

Southern Methodist University

**SMU Scholar**

---

Computer Science and Engineering Theses and  
Dissertations

Computer Science and Engineering

---

Summer 7-29-2021

## Collaborative Filtering based Generative Networks

Raghuram Srinivas

*Southern Methodist University*, [rsrinivas@smu.edu](mailto:rsrinivas@smu.edu)

Follow this and additional works at: [https://scholar.smu.edu/engineering\\_compsci\\_etds](https://scholar.smu.edu/engineering_compsci_etds)



Part of the [Technology and Innovation Commons](#)

---

### Recommended Citation

Srinivas, Raghuram, "Collaborative Filtering based Generative Networks" (2021). *Computer Science and Engineering Theses and Dissertations*. 19.

[https://scholar.smu.edu/engineering\\_compsci\\_etds/19](https://scholar.smu.edu/engineering_compsci_etds/19)

This Dissertation is brought to you for free and open access by the Computer Science and Engineering at SMU Scholar. It has been accepted for inclusion in Computer Science and Engineering Theses and Dissertations by an authorized administrator of SMU Scholar. For more information, please visit <http://digitalrepository.smu.edu>.

CFGNNETS : COLLABORATIVE FILTERING BASED GENERATIVE  
NETWORKS

Approved by:

---

Dr.Eric C Larson  
Department of Computer Science  
Dissertation Committee Chairperson

---

Dr. Jia Zhang  
Department of Computer Science

---

Dr.King Ip Lin  
Department of Computer Science

---

Dr.Corey Clark  
Department of Computer Science

---

Dr.Elfi Kraka  
Department of Chemistry  
Dedman College of Humanities & Sciences, SMU

CFGNNETS : COLLABORATIVE FILTERING BASED GENERATIVE  
NETWORKS

A Dissertation Presented to the Graduate Faculty of the  
Lyle School of Engineering  
Southern Methodist University  
in  
Partial Fulfillment of the Requirements  
for the degree of  
Doctor of Philosophy  
with a  
Major in Computer Science  
by

Raghuram Srinivas

M.S., Data Science, Southern Methodist University  
B.E., Computer Science, PES Institute of Technology

August 4, 2021

Copyright (2021)  
Raghuram Srinivas  
All Rights Reserved



Srinivas, Raghuram

M.S., Data Science, Southern Methodist University  
B.E., Computer Science, PES Institute of Technology

CFGenNets : Collaborative Filtering based Generative Networks

Advisor: Dr.Eric C Larson

Doctor of Philosophy conferred August 4, 2021

Dissertation completed August 4th, 2021

Collaborative Filtering, a popular method for recommendation engines, models its predictions using past interactions between the entities in question (aka users/movies or customers/products etc). The method does not rely on the explicit properties of the entities, the identification of which may be intractable. In this work, we leverage this advantage rendered by Collaborative Filtering where the explicit features need not be defined apriori by evaluating its application to the domain of Ligand based Virtual Screening. We further attempt to address the drawback of Collaborative Filtering , ie the lack of interpret ability of the factors discovered through collaborative filtering by creating a novel class of generative deep learning models ,called Collaborative Filtering based Generative Networks (CFGenNets). We show the utility of CFGenNets in 2 domains 1) Ligand based Virtual Screening and 2) Image generation from keyword tags/meta descriptors.

## TABLE OF CONTENTS

LIST OF FIGURES . . . . .	ix
LIST OF TABLES . . . . .	xvi
CHAPTER	
1. Introduction to CFGenNet . . . . .	1
1.1. Motivation . . . . .	1
1.1.1. Ligand-based Virtual Screening Motivation . . . . .	2
1.2. A primer on molecules . . . . .	3
1.3. Ligand based Virtual Screening . . . . .	4
1.3.1. Ligand Fingerprints . . . . .	6
1.3.2. Fingerprints and Virtual Screening . . . . .	7
1.3.3. Computer Aided Drug Design . . . . .	8
1.4. Dissertation Objectives . . . . .	9
1.4.1. Aim 1: Implicit fingerprint discovery by means of collaborative filtering, for application to virtual screening . . . . .	9
1.4.2. Aim 2: Generative deep learning based drug design using Implicit Ligand fingerprints . . . . .	10
1.4.3. Aim 3: Cross domain applicability of Implicit Fingerprints for Image Generation . . . . .	11
1.5. Dissertation Contributions . . . . .	11
1.5.1. Contributions from Research Aim 1 . . . . .	11
1.5.2. Contributions from Research Aim 2 . . . . .	12
1.5.3. Contributions from Research Aim 3 . . . . .	13
1.6. Definitions . . . . .	14
2. Collaborative Filtering . . . . .	15

