Nazarbayev University

Final Project Report

ML-based Prediction and Generation of Fluorescent Molecules and Their Properties

CSCI409 Group: 30

Team members: Miras Nurkin, Dias Kuatbekov, Kuanysh Akhmetzhanov

Advisor: Professor Siamac Fazli, PhD

2024

Executive Summary

This report provides an in-depth overview of the "ML-based Prediction and Generation of Fluorescence Molecules and Their Properties", which dives into the usage of machine learning techniques to accelerate the discovery process of new fluorescence molecules and predict their properties efficiently. This project aims to reduce the time and costs associated with discovery and property prediction of organic compounds.

A key component of our work was the use of generative models, including Transmol and Chemical Language Model and the usage of deep learning architectures for property prediction. The results of our approaches have been highly promising. We achieved nearly state-of-art performance in predicting molecular properties and constructed a database of newly sampled molecules. It is worth mentioning that with ensemble learning techniques, we attained an R^2 score of 0.956 for absorption max, 0.901 for emission max, and 0.73 for quantum yield. It demonstrates the high performance of our models.

During working on this project, we have learned different domains, such as RDKit for cheminformatics, various molecular representations, and the integration of generative models with classical machine learning algorithms to improve property prediction and synthesis.

Finally, we have developed a database containing newly sampled molecules and with their properties, as predicted by our best performing models.

Introduction

Fluorescent molecules are integral in diverse applications, such as optoelectronics, organic light-emitting diodes (OLEDs), bioimaging, chemical sensors and dyes. Consequently, development of these molecules is continuously pursued due to its importance. Traditionally, the discovery and development of fluorescent molecules are based on trial and error methods. Moreover, the property prediction of the optical properties of such molecules are based on theoretical calculations using Density Functional Theory, which requires high computation power (Greenman et al., 2022).

Considering the challenges, there is a need for alternative methodologies that can accelerate the discovery of fluorescent molecules and property predictions. Machine learning is a promising solution, having demonstrated success in similar scientific domains. If applied successfully, machine learning can streamline the process that this project aims to achieve.

Our project utilizes a combination of classical machine learning and deep learning techniques. We have developed regression models trained on a dataset of fluorescent molecules to predict key properties, such as absorption and emission maxima, quantum yield and lifetime. Simultaneously, we worked on generative models to synthesize novel molecular structures.

This report is structured to cover two main aspects of the project. For property prediction, we explore methodologies involving various vector encodings such as Morgan Fingerprints, Avalon Fingerprints, and Molecular ACCess System Fingerprint. The prediction model employs both classic machine learning algorithms including linear regression, lasso regression, KNN regression, and ridge regression, and a deep learning algorithm, specifically a Graph Neural Network. In the section of molecule generation, we discuss our use of two pre-trained models: Chemical Language Model (CLM) and Transmol. Moreover, as a final part of the work, we have predicted the properties of newly synthesized molecules.

Background and Related Work

The development of fluorescent molecules is a rapidly evolving field, underscored by the integration of computational methods to improve the efficiency and accuracy of molecular discovery and property prediction. Our project builds upon a foundation of existing research in molecular representations, regression and generative models, and databases of optical properties.

Initially, we conducted a literature review focused on the different molecular representations and the overview of fluorescent molecules. This review helped us understand the current landscape of regression models and generative models applied in this field.

For our primary dataset, we used the "Experimental database of optical properties of organic compounds," which was first published in Nature Scientific Data in 2020. This database comprises 20,236 different chromophore-solvent pairs with their optical properties (Joung et al., 2020).

To get a better understanding of effective strategies in the domain, we have looked at similar research works. Partially, the work by Ye (Ye et al., 2020) on emission max, Ju (Ju et al., 2021) on quantum yield, and Wang (Wang et al., 2023) provided valuable reference points. These studies were helpful in understanding molecular representations, algorithms and techniques that proved to be effective in property prediction.

In the review of molecular generative models, we have looked at several works. One such work is "A biologically-inspired multi-modal evaluation of molecular generative machine learning," which was recommended and co-authored by our advisor (Vinogradova, 2022). This paper guided our decision to focus on the Chemical Language Model (Moret et al., 2020) as a candidate model. Additionally, we employed the Transmol model, which was developed by alumni from our university (Zhumagambetov et al., 2021)

Our investigation into various forms of molecular representations led to the selection of three specific types for our project:

1. Morgan Fingerprints: This is an industry standard to encode molecular structure for machine learning models.
2. Avalon Fingerprints: Known for capturing chemical specificity.
3. MACCS Keys: A standard tool in cheminformatics, useful for its simplicity and effectiveness.

The results of different molecular representations are summarized in the table below:

Dimension	Name	Type	Method	Advantages and Disadvantages	Year	Link to the publication
1	SMILES	Linear	Rule-Based	+ most common used representation + Simple to use - Generative models issues (requires certain preprocessing)	1980s	Weininger, 1988 Link
1	Deep SMILES	Linear	Rule-Based	+ more compatible for Machine learning tasks	2018	O'Boyle &

				+ removed paring rings closure and unbalanced parenthesis problems		Dalke, 2018 Link
1	International Chemical Identifier	Non-Linear	Rule-Based	+can represent more complex structures, including stereochemistry -more complex and computationally constantly -might not capture tautomeric forms	2015	Heller et al., 2015 Link
2	A radial distribution function description	Graph	Descriptor-Based	+more empathize on spatial distribution of atoms -limited pairwise information -computationally demanding	1976	Schutt et all., 2015 Link
2	Bag of Bonds	Vector	Descriptor-Based	+better performance for machine learning models +simplicity -loss of spatial information and oversimplification	2015	Hansen et al., 2015 Link
2	Adjacency matrix	Matrix	Graph-based	+explicit structural representation +universality and standardization -scalability issues -insensitive to 3D geometry	1996	David et al., 2020 Link
2	Morgan Fingerprints	Vector	Descriptor-Based	+rich information content +scalability -collisions	1965	Morgan, 1965 Link
2	Avalon Fingerprint	Vector	Descriptor-Based	+customizability +structural encoding -computationally costly	2006	Gedeck et al., 2006 Link
2	Extended Connectivity Fingerprints	Vector	Descriptor-Based	+detailed representation of molecular topology +adaptability for ML -Information loss	2010	Rogers & Hahn, 2010 Link
2	MinHash	Vector	Descriptor-	+efficient for large chemical	2018	Probst

	Fingerprint		Based	spaces +high sensitivity for molecular diversity -hash collision -dependencies on chemical similarity searching		& Reymond, 2018 Link
2	Hierarchical Organization of Spherical Environments	Hierarchical Code	Descriptor-Based	+high predictive accuracy +useful for structure elucidation -limited global information	2019	Kuhn & Johnson, 2019 Link
2	Bond Graph Linear Fingerprint	Linear	Descriptor-Based	+detailed bond focus +suitable for ML -loss of 3D structure information -dependencies on accurate bond representation	2011	O'Boyle et al., 2011 Link
2	Neural Fingerprint	Vector	Descriptor-Based Neural Network-Generated	+adaptive representation +high dimensionality and depth -computationally intense -Black box nature	2015	Duvenaud et al., 2015 Link
2	Seq2seq fingerprint	Vector	Deep-Learning	+dynamic representation +unsupervised -development complexity	2017	Xu et al., 2017 Link
2	Molecular ACCess System (MACCS)	Vector	Substructure-Based Descriptor	+easy of use +efficient comparisons -potential for collisions	2002	Durant et al., 2002 Link
2	Mol2Vec	Vector	Deep-Learning	+high-quality molecular representation +transfer learning capability -dependencies on training data	2018	Jaeger et al., 2018 Link
2	CHEM-BERT	Transformer-Based	Representation-Learning	+high performance on chemical tasks +transfer learning +advanced contextual understanding	2020	Chithrananda et al., 2020

