libraries, and we flag or remove molecules that trigger these filters. And then there are tools like RDKit, SwissADME, MolVS, and FAF-Drug4 that

can be used for filtering those molecules using PAINS or MCF filters. And then, retrosynthetic complexity estimates the number and difficulty of the synthetic steps required to produce a molecule using the retrosynthesis planning tools.

So it is a little bit different from the synthetic accessibility score because synthetic accessibility only tells us whether those molecules are synthesizable. And then, this tool typically tells us whether these molecules can be easily synthesized. So they overlap, but are a little bit different from each other. So it ensures that the generated molecules are accessible for lab synthesis, and it aids in prioritizing compounds for real-world drug development as well. So we run retrosynthetic analysis from the product to the precursors, and then we score based on the number of steps available to the building blocks and the reaction confidence.

And then, these are some of the tools that can be used, like AI Zynth Finder, ASxKCOS, IBM RxN, and RetroTrae. And then drug likeness, which is another, of course, very important feature, evaluates how closely a molecule adheres to non-drug-like chemical space. Often, it is done via QED, which is a quantitative estimate of drug-likeness, or Lipinski's Rule of Five. So, it helps identify bioavailable, stable, and safe molecules, and filters out unlikely drug candidates very early on. Okay, so we do it using QED, which is a composite score based on molecular weight, log P, PSA, H-bonding, etc.

Or we can use Lipinski's rule, which filters out compounds that violate more than one of the five rules of the criteria. And then we can use RDKit again, which has this QED or Lipinski's rule checker, and Swiss ADME or Molsoft can be used to determine drug likeness. Okay, so those were, you know, the parameters that were used for evaluating those tools. So now we will talk briefly about some of those benchmarking tools, which are either servers, programs, or algorithms that can help us evaluate all those parameters in generative models.

So MOSES is one of them, actually. It is known as Molecular Sets, a benchmarking platform developed to evaluate generative models for de novo molecular design. So, it provides a sterilized data set and a suite of robust evaluation metrics to enable fair comparison across models. So, it has a curated data set based on the zinc clean lead data set containing 1.9 million drug-like molecules, which are filtered to remove undesirable properties like pain or reactive groups and then focuses on properties relevant to lead discovery. So, it has a benchmarking suite, which is a comprehensive set of quantitative metrics to assess generated molecules, including validity, uniqueness, novelty, FCD, SNN, scaffold similarity, internal diversity, property distribution, and so on.

So, we talked about all these parameters, and MOSES is able to evaluate your generative

model based on them. And then it also has the baseline generative models, which include several ready-to-use models like variational autoencoders, AAEs, and Char-RNNs. latent GAN which serves as a baseline for comparing your own molecular generators. And then another important benchmarking tool is GuacaMol. So, which is specifically for goal-directed molecule generation? So, it is from benevolent AI that a benchmarking suite has been designed to evaluate molecular generative models on both distribution learning and goal-directed generation tasks.

So, it provides a common ground for comparing the performance of models in drug-like molecule generation. So it is based on the ChEMBL database, which contains around 1.6 million molecules, and these are preprocessed to ensure drug-likeness and synthetic accessibility. It uses the SMILES format throughout for compatibility, and there are two benchmark types. One is a distributed learning benchmark that evaluates how well a model can learn and reproduce the distribution of known molecules.

And it includes validity, uniqueness, novelty, KL divergence, FCD, and also has goal-directed benchmarks that test a model's ability to generate molecules optimizing specific properties like similarity-based tasks, physicochemical property targets, isomer generation, and multi-objective optimization, etc. So, it is one of the most popular benchmarking tools for de novo generative drug design. So, we have another important benchmarking tool called the MOL score. So, it is a flexible, modular benchmarking framework designed to evaluate goal-directed molecule generation and optimization algorithms. So, it was developed to address the limitations in reproducibility and customization seen in earlier benchmarks like GuacaMol.

MolScore provides a unified interface to configure, run, and analyze molecular optimization tasks. So, we have a customizable benchmark here in MolScore. So, it can easily combine physicochemical descriptors, QSAR models, structural similarity, and penalty functions into multi-objective scoring functions, and it also integrates custom scoring components using a consistent API. So, it has a modular design built with Pydantic for validation configuration which is fully extensible for you know adding new scoring functions or generators. And it uses out-of-the-box coding modules like LogP, QED, TPSA, Synthetic Accessibility, Similarity Metrics, Fragment Penalties, etc. and supports external predictors via APIs or custom wrappers.

And we can also customize the generative models. It is compatible with generative models as well, works seamlessly with genetic algorithms, Bayesian optimization, and RL-based workflows, and supports SMILES, SELFIES, and graph-based representations. And it is also, you know, good at logging and analysis, as well as providing a standardized output format for tracking performance over generations. It has tools for visualizing molecular evolution, scoring trends, and diversity.

Now we see all those parameters, as well as some of those benchmarking tools. So let us talk about some of the challenges in benchmarking DDoS generative models. So one of the big challenges is the limitation of the evaluation metrics, such as the synthetic feasibility gap. Models often prioritize matrix-like validity and novelty but ignore synthesizability, leading to impractical molecules. An overreliance on proxy scores, like benchmarks, uses simplified properties such as QED or LOGP that do not reflect the complexities of real-world drug discovery.

Because, in reality, drug discovery is highly complex and challenging. And we cannot simply say that, okay, if a molecule satisfies the QED or logP, it is going to work. And then there is a bias in the distribution matrix, as well. The FCD and KL divergence may favor chemically unrealistic distributions that superficially mimic training data. And then there are challenges with model overfitting and data bias as well. Where the training set dependency exists because the models trained on datasets like ChemBL may reproduce known molecules rather than explore the novel chemical space.

And then, there is also a copying problem where some models exploit trivial modifications, like adding carbons, to artificially boost the novelty score without meaningful innovation. Then there are challenges with the optimization versus the real-world relevance because these are, you know, kind of a disconnect from wet lab validation. High-scoring molecules in silico often fail experimental tests due to unmodeled factors such as toxicity or metabolic stability. And then they have limited multi-objective benchmarking as well because most tasks focus on a single property. Ignoring the complex trade-offs required in drug design, where, you know, you have to, again, balance multiple parameters at the same time.

And then there are some algorithmic trade-offs as well, where there is an exploration-exploitation imbalance; for example, genetic algorithms excel at optimization but produce low-quality molecules, while neural network models balance quality and performance. And then mode collapse, which was one of the major issues with those models, such as GAN-based architectures, struggled to generate diverse molecules, scoring poorly on the distribution learning task as well. And then the benchmark design flows like a trivial task; saturation and simple objectives, like logP optimization, are easily solved by all models, failing to differentiate the true innovation. The lack of standardization, inconsistent task definitions, and scoring methodologies across studies hinders fair model comparisons. Okay, and then coming to the summary, benchmarking is essential for assessing the performance and reliability of generative models in drug discovery.

Core evaluation metrics include validity, uniqueness, novelty, synthesizability, drug-

likeness, and diversity. And standardized platforms like MOSES, GuacaMol and MolScore, they enable fair comparison using the curated dataset and metrics. And future benchmarking efforts should also incorporate multi-objective scoring and experimental validation. And this ensures that the generated molecules are not only computationally promising but also practically viable.

And only then can we, you know, utilize those excellent models to discover new drugs. So I have some suggestions for further reading that you can go through if you are interested in learning more about this topic. And then I have some suggested readings and references that you can go through if you are interested in learning more about this topic. And with that, thank you.