2.1. Overview of Collaborative Filtering . . . . .	15
2.1.1. Neighborhood-based Collaborative Filtering . . . . .	15
2.1.2. Matrix Factorization Method . . . . .	16
2.1.3. Conclusion . . . . .	18
3. Ligand based Virtual Screening using Collaborative Filtering . . . . .	20
3.1. Data Preparation . . . . .	20
3.1.1. Evaluation Criteria . . . . .	21
3.1.2. Variance and Bias in Validation Set Selection . . . . .	24
3.2. Collaborative Filtering Model Selection . . . . .	26
3.2.1. Conclusion . . . . .	28
4. Performance Analysis of Collaborative Filtering for ligand based virtual screening	29
4.1. Overall Performance of Collaborative Filtering . . . . .	29
4.2. Performance Comparisons by Number of known Actives . . . . .	31
4.2.1. Conclusion . . . . .	33
5. Introducing Implicit Target and Ligand Fingerprints . . . . .	34
5.1. Overview . . . . .	34
5.2. Limitations . . . . .	37
5.3. Conclusion of Research Objective 1 . . . . .	38
6. Introduction to CFGenNet, Collaborative Filtering based Generative Networks .	39
6.1. Decoders to generate SMILES code . . . . .	40
6.1.1. Related Works . . . . .	40
6.2. Neural Network Architecture . . . . .	43
6.3. Dataset Description . . . . .	45
6.4. Construction of the Neural Network . . . . .	47
6.4.1. Gated Recurrent Neural Networks . . . . .	47

7. Syntactic & Novelty Analysis of Generated Ligands . . . . .	54
7.1. Introduction . . . . .	54
7.2. Syntactic Analysis of Generated Ligands . . . . .	54
7.3. Novelty analysis of Generated Ligands . . . . .	55
7.3.1. Conclusion . . . . .	58
8. Semantic Analysis of Generated Ligands . . . . .	60
8.1. Physical Properties of Generated and Anchor Ligands . . . . .	60
8.2. Binding Affinity predictions of the potentially novel ligands . . . . .	65
8.3. Comparison with FDA Approved Drugs . . . . .	69
9. CFGenNets for Image Generation . . . . .	73
9.1. Motivation for Image Synthesis . . . . .	74
9.1.1. A Primer on Related Work in Image Synthesis . . . . .	74
9.2. Methodology . . . . .	78
9.2.1. Dataset details . . . . .	78
9.2.2. Collaborative Filtering on CelebA dataset . . . . .	79
9.3. Analysis of Implicit Fingerprint . . . . .	80
9.3.1. Implicit Feature Fingerprints . . . . .	82
9.3.2. Conclusion . . . . .	84
10. CFGenNets for Image Generation : Architecture . . . . .	85
10.1. Decoders to generate images . . . . .	85
10.2. Decoder Architecture . . . . .	85
10.2.1. Convolution Layer Design . . . . .	88
10.3. Results . . . . .	89
10.3.1. Limitations and Next Steps . . . . .	92
11. CFGenNets and Variational Autoencoders . . . . .	94

11.1. CFGenNets and Variational Autoencoders . . . . .	94
11.1.1. Visualization of latent vectors . . . . .	98
11.1.2. Conclusion . . . . .	100
12. Image Attribute Manipulations using CFGenNets . . . . .	103
12.1. Attribute Manipulation using Composite CFGenNet Decoders . . . . .	103
12.2. Selective Attributes Manipulation using CFGenNets . . . . .	106
12.3. Conclusion & Future work . . . . .	111
APPENDIX A : Sample Images generated from CFGenNets . . . . .	114
APPENDIX B : Image Manipulation via Composite CFGenNet Decoders . . . . .	115
APPENDIX C : Image Manipulation via Feature Gated Composite CFGenNet Decoders	117
BIBLIOGRAPHY . . . . .	119