				-computationally intense -complexity in implementation		Link
2	SMILES-BERT	Transformer-Based	Representation-Learning	+versatility in chemical predictions +enhanced contextual understanding -high resource demand -dependencies on SMILES data	2019	Wang et al., 2019 Link
2	Knowledge-Guided Pre-training of Graph Transformers	Graph Neural Network	Knowledge Enhanced	+Enhanced representation learning +improved generalization -complexity in implementation -computationally demands	2021	Zhao & Zeng, 2022 Link
2	SMILESVec	Vector	Embedding-Based	+scalability and efficiently +leverages NLP techniques -dependencies on SMILES data -potential information loss	2018	Öztürk et al., 2018 Link

Table-1. Molecular Representation

In order to start working on the project, we also had to understand more robust machine learning and deep learning algorithms. More detailed information about these algorithms will be provided in the Project Approach section.

Project Approach

The system consisted of two main components: ML-based prediction and Generative models. The high-level architecture of the system is shown in Figure-1. The original database will be used to train machine learning algorithms and generative models. Molecules that are newly generated by the generative models are then input into the predictive models, allowing for their optical properties prediction.

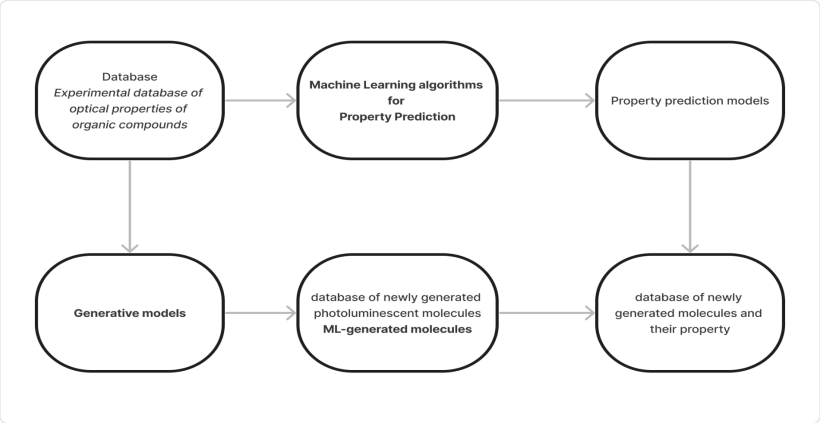


Figure-1. High-level overview of the project

The predictive model pipeline is detailed as follows: initially, the dataset is divided into training, testing and validation subsets. Various molecular representations are then used for training with different machine learning models, creating model-representation pairs for each target feature. Additionally, we applied randomized cross-validation for hyperparameter tuning. Finally, the three most effective model-representation pairs are selected and integrated using ensemble learning techniques, where the outputs from each model are averaged to produce a final result. The pipeline is shown below in the Figure2.

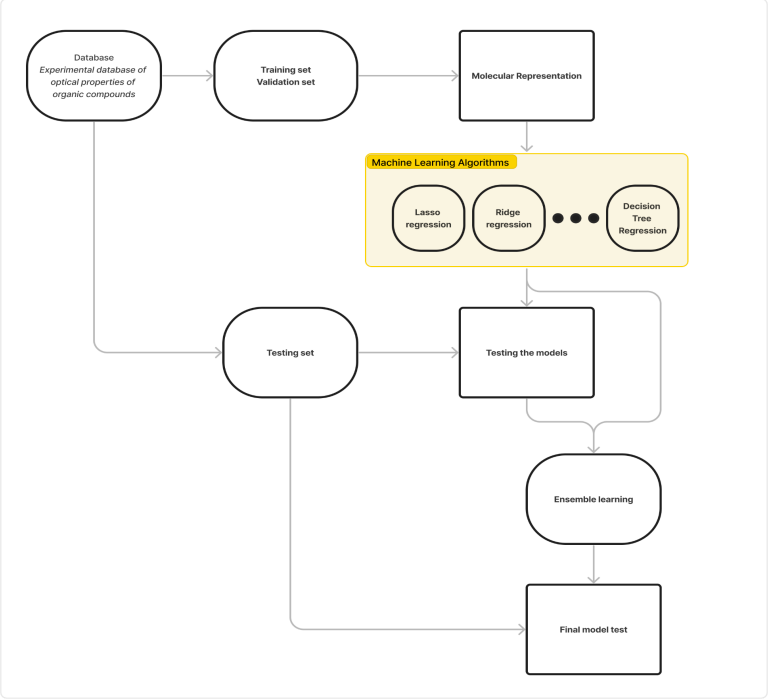


Figure-2. Predictive model pipeline

RDKit is the main third-party component that our project is based on. RDKit is an open-source toolkit for cheminformatics. We use this library to create molecular representations and obtain vector embeddings for our models.

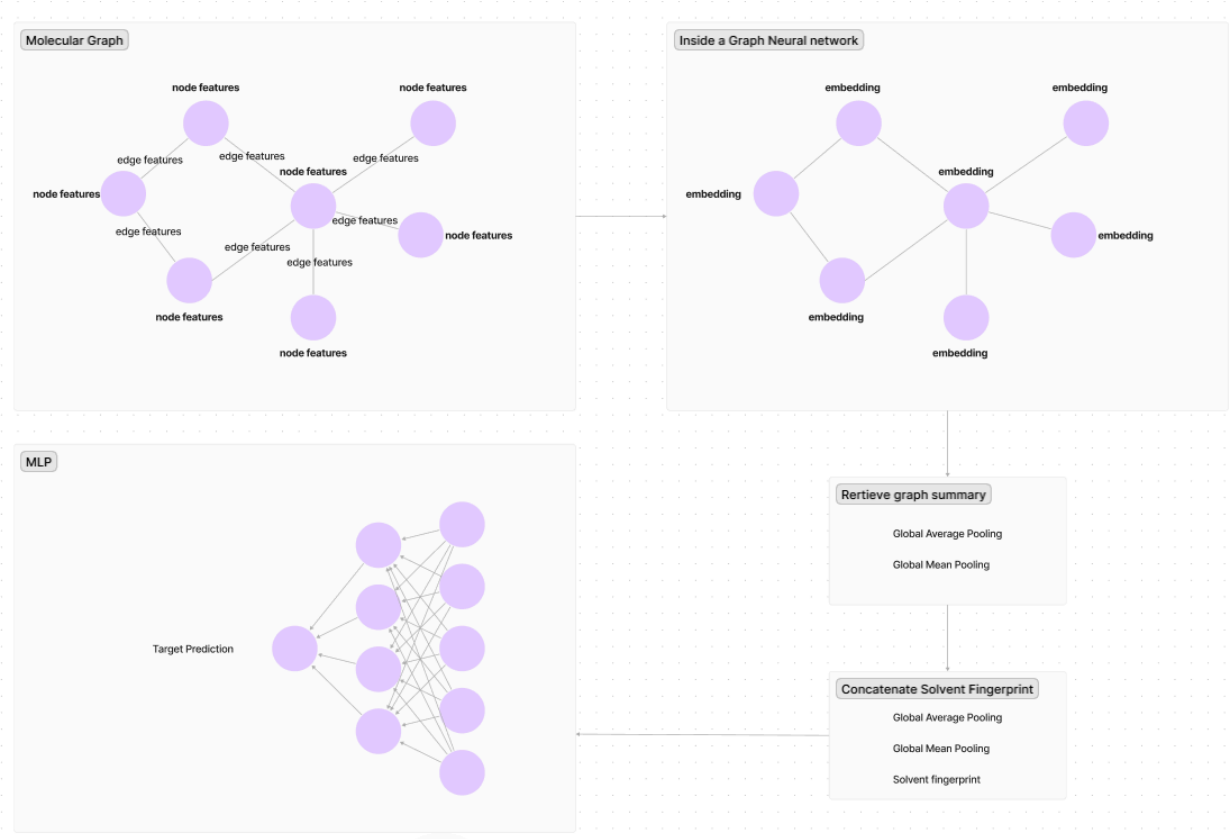
For the predictive analysis, we employed a vast range of classical machine learning models.