## LIST OF FIGURES

Figure		Page
1.1.	A sample molecule : Illustration of Caffeine molecule with its constituent atoms and bonds. . . . .	4
1.2.	Ligand Based Virtual Screening Methods :Several classes of algorithms exist for performing ligand based virtual screening. As illustrated, the classes algorithms can be categorized by the usage of known active ligands v/s both known active and inactive ligands. The categorization of our method, utilizing collaborative filtering is also illustrated in the figure. . . . .	5
2.1.	Latent Factor Embedding : Sub Figure a illustrates the concept of latent factor the latent factor recommendation for a movie recommendation engine. The latent factor method relies on learning hidden factors from the user-movie ratings alone. In this simplified example the system learns two dimensions from the ratings and places the movies and users in this 2D space. The users predicted rating of the movie would be a dot product of the user's and movie's location in the 2D space. Figure b illustrates the same concept for a target-ligand embedding. Here the latent factors correspond to properties that the ligands and protein targets can be jointly modeled with. The properties may correspond to a distinct chemical property, but might also pertain to a factor not well described by traditional cheminformatics. . .	17
2.2.	Illustration of the matrix factorization method :In this example the highlighted cells represent the known affinities. The SVD method generates the matrices $P$ and $Q$ . Optimization methods are employed to minimize the error between the known affinities and their predicted values. . . . .	19
3.1.	Number of targets by known assay counts : Illustration of the distribution of the number of targets by the known assay counts for each target. More than half of the targets have less than about 200 recorded assays. Therefore, it is imperative that the virtual screening methods perform well when the number of assays is relatively sparse. . . . .	22

3.2.	AVE Bias : The image to the left visualizes the comparison of AVE Bias across 4 sets of training and validation data. The figure shows that the 4 training and validation sets are randomly split with similar bias measures across the sets, thereby minimizing the impact of variance across our study. The image to the right shows the spread of the AVE Bias for each target in the dataset. The distribution of the AVE Bias from -0.5 to 1.5 (where closer to 0 indicates no bias) across targets facilitates the study of the resilience of the collaborative filtering algorithm to the impact of such a bias. . . . .	24
3.3.	Hyperparameter tuning for matrix factorization collaborative filtering : Illustration of the effects of the number of factors, $f$ , and regularization parameter, $\lambda$ , on model performance. While model performance improves with more factors, it may lead to over fitting. The regularization parameter does not have a consistent performance, revealing that performance is not sensitive to $\lambda$ . The error bars represent the standard error of the mean across the BEDROC <sub>20</sub> scores for each Target at the specified parameter value . . . . .	28
4.1.	Comparison of AUC, BEDROC <sub>20</sub> , and EF <sub>1%</sub> scores across all algorithms : The collaborative filtering based implicit structure method, based only on assay outcomes performs on par with the other methods across the three evaluation criteria. The across target Random Forest model is the next best performer. All other models perform about the same and poorer than collaborative filtering. . . . .	29
4.2.	Comparison of AUC, BEDROC <sub>20</sub> , and EF <sub>1%</sub> scores by number of known activities : Illustration of the comparative performance of the algorithms by the number of known affinities per target in the training data. Error bars correspond to standard error. The collaborative filtering based implicit structure methods significantly outperform other algorithms when the training data has 100 or fewer activities. Beyond 100 activities the performance of all algorithms converge. . . . .	30
4.3.	Distribution of AUC, BEDROC <sub>20</sub> , and EF <sub>1%</sub> : Distribution of AUC, BEDROC <sub>20</sub> , and EF <sub>1%</sub> scores across all algorithms when the number of known affinities per target is less than or equal to 100 activities. Based on a two-tailed t-test of the scores ( $p < 0.01$ ), the difference in scores for collaborative filtering with other methods is statistically significant. . . . .	31