Technique	Description
Lasso Regression	Linear regression model that uses shrinkage, meaning the data values are shrunk towards a central point. It uses a regularization term to the loss function, thus reduces overfitting
Ridge Regression	Regression techniques that are used dealing with regression data suffering from multicollinearity. It incorporates a regularization term to the loss function, preventing overfitting.
Random Forest Regression	Ensemble learning technique by building a multitude of decision trees at training time and outputting the mean prediction of the individual trees.
Multilayer Perceptron	MLP functions by forwarding input data through the layers and each neuron processes the input by applying a weighted sum and a nonlinear activation function. By backpropagation, weights are adjusted to reduce the loss function.
KNN Regressor	Property prediction is done based on proximity in feature space. By analyzing ‘k’ closes examples from the training dataset, target value is found by averaging neighbors.
XgBoost Regressor	Ensemble technique that builds multiple decision trees sequentially, each tree correcting errors made by the previous ones by focusing on minimizing a loss function through gradient descent.
Decision Tree Regression	Tree-line structure is built by segmenting the dataset into smaller subsets based on the feature values. The target value is found by navigating the branches until it reaches the leaf node.

Table-2 Classical Machine Learning Models

We moreover used deep learning approaches, namely Graph Neural Network. To explore graph-based approaches for molecular property prediction, we implemented a custom design for a GNN. The graph model learns embeddings of a chromophore from its graph representation, and uses an Avalon

descriptor of the solvent to predict its physical properties. Any molecule can be represented as a graph if we treat its atoms as graph nodes, and bonds as graph edges. We can also add descriptions into nodes, which can include but are not limited to an atom's charge, degree and hybridization. Descriptors in molecular graph edges generally include one-hot encoded bond type, stereo configuration of a bond, and information about if the atoms are in the same ring and conjugated. These kinds of operations can be done using a library called DeepChem. DeepChem provides a MolGraphConv featurizer that returns a molecular graph with all most important node level and edge level descriptors.



Figure–3. Overview of Graph Neural Network

Our model consists of 2 transformer based convolutional layers to learn a representation of a molecular graph. We use both global average pooling (GAP) and global mean pooling (GMP) to extract most relevant information from the graph. If we concatenate GAP and GMP results, we obtain a vector that summarizes a chromophore's structural information. We use that vector along with the solvent's Avalon descriptor for molecular property prediction.

Talking about generative models, we used a dataset to fine-tune existing molecular generative models. Transmol is a variant of the transformer architecture, developed for natural language processing, adapted for the task of generation of the new molecules. Transmol can be used as either input-guided or diversity-driven generation modes by applying traditional one-seed or a novel two-seed approach. For our use case, we used Transmol using a one-seed approach, because one-seed approach allows targeted generation of new molecules as the model will strive to replicate the input or in our case fluorescent molecules that we will provide to the model , hence input-guided mode.

Transmol uses a vanilla transformer architecture that has two parts: an encoder and decoder. The encoder maps input to a latent representation z. The decoder accepts z as input and produces one symbol at a time to generate molecules. The notable attribute is the use of attention mechanisms throughout the model instead of recurrent or convolutional layers. Self-attention allows the model to selectively focus on the most important parts of the input.

For generation, a seed SMILES string provides context to the decoder. The decoding process begins by inputting a special starting symbol. Subsequently, the decoder produces an output, and the first symbol is generated. To obtain the next symbol, the previously generated characters are fed back into the decoder. The decoding process continues until either the decoder outputs a special terminal symbol or the maximum length is exceeded. The decoder outputs are converted to SMILES characters using techniques like greedy search or beam search.

Alternatively, we also focused on the Chemical Language Model for molecular generation. It is based on RNN, namely long short-term memory units. CLM ensures generation of chemically valid structures through the learned syntax of SMILES notation. The generative process begins with the encoding of molecules into a one-hot vector format, representing the SMILES strings of known bioactive molecules. The model then learns the distribution of molecular features from this data, enabling it to predict new molecules by sampling from this distribution.

Project Execution

Over the past two semesters, our team has faced numerous challenges and made several key design decisions during the execution of our capstone project. The project was structured into two main phases: property prediction in the first semester and molecular generation in the second. Initially, we faced a lack of expertise in computational chemistry. However we received invaluable help from Professor Dr. Vsevolod Peshkov from the Chemistry Department and we want to acknowledge his support.

Suitable molecular representations and null value problems.

Our first major task involved exploring various molecular representations to select the most suitable ones based on our capabilities and available resources. We decided to use molecular fingerprints, such as Morgan, Avalon and MACCS fingerprints from RDKit to capture the molecular structures.

Each team member focused on different molecular representations to develop regression models. Simple models such as Lasso, Ridge, and Linear regression performed well for predicting absorption and emission maxima. However, predicting quantum yield and lifetime proved challenging due to a high prevalence of null values in the data:

- Absorption max (Float): 2941 null values
- Emission max (Float): 2094 null values
- Quantum yield (Float): 6399 null values
- Lifetime (Float): 13726 null values

To address these challenges, especially with quantum yield, we used sophisticated models, such as Graph Neural Networks and implemented ensemble learning techniques, combining outputs from various models to enhance prediction accuracy. Another critical aspect was integration of solvent data, as

optical properties are significantly influenced by solvents. We used vector concatenation to combine solvent with chromophore data in our models.

Graph Neural Network design choices and challenges.

Regarding Graph Neural Network, it was one of significant milestones in our project. For this to happen, we had to research how to represent molecules in graphs, construct a graph neural network that could treat both chromophores and solvents, and find optimal hyperparameters for our model.