5.1.	Implicit Target Fingerprints : t-SNE plot reducing the dimensions of the 50-dimensional implicit fingerprints into two dimensions for a subset of targets in the ChEMBL database, highlighting known cancer and thyroid related protein targets. The method successfully clusters related targets close to each other, as indicated in the zoomed version of the largest cluster of cancer targets. Interestingly the clusters also contain other targets which are found to have expressions with known cancer/thyroid targets as in the example of Vascular Endothelial Growth Factor Receptor-2 Expression in breast cancer 1 protein. . . . .	35
5.2.	t-SNE Plots of Implicit Ligand Fingerprints : Plots for three cancer targets are shown where each point represents a compound assayed from the ChEMBL database. The concentration results of the assays are color-coded. t-SNE plots of the 50-dimensional implicit representations, reduced to 2 dimensions preserving distance. . . . .	36
6.1.	Deep learning based Ligand Design using Implicit Fingerprints from Collaborative Filtering - Architecture . . . . .	45
6.2.	Data Distribution : The first figure illustrates the number of molecules against number of assays, binned at specified values or ranges. Close to 50% of the molecules have only 2 prior assays. The second figure illustrates the number of ligands against the number of known assays with positive affinities. It can be observed that 62% of the ligands with only one assay with positive binding affinity. . . . .	46
6.3.	Implicit Fingerprints to SMILES Decoder : The deeplearning network learns ligand representations by employing data augmentation technique at the input layer. The continuous representation obtained is then fed into a series of dense layers followed by a Gated Recurrent Unit Neural Network to obtain the corresponding SMILES string. . . . .	48
6.4.	Train and Validation Losses of the Neural Network : Training and validation losses across multiple runs of the neural network. . . . .	51
6.5.	Neural network architecture of the deep learning model. . . . .	52
6.6.	Layers and parameters trained in the deep learning model. . . . .	53



- 7.1. Novelty of generated ligands : Distribution of the number potentially novel ligand and their closest hit pairs v/s differences in the functional groups & TC similarity ranges. The figure demonstrates that A degree of novelty could be associated with the generated ligands when compared with the 1.3Billion known ligands from ZINC DB. The differences in the number of functional groups between the anchor and generated ligands range from 0 to 9,with at least 67% ligands with 2 or more differing functional groups. The TC similarity bins help gauge the distribution of the TC similarities between the pairs. It is seen that the lower the similarity,more likely it is for the pair having varying functional groups. . . . . 57
- 7.2. Ligand generation v/s total assays : Correlation of the ability to generate potentially novel ligands with prior assays: Box plots show the co-relation between the two sets of anchor ligands - one set from 1 or more potentially novel ligands were generated and the second set which yielded no ligands when sampled in the implicit fingerprint latent space. The first figure visualizes the total number of known assays that exists for each set. The second box-plot visualizes the total number of positive binding affinities already recorded for each assay. . . . . 59
- 8.1. Property Distribution between anchor ligands and generated ligands : (A) Quantitative Estimate of DrugLikeness(QED) (B) Partition Coefficient (LogP) (C) Synthetic Accessibility Score (SAS) (D) Number of Benzene rings. The figure demonstrates that the property distributions of the anchor ligands is similar to the potentially novel ligands generated from the corresponding anchors across all 4 properties. . . . . 62
- 8.2. Lipinski's Rule of 5 : Lipinski's Rule of 5 valuated on the potentially novel ligands generated from implicit fingerprints. The figure demonstrates that 80% out of the 1,831 potentially novel ligands satisfy 3 or more rules, signifying that the generated ligands have properties to be an effective drug. 64
- 8.3. Pairwise binding affinity scores : The plot illustrates the similarities in the bio-activity between each pair of anchor ligands and their corresponding generated ligands to the 102 DUDE protein targets. The large regions of dark blue hue in the heat map demonstrate a strong co-relation between the binding affinities for most pairs with the DUDE targets. The scatter plot illustrates two sample pairs, with the top right plot representing a pair with very similar affinity scores, with the bottom right plot illustrating a pair where the affinities differ between the anchor and generated ligand. . 66

8.4.	IoU scores and the Tanimoto coefficient scores : A) Comparison of the IoU scores and the Tanimoto coefficient scores between the anchor and generated ligands. : The figure illustrates that there is no strong correlation between the anchor and generated ligands. B)Histogram of TC scores across all the pairs : Illustrates the distribution of the similarity scores between the pairs.	67
8.5.	Scaffold analysis : A pseudohilbert curve is plotted for anchor ligands and generated ligands. The color denotes SSnet scores. Similarity between anchor and generated pseudohilbert curves and the low difference among them, signifies that our method retains scaffolds from the anchor ligands while also predicting similar bioactivities. . . . .	68
8.6.	Novel ligands generated around known cancer drug, DATASINIB : It is observed that the generated ligands have ring structures similar to the anchor drug compound. In order to further validate if the rings are indeed realistic and ligands pragmatic,we investigated synthesizability, and like-lieness of the ring to bind a protein. Section 0.2.1 from Supporting Information [1] enumerates information about the novelty and the binding affinities exhibited by these generated ligands. . . . .	70
8.7.	SMINA scores for generated ligands around DATASINIB : Sample test on various active/inactive targets for anchor ligands. The first 5 targets 2FO0, 1PKG, 1AVZ, 1GQ5 and 1MQB are active and the rest inactive. The red color denotes the anchor ligands and the black denotes generated ligands respectively. . . . .	71
8.8.	Comparison with Variational Autoencoder based fingerprints : Comparison of Implicit and latent fingerprints on FDA approved drugs and their corresponding targets. The red color denotes the anchor ligand and the black color denotes generated ligands respectively. The latent label and implicit label shows the binding affinities for generated ligands from the method developed by [2] (in blue) and our method (in green). . . . .	72
9.1.	Sample Images from CelebA dataset. Reproduced from the original article [3] here for easy reference. . . . .	79
9.2.	CelebA dataset recast as a matrix with each row representing one image from the dataset and the columns representing the attributes described for the image. . . . .	80
9.3.	Representation of the implicit image and feature fingerprints in a 2 dimensional space using t-SNE algorithm. The images and the features both share the same latent space, with the features interspersed amidst the images. . . .	81

9.4. Performance Metrics of facial attribute prediction classifier trained on the implicit image fingerprints. . . . .	82
9.5. The hash map illustrates the abilities of the Implicit Feature Fingerprints to predict the presence of respective facial descriptor attributes in the images. The Implicit feature fingerprints generate accurate predictions even though they are fed into the classifier trained solely on the Implicit Image Fingerprints. This is possible because the features and images share the same latent fingerprint space when generated via Collaborative Filtering. . . . .	83
10.1. CFGenNets architecture for generating images from implicit image fingerprints generated from collaborative filtering. . . . .	86
10.2. CFGenNets for Images : Network design using transpose convolution operations and bi-linear upsampling. . . . .	87
10.3. Performance metrics across the 2 networks trained on celebA dataset. After 100 epochs, the network encompassing transpose convolution layers has a lower mean square error and higher reconstruction accuracy. . . . .	90
10.4. A small sample of generated images in comparison with the original images. Appendix A12.3 contains a larger sample of images generated from CFGenNets. . . . .	91
10.5. CFGenNets for Image generation : Performance metrics at the end of 100 epochs on the final model trained on 160k images and evaluated on 40k images. . . . .	92
11.1. Composite CFGenNet decoders Architecture: These decoders consume the latent fingerprints from Variational Autoencoders and Collaborative Filtering as inputs. These are eventually averaged in the network to obtain the corresponding image. . . . .	96
11.2. VAEs v/s vanilla CFGenNets v/s Composite CFGenNets: a random sample of images processed via 3 different types of networks. The reconstruction accuracy is highest for the Composite CFGenNet Decoders. They take advantage of the information encoded within the latent spaces generated by Variational Autoencoders and Collaborative Filtering . . . . .	97
11.3. VAEs v/s vanilla CFGenNets v/s Composite CFGenNet Decoders : a random sample of images processed via 3 different types of networks. Visually, it is evident that the outcomes from the Composite CFGenNet Decoders are more aligned to the original image then the other 2. . . . .	98