The most significant among those was designing a graph neural network. Initially, we attempted to construct a GNN that could treat both chromophores and its solvents as graphs but the idea was abandoned due to major complications with regard to implementing such a model in PyTorch Geometric. Another complication was that such an approach would significantly increase the power of a model, and would not be a proper approach considering the limited size of our dataset. Instead, we decided to treat chromophores as graphs, and represent their solvents as Avalon fingerprints. Apart from that, PyTorch Geometric provides a rich choice of layers for treating molecular graphs. By means of experimentation, we decided to stop at TransformerConv layers.

Training graph-based Deep Learning models is computationally expensive. This made experimenting with our GNN a time consuming process, especially for searching hyperparameters through bayesian optimization.

That being said, we made a lot of errors when designing a GNN. One example is when we tried to add regularization methods to our model by implementing dropout before batch normalization. However, bayesian search results revealed it as a significant mistake. Only after that we discovered that this is a well-studied phenomena, and a sane person would never use dropout before batch normalization.

In order to improve the results of the models, we additionally applied ensemble learning techniques. That also required us to split the dataset similarly while training the component models to ensure that there will be no overlapping of test data in training in any of the models.

Generative models.

In the second semester, we focused on generative models, particularly the Chemical Language Model (CLM) and Transmol models. The CLM presented challenges due to its development in an older version of TensorFlow, which did not support GPU acceleration by default. That resulted in taking over 2-3 hours per each epoch. Efforts to adapt the CLM to GPUs were unsuccessful due to compatibility and dependency issues with newer TensorFlow versions. It was recommended by authors of the paper to train for at least 40 epochs, which proved to be impossible as it was time-consuming even on the laboratory computers.

Secondly, we encountered challenges with the integration of solvents, which, as previously mentioned, play a crucial role in determining optical properties. We faced a decision between two approaches: training generative models on all chromophores and later integrating the solvent to determine properties, or subsetting the database based on the solvent and training the generative model solely on that subset of chromophores. For the CLM, we opted for the latter approach, while for Transmol, we implemented the former.

Lastly, we have faced the problem of integrating the MOSES benchmark to evaluate the performance of Chemical Language Model.

During our work with the Transmol model, we encountered several challenges while attempting to run the model on our system. The first issue was related to dependency conflicts between two

packages: PyTorch and TorchText. To resolve this, we installed the latest versions of both packages in a conda environment, not the versions specified in the environment.yml file and used a Linux operating system, as there were persistent problems with installing TorchText and other packages on a Windows system.

To fine-tune the given Transmol model and make it output molecules similar to our dataset, we provided the model with all of our molecules in SMILES format. We visually compared the output of the Transmol model using default settings and with our dataset fed into it. The generated molecules were different, with the primary difference being that the Transmol model fed with our dataset generated longer molecules compared to the default Transmol model. However, as the default Transmol model's test dataset contains 176k molecules compared to our 20k, the default Transmol model generates far more molecules for the same number of samples compared to the modified Transmol model. For example, for 1000 samples, the default Transmol model generates a .txt file with a size of 14.6 MB in 8-10 hours, while the modified Transmol model generates a .txt file with a size of 716 KB in just five minutes.

Evaluation

To check the efficacy of our solution, we have looked at the two key common metrics for regression models, namely coefficient of determination (R2) and Mean Absolute Error (MAR). These metrics were calculated using pre-implemented functions from the scikit-learn library in Python.

The results of target metrics, namely absorption max, emission max, quantum yield, lifetime and molecular representation and ML model pairs are given below from Table 3 to Table6.

Prediction Target	Machine Learning Model	Fingerprint Type	R^2	MAE
Absorption Max (nm)	Ensemble Learning	-	0.956	13.462
	KNN Regressor	Morgan	0.936	14.345
	XgBoostRegressor	Morgan	0.938	16.859
	Decision Trees Regressor	Morgan	0.866	22.703
	Graph Neural Network	Avalon (solvent)	0.942	16.647
	MLP	MACCS	0.859	29.054
	Ridge	MACCS	0.644	46.863
	Lasso	MACCS	0.630	47.774
	Random Forest	MACCS	0.897	22.817

Table-3. Model results for absorption max

Prediction Target	Machine Learning Model	Fingerprint Type	R^2	MAE
Emission Max (nm)	Ensemble Learning	-	0.901	20.412
	KNN Regressor	Morgan	0.869	22.001
	XgBoost Regressor	Avalon	0.900	20.547
	Decision Trees Regressor	Morgan	0.866	22.703
	Graph Neural Network	Avalon (solvent)	0.904	20.311
	MLP	MACCS	0.767	33.710
	Ridge	MACCS	0.549	48.920
	Lasso	MACCS	0.550	48.870
	Random Forest	Morgan	0.818	28.000

Table-4. Model results for emission max

Prediction Target	Machine Learning Model	Fingerprint Type	R^2	MAE
Quantum Yield	Ensemble learning	-	0.733	0.112
	KNN Regressor	Morgan	0.683	0.128
	XgBoost Regressor	Avalon	0.713	0.118
	Decision Trees Regressor	Morgan	0.464	0.151
	Graph Neural Network	Avalon	0.689	0.114
	MLP	Morgan	0.605	0.138
	Ridge	MACCS	0.263	0.218
	Lasso	MACCS	0.121	0.248

	Random Forest	Morgan	0.589	0.157

Table-5. Model results for quantum yield

Prediction Target	Machine Learning Model	Fingerprint Type	R^2	MAE
Lifetime (ns)	Ensemble learning	-	0.707	0.333
	KNN Regressor	Morgan	0.713	0.322
	XgBoost Regressor	Avalon	0.750	0.314
	Decision Trees Regressor	Morgan	0.485	0.421
	Graph Neural Network	Avalon (solvent)	0.707	0.333
	MLP	Morgan	0.665	0.385
	Ridge	MACCS	0.375	0.545
	Lasso	Avalon	0.282	0.601
	Random Forest	Morgan	0.651	0.366

Table-6. Model results for lifetime

As can be observed, our approach to ensemble learning provides better results most in predicting Absorption Max and Quantum yield but worse results in predicting Emission max and Lifetime. That might be due to differences in data characteristics or ensemble method configuration. Initially, as it was stated before, we considered ensemble learning techniques to achieve the best results possible. We constructed 2 criterias for model selection to construct our ensemble:

1. The models should belong to different classes of ML algorithms
2. The models should have adequate prediction accuracy

Our criteria helped us to decide on an ensemble consisting of three models: KNN regressor, XGBoost regressor and a Graph Neural Network. The reason is that these models performed the best in almost all predictive cases.

To visualize ensemble learning performance, we constructed graphs that plot the line of best fit against the actual values for each predicted property. They are presented from Figure 4 to Figure 7.