11.4. Visualization of 20 x 20 face images based on VAE latent vectors by t-SNE algorithm . . . . .	101
11.5. Visualization of 20 x 20 face images based on Implicit Image Fingerprints by t-SNE algorithm . . . . .	102
12.1. Image manipulation via Composite CFGenNet decoders : Images are generated by feeding the latent encoding obtained by VAEs and Implicit Feature fingerprint obtained from Collaborative Filtering. The first set of images are fused with the feature vector representing the attribute 'Wearing Lipstick' and the second set of images with the vector representing the attribute 'Smiling'. . . . .	105
12.2. Feature Gated Composite CFGenNets : Architecture. . . . .	108
12.3. Feature Gated Composite CFGenNets: sample images with 1 dimensional feature manipulations . . . . .	109
12.4. Feature Gated Composite CFGenNets : sample images with 2 dimensional feature manipulations . . . . .	110
12.5. Sample images generated from CFGenNets with Transpose Convolution layers and Bi-linear Upsampling layers . . . . .	114
12.6. Random Image 1 manipulated using Implicit Feature fingerprints via Composite CFGenNet decoder . . . . .	115
12.7. Random Image 2 manipulated using Implicit Feature fingerprints via Composite CFGenNet decoder . . . . .	116
12.8. Random Image 1 manipulated using Implicit Feature fingerprints via Feature Gated Composite CFGenNet decoder . . . . .	117
12.9. Random Image 2 manipulated using Implicit Feature fingerprints via Feature Gated Composite CFGenNet decoder . . . . .	118

## LIST OF TABLES

Table	Page
1.1. Definitions . . . . .	14
3.1. Collaborative Filtering hyper-parameter tuning : Tabulated top 5 results from training multiple collaborative filtering models using neighborhood methods and matrix factorization methods. . . . .	27
7.1. Summary of Generated Ligands . . . . .	58
8.1. Properties of anchor and potentially novel ligands. . . . .	61

# Chapter 1

## Introduction to CFGenNet

### 1.1 Motivation

Collaborative filtering algorithms have been historically used in the context of designing recommendation systems such as movie recommendation engines, as well as up-sell and cross-sell recommendation engines for e-commerce sites [4–7]. In general, collaborative filtering is a method for making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from other users (collaborating) [8]. This approach relies on modeling predictions using past interactions between the users and the items rated. The method aims to infer the missing ratings and impute them into a sparse “rating” matrix. This is in contrast to traditional machine learning that models individual users or items based on their attributes. For example in the context of a movie recommendation application, a movie could be described by its genre, reviews, starring actors, and awards, and a user could be described by her/his demographic information, any past reviews, genre preferences, friends reviews, and so on. It quickly becomes evident that identifying the entire gamut of properties that accurately represent the users and movies is a daunting, if not an intractable, task. Collaborative filtering, then, is an alternate technique which relies on past transactions without relying on explicit attributes of the user or movie—which has both positive and negative consequences. Positively, the features of items and users need not be defined apriori. Negatively, the factors discovered through collaborative filtering are not readily interpretable. In this work, we attempt to mitigate this lack of interpretability through creating generative models. We show utility in two application domains: (1) ligand based virtual screening and (2) image generation from key word tags . Because our ligand based virtual screening research is more mature, we motivate and explain its motivation more fully.

### 1.1.1 Ligand-based Virtual Screening Motivation

We extend the concept of collaborative filtering to the domain of ligand based virtual screening. Ligand based Virtual screening is an automated computational method of filtering candidate ligands (e.g., drugs) based upon their inferred relationship with a given target (e.g., a cancer protein). Typically the aim is to assess binding affinity for a ligand-target pair. That is, the objective is to understand if the ligand and protein will interact with one another. In order to test the applicability of collaborative filtering algorithms to this domain, we liken the targets to users and ligands to movies. The “rating” between targets and ligands can be represented by the known binding affinities (active or inactive). These binding affinities are collected by scientists around the world that assay the given target and ligands and provide the result of the assay to a network of databases.

The original movie recommendation algorithm proposed by Simon Funk [9] factorized the user-movie rating matrix as the product of two lower dimensional matrices, the first one has a row for each user, while the second has a column for each movie. The row or column associated to a specific user or movie is referred to as a latent factor. In the context of the movie recommendation engine, the latent factors of the users and the movies intuitively encode key features of the users, movies, and the context (when the interaction occurs), resulting in the recommendations. However, the collaborative filtering algorithms do not provide a mechanism to establish the set of such features that are encoded in the shared latent spaces of the items and users. That is, the factors are mathematical constructs that might also have a semantically meaningful interpretations but those meanings are not immediately obvious.