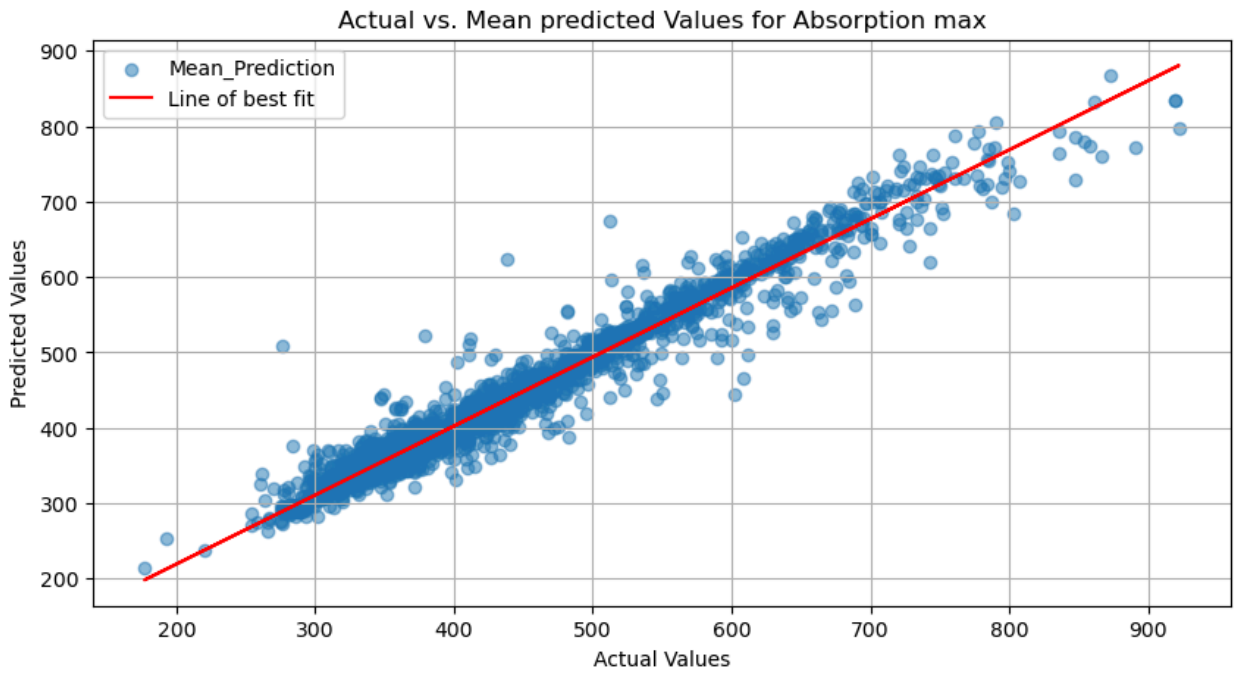


Figure 4. Predicted vs. Actual values of Absorption max by Ensemble Learning. R2=0.956, MAE = 13.462

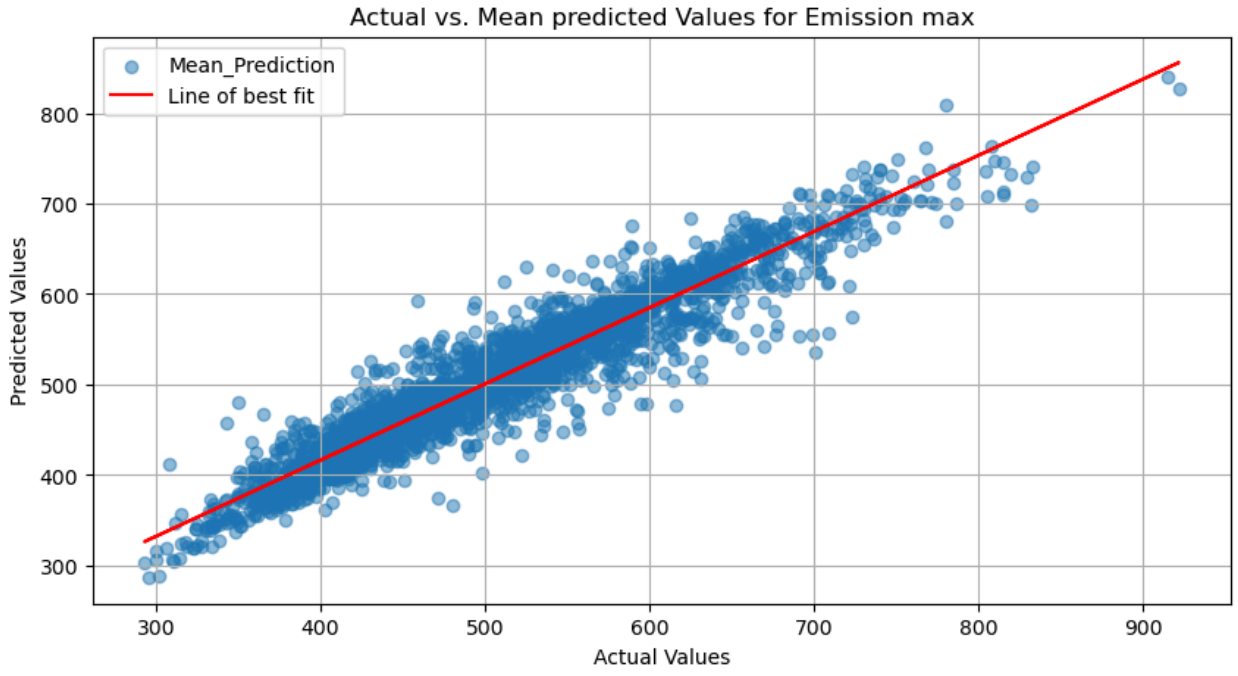


Figure 5. Predicted vs. Actual values of Emission max by Ensemble Learning. R2=0.901, MAE = 20.412

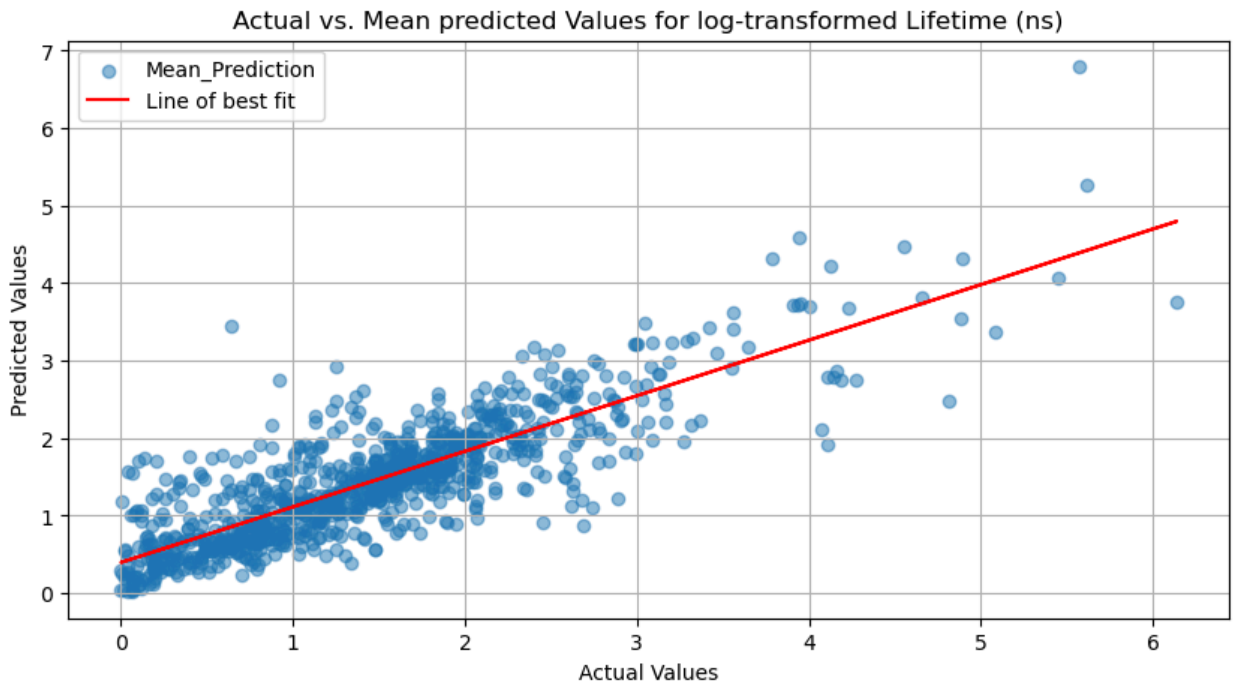


Figure 6. Predicted vs. Actual values of log-transformed lifetime (ns) by Ensemble Learning. R2=0.707, MAE=0.333