Therefore, we investigate the possibility of using the latent factors obtained from collaborative filtering to map back to the corresponding known molecular structures. In particular, we find that deep learning algorithms are highly suitable for this process. The ability to navigate from desired binding affinity between ligands and targets to the explicit structure of the ligand can be a potent utility in the field of de-novo drug design. To this extent, we introduce a novel class of generative algorithms called **CFGenNet**, Collaborative Filtering based Generative Networks. In the context of de-novo drug design, CFGenNet aims to generate novel ligands from the latent factors obtained from collaborative filtering. To

the best of our knowledge, this is the first use of generative algorithms to extend the utility of discovered latent space from collaborative filtering. In order to extend the conclusions of this concept to other application domains beyond virtual screening, we further wish to investigate the cross domain applicability of CFGenNets by applying the said methods to a completely different data set. For this purpose, we intend to evaluate the abilities of CFGenNets to generate images from their corresponding implicit image fingerprints obtained from collaborative filtering.

In the remainder of this chapter, we provide an introduction to certain relevant topics of cheminformatics to help orient the readers to the broader context before we begin describing our research. Subsequently, we present an overview of our research objectives followed by our contributions to the field. Subsequent chapters contain in-depth discussions into the data, methods, and results supporting this thesis.

## 1.2 A primer on molecules

Molecules form the basic unit of any chemical compound or ligand that take part in a chemical reaction. A molecule is a group of two or more atoms that form the smallest identifiable unit into which a pure substance can be divided and still retain the composition and chemical properties of that substance [10]. Figure 1.1 visually illustrates the caffeine molecule.

As is evident from Figure 1.1, molecules have two or more atoms held together by chemical bonds. These bonds that hold atoms together themselves can be of two kinds: covalent or ionic. A covalent bond is a chemical bond that involves the sharing of electron pairs between atoms. The ionic on the other hand is formed by the complete transfer of some electrons from one atom to another resulting in an electrostatic attraction between the two ions of opposite charge. The molecules are dynamic entities; The atoms that make up the molecules are in a constant state of motion relative to one another thereby causing their relative positions and lengths of the bonds to be changing frequently. This consequently yields multiple “molecule conformations,” which refer to any one of the infinite number of possible spatial arrangements of atoms in a molecule. To make things more complex, the molecules can also exhibit a property of chirality, implying that a mirror image of a molecule is not the same as itself.