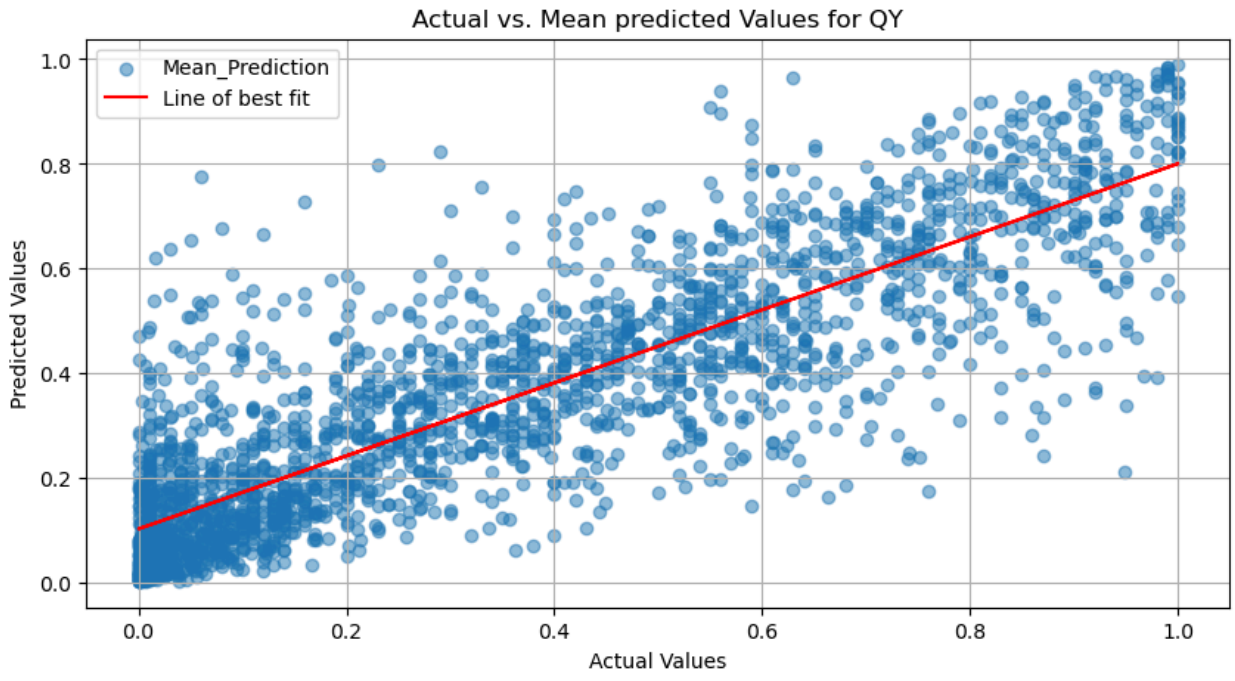


Figure 7. Predicted vs. Actual values of Quantum Yield by Ensemble Learning. R2=0.733, MAE = 0.112

Generative model evaluation

Overall, we generated 8321 molecules using Transmol based on our dataset. To benchmark those generated new molecules we used MOSES benchmark (Polykovskiy et al., 2020). From Table 7, several metrics can be seen, where Valid indicates fraction of valid molecules checked by RDKit's molecular structure parser. Unique@1k and Unique@10k indicate the fraction of unique molecules among first K valid molecules, 1000 and 10000 molecules respectively. Filters are the fraction of molecules that pass filters constrained by the MOSES benchmark which include medicinal chemistry filters (MCFs) and PAINS filters. Those filters are used to discard molecules that are unstable or can become toxic through biotransformations. Novelty describes a fraction of the molecules that are not present in the training set. $IntDiv_1$ and $IntDiv_2$ both assess the chemical diversity of the generated molecules. Synthetic Accessibility Score (SA) is an estimation of how hard it is to synthesize a given molecule, where lower score means better accessibility.

Valid	Unique@ 1k	Unique@ 10k	Filters	Novelty	$IntDiv_1$	$IntDiv_2$	SA
0.438	1.0	1.0	0.791	0.992	0.867	0.855	0.190

Table-7. Benchmark results of Transmol on MOSES benchmark. Higher numbers indicate better results for all metrics except for SA.

Model	Valid	Unique @1k	Unique @10k	Filters	Novelty	$IntDiv_1$	$IntDiv_2$	SA
HMM	0.076	0.623	0.5671	0.9024	0.9994	0.8466	0.8104	0.64
NGram	0.2376	0.974	0.9217	0.9582	0.9694	0.8738	0.8644	0.23
Combinatorial	1.0	0.9983	0.9909	0.9557	0.9878	0.8732	0.8666	0.28
CharRNN	0.975	1.0	0.999	0.994	0.842	0.856	0.85	0.016
VAE	0.977	1.0	0.998	0.997	0.695	0.856	0.85	0.066
AEE	0.937	1.0	0.997	0.996	0.793	0.856	0.85	0.014
JTN-VAE	1.0	1.0	0.9996	0.976	0.9143	0.8551	0.8493	0.16
LatentGAN	0.897	1.0	0.997	0.973	0.949	0.857	0.85	0.085

Table-8. Benchmark results of baseline models on MOSES benchmark taken from Molecular Sets

As it can be seen from both tables, Transmol has a Valid score of 0.438 which is significantly lower than some of the baseline models as more than half of the generated molecules are chemically

invalid. The reason for such a low score is described in a Transmol paper (Zhumagambetov et al., 2021), that states that one of the possible reasons for this is the architecture of the model. Still, Transmol excels in generating unique molecules achieving a perfect score of 1.0 in Unique metrics. In terms of Filters metric, Transmol performs the worst at a score of 0.791, containing the highest proportion of unstable generated molecules.

At Novelty metric, Transmol achieves a high result of 0.992 that shows that Transmol is highly capable of generating molecules not present in the training set. The internal diversity scores of Transmol are competitive compared with other baseline models indicating it is able to generate chemically diverse sets of molecules. And finally, the SA score of 0.190 is a relatively low score that shows that molecules generated by Transmol can be synthesized in an accessible way.

Based on these results, Transmol shows promise in generating novel and unique molecules with high levels of diversity with accessible synthesizing opportunities, however its low validity score indicates area of improvement, as having more than half of newly generated molecules being invalid limits its practical application in real-world setting.

For the CLM, We were unable to assess the performance of the CLM as initially planned. This limitation is due to the challenges associated with the integration of the CLM into our existing *MOSES* benchmark framework.

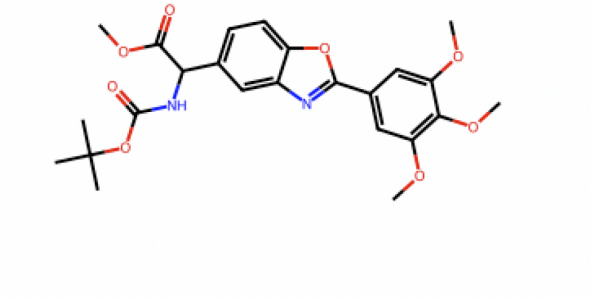
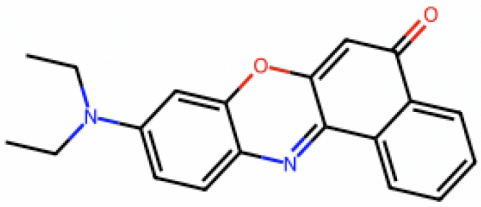
Finally, we have constructed a dataset of newly sampled molecules and predicted properties from best-performing models. The results are shown in Table-9.

Chromophore	Solvent	Absorption max	Emission max	Quantum yield	Lifetime (ns)
COc1ccc(N2CCN(c3ccc4nc(-c5ccc(N(C)C)cc5)oc4c3)CC2)cc1	CICII	458.72644	517.27563	1.000	109.97374
CCN(CC)c1ccc(-c2nc3cc(/C=C/c4ccc(C)cc4)ccc3o2)cc1	CICII	515.5474	587.9956	0.9483274	47.525497
Cc1cc(C)n2c1C=C1c3cc(N(C)C)ccc3C(c3ccc(O)cc3)=[N+]1[B-]2(F)F	CICII	411.62762	513.18585	0.94702464	244.78694
COc1cc(C2=[N+]3C(=Cc4c5ccccc5c(-c5ccccc5)n4[B-]3(F)F)c3nc4ccccc4cc32)ccc1O	CICII	428.78024	608.7465	0.9154229	196.98175
COc1cc(/C=C/c2ccc(N(C)C)cc2)ccc1-c1nc2ccccc2o1	CICII	374.19922	461.71155	0.8876498	94.89495

c1ccc(-c2nc3ccc(N4CCCCC4)cc3o2)cc1	CICII	421.28857	497.51483	0.8632416	92.515274
COc1cc(OC)cc(C2=[N+]3C(=Cc4c5ccccc5c(-c5ccccc5)n4[B-]3(F)F)c3ccc4ccccc4c32)c1	CICII	368.9753	444.6679	0.86099166	149.97249
COc1cc(C2=[N+]3C(=Cc4c5ccccc5c(-c5ccccc5)n4[B-]3(F)F)c3ccc4cc(Br)ccc4c32)ccc1O	CICII	447.32825	594.4784	0.85182315	93.6159
CCN(C)c1ccc2c(c1)C(c1ccc(N(C)C)cc1)=[N+]1C2=Cc2c(C)cc(C)n2[B-]1(F)F	CICII	417.13477	478.60864	0.84852964	206.50757
CCc1ccc(C2=C3c4ccccc4C(c4ccc5ccccc5c4)=[N+]3[B-](F)(F)n3cccc32)cc1	CICII	578.72675	595.71545	0.82670295	146.03694

Table-9. Optical properties of newly sampled molecules

Here is the molecular structure of the newly synthesized molecules using Transmol with highest quantum yield.

	
COC(=O)C(NC(=O)OC(C)(C)C)c1ccc2oc(-c3cc(OC)c(OC)c(OC)c3)nc2c1 Predicted Quantum Yield: 0.815	CCN(CC)c1ccc2nc3c4ccccc4c(=O)cc-3oc2c1 Predicted Quantum Yield: 0.817

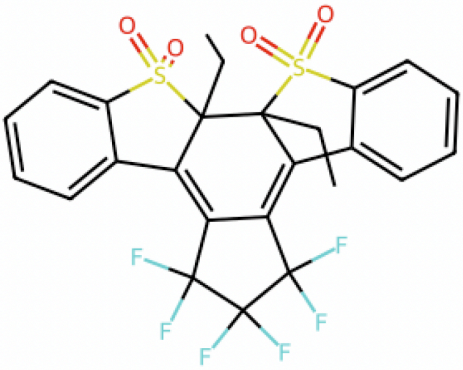
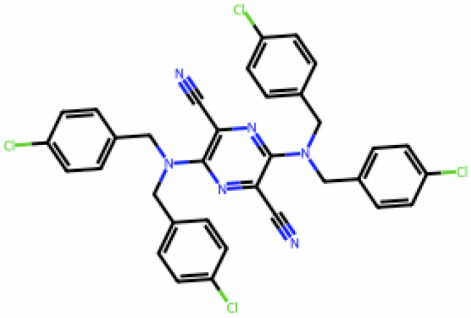
	
CCC12C(=C3C(=C4c5ccccc5S(=O)(=O)C41CC)C(F)(F)C(F)(F)C3(F)F)c1ccccc1S2(=O)=O Predicted Quantum Yield: 0.818	N#Cc1nc(N(Cc2ccc(Cl)cc2)Cc2ccc(Cl)cc2)c(C#N)nc1N(Cc1ccc(Cl)cc1)Cc1ccc(Cl)cc1 Predicted Quantum Yield: 0.822

Table-10. Molecular structure of new synthesized molecule from Transmol and CICII solvent

Conclusion and possible future work

In this project, we have achieved our initial goal of applying Machine Learning techniques to predict optical properties of chromophores. Moreover, we were able to sample new fluorescence molecules using generative models. We started by analyzing the previous work that was done in the industry, and moved on to property prediction using classical Machine Learning algorithms, and then focused on deep learning approaches. Through the use of ensemble learning and the integration of complex molecular data, we significantly improved the predictive accuracy of properties such as absorption and emission maxima. Finally, we were able to apply predictive models to newly sampled molecules. Challenges remained, particularly in predicting properties like quantum yield and lifetime, where data distribution issues and modeling complexities. We were able to achieve this by proper management and working collaboratively with Faculty members.

Future works:

For the future of this project, there are multiple things that can be done:

- Firstly, efforts could focus on improving the predictive performance for both quantum yield and lifetime. These efforts might involve experimenting with additional molecular descriptors, more sophisticated machine learning models, or larger datasets to better capture the nuances of these properties.
- Secondly, re-training the chemical language models. Given the challenges faced with the CLM due to its outdated architecture, re-training the model using a modern machine learning framework that supports GPU acceleration could drastically reduce training times and improve model efficiency.

- Thirdly, to validate the results that were made by our models, DFT calculations could be applied to compute the actual properties of the generated molecules. This would not only provide a benchmark for our current models. Moreover, that would be even more beneficial for newly generated molecules.
- Lastly, usage of advanced techniques for handling missing data, such as imputation methods could further enhance the robustness and accuracy of our predictions.

This project has successfully demonstrated the application of machine learning techniques for the prediction and generation of fluorescence properties of molecules. We are very satisfied with the results obtained during working on this project, and we are extremely grateful for our advisor, Professor, Dr. Siamac Fazli and Dr. Vsevolod Peshkov for their support.

References

Greenman, K. P., Green, W. H., & Gómez-Bombarelli, R. (2022). Multi-fidelity prediction of Molecular Optical Peaks with deep learning. *Chemical Science*, *13*(4), 1152–1162. https://doi.org/10.1039/d1sc05677h

Joung, J. F., Han, M., Jeong, M., & Park, S. (2020). Experimental database of optical properties of organic compounds. *Scientific Data*, *7*(1). https://doi.org/10.1038/s41597-020-00634-8

Elizaveta Vinogradova, Abay Artykbayev, Alisher Amanatay, Mukhamejan Karatayev, Maxim Mametkulov, Albina Li, Anuar Suleimenov, Abylay Salimzhanov, Karina Pats, Rustam Zhumagambetov, Ferdinand Molnár, Vsevolod Peshkov, & Siamac Fazli. (2022). A biologically-inspired multi-modal evaluation of molecular generative machine learning.

Zhumagambetov, R., Molnár, F., Peshkov, V. A., & Fazli, S. (2021). Transmol: Repurposing a language model for molecular generation. *RSC Advances*, *11*(42), 25921–25932. https://doi.org/10.1039/d1ra03086h

Moret, M., Friedrich, L., Grisoni, F., Merk, D., & Schneider, G. (2020). Generative molecular design in low data regimes. *Nature Machine Intelligence*, *2*(3), 171–180. https://doi.org/10.1038/s42256-020-0160-y

Ye, Z.-R., Huang, I.-S., Chan, Y.-T., Li, Z.-J., Liao, C.-C., Tsai, H.-R., Hsieh, M.-C., Chang, C.-C., & Tsai, M.-K. (2020). Predicting the emission wavelength of organic molecules using a combinatorial QSAR and machine learning approach. *RSC Advances*, *10*(40), 23834–23841. https://doi.org/10.1039/d0ra05014h

Ju, C.-W., Bai, H., Li, B., & Liu, R. (2021). Machine learning enables highly accurate predictions of photophysical properties of organic fluorescent materials: Emission wavelengths and quantum yields. *Journal of Chemical Information and Modeling*, *61*(3), 1053–1065. https://doi.org/10.1021/acs.jcim.0c01203

Shuai Wang, ChiYung Yam, Shuguang Chen, Lihong Hu, Liping Li, Faan-Fung Hung, Jiaqi Fan, Chi-Ming Che, & GuanHua Chen. (2023). Predictions of photophysical properties of phosphorescent platinum(II) complexes based on ensemble machine learning approach.

Daniil Polykovskiy, Alexander Zhebrak, Benjamin Sanchez-Lengeling, Sergey Golovanov, Oktai Tatanov, Stanislav Belyaev, Rauf Kurbanov, Aleksey Artamonov, Vladimir Aladinskiy, Mark Veselov, Artur Kadurin, Simon Johansson, Hongming Chen, Sergey Nikolenko, Alan Aspuru-Guzik, & Alex Zhavoronkov. (2020). Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models.

Öztürk, H., Ozkirimli, E., & Özgür, A. (2018). A novel methodology on distributed representations of proteins using their interacting ligands. *Bioinformatics*, *34*(13), i295–i303. https://doi.org/10.1093/bioinformatics/bty287

Li, H., Zhao, D., & Zeng, J. (2022). KPGT: Knowledge-Guided Pre-training of Graph Transformer for Molecular Property Prediction. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. ACM.

Wang, S., Guo, Y., Wang, Y., Sun, H., & Huang, J. (2019). Smiles-bert. Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics. https://doi.org/10.1145/3307339.3342186

Seyone Chithrananda, Gabriel Grand, & Bharath Ramsundar. (2020). ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction.

Jaeger, S., Fulle, S., & Turk, S. (2018). Mol2vec: Unsupervised machine learning approach with chemical intuition. Journal of Chemical Information and Modeling, 58(1), 27–35. https://doi.org/10.1021/acs.jcim.7b00616

Durant, J. L., Leland, B. A., Henry, D. R., & Nourse, J. G. (2002). Reoptimization of MDL keys for use in Drug Discovery. Journal of Chemical Information and Computer Sciences, 42(6), 1273–1280. https://doi.org/10.1021/ci010132r

Xu, Z., Wang, S., Zhu, F., & Huang, J. (2017). Seq2seq fingerprint. Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology,and Health Informatics. https://doi.org/10.1145/3107411.3107424

David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, & Ryan P. Adams. (2015). Convolutional Networks on Graphs for Learning Molecular Fingerprints.

O'Boyle, N. M., Campbell, C. M., & Hutchison, G. R. (2011). Computational design and selection of Optimal Organic Photovoltaic Materials. The Journal of Physical Chemistry C, 115(32), 16200–16210. https://doi.org/10.1021/jp202765c

Kuhn, S., & Johnson, S. R. (2019). Stereo-aware extension of HOSE codes. ACS Omega, 4(4), 7323–7329. https://doi.org/10.1021/acsomega.9b00488

Probst, D., & Reymond, J.-L. (2018). A probabilistic molecular fingerprint for Big Data Settings. Journal of Cheminformatics, 10(1). https://doi.org/10.1186/s13321-018-0321-8

Rogers, D., & Hahn, M. (2010). Extended-connectivity fingerprints. Journal of Chemical Information and Modeling, 50(5), 742–754. https://doi.org/10.1021/ci100050t

Gedeck, P., Rohde, B., & Bartels, C. (2006). QSAR − how good is it in practice? comparison of descriptor sets on an unbiased cross section of corporate data sets. Journal of Chemical Information and Modeling, 46(5), 1924–1936. https://doi.org/10.1021/ci050413p

Morgan, H. L. (1965). The generation of a unique machine description for chemical structures-a technique developed at Chemical Abstracts Service. Journal of Chemical Documentation, 5(2), 107–113. https://doi.org/10.1021/c160017a018

David, L., Thakkar, A., Mercado, R., & Engkvist, O. (2020). Molecular representations in AI-Driven Drug Discovery: A review and practical guide. Journal of Cheminformatics, 12(1). https://doi.org/10.1186/s13321-020-00460-5

Hansen, K., Biegler, F., Ramakrishnan, R., Pronobis, W., von Lilienfeld, O. A., Müller, K.-R., & Tkatchenko, A. (2015). Machine learning predictions of molecular properties: Accurate many-body potentials and nonlocality in chemical space. The Journal of Physical Chemistry Letters, 6(12), 2326–2331. https://doi.org/10.1021/acs.jpclett.5b00831

Schutt, K., Glawe, H., Brockherde, F., Sanna, A., Müller, K., & Gross, E. (2014). How to represent crystal structures for machine learning: Towards fast prediction of electronic properties. Phys. Rev. B, 89, 205118.

Heller, S. R., McNaught, A., Pletnev, I., Stein, S., & Tchekhovskoi, D. (2015). InChI, the IUPAC International Chemical Identifier. Journal of Cheminformatics, 7(1). https://doi.org/10.1186/s13321-015-0068-4

O'Boyle, N., & Dalke, A. (2018). Deepsmiles: An Adaptation of Smiles for Use in Machine-Learning of Chemical Structures. https://doi.org/10.26434/chemrxiv.7097960.v1

Weininger, D. (1988). SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. Journal of Chemical Information and Modeling, 28.