## The caffeine molecule

chemical name: 1, 3, 7-trimethylxanthine  
chemical formula:  $C_8H_{10}N_4O_2$

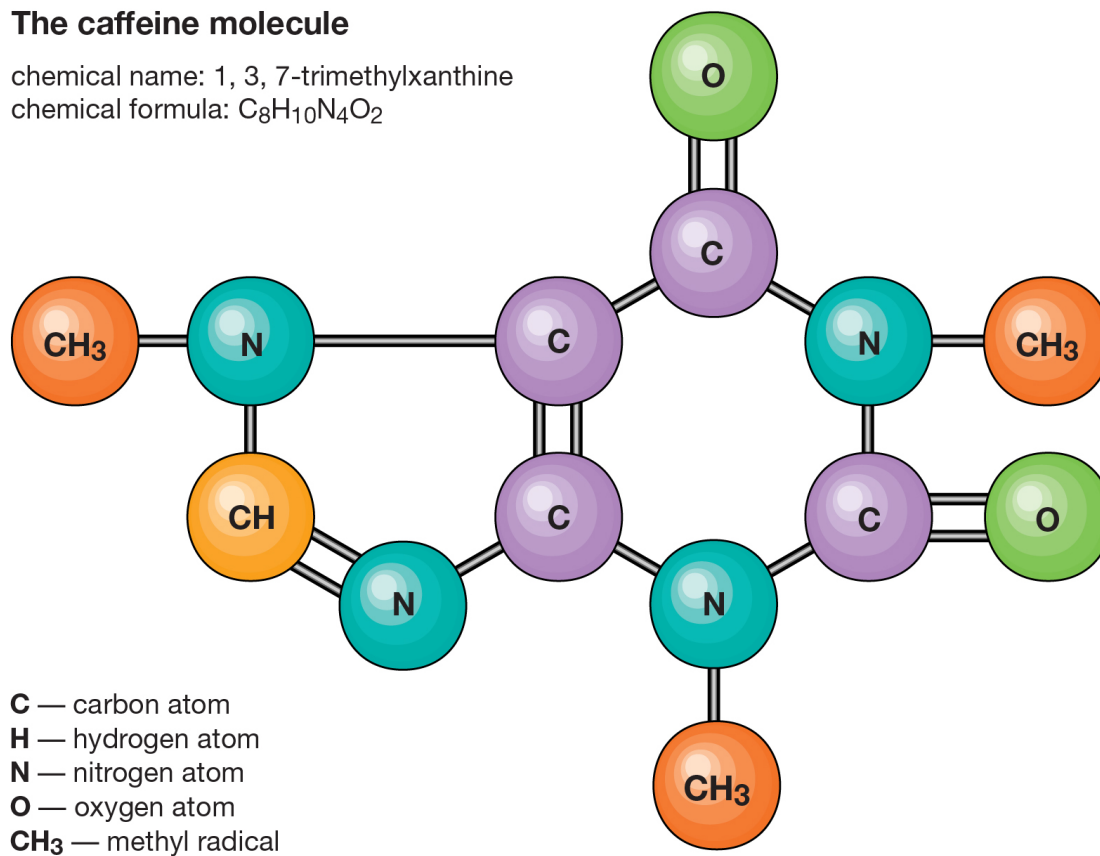


Figure 1.1: A sample molecule : Illustration of Caffeine molecule with its constituent atoms and bonds.

That is, the molecules can come in two different forms that are intuitively mirror images of each other (like our hands, a right hand and a left hand). While the physical forms are similar, they may exhibit different behaviour in practice. The biological and pharmaceutical activity of many molecules is often directly related to their chirality.

### 1.3 Ligand based Virtual Screening

Ligand based Virtual Screening refers to computational techniques used in drug discovery to search libraries of molecules to identify those structures which are most likely to bind to a drug target [11]. The target in this context is usually a biological organism against which the drugs are directed—thereby resulting in a change in the said target’s behavior or function. The ligand is a molecule that produces a signal by binding to a site on a target protein.

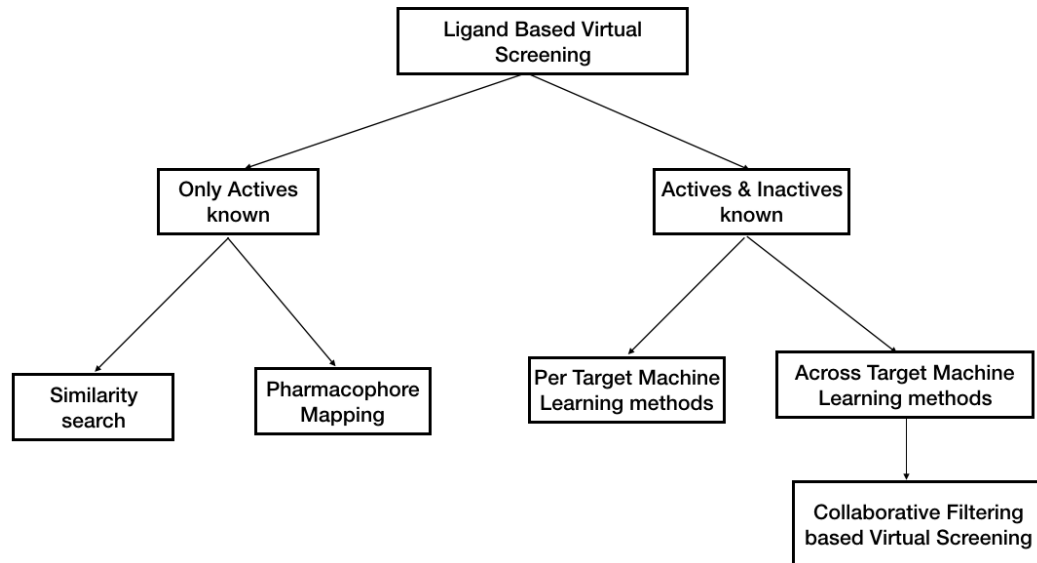


Figure 1.2: Ligand Based Virtual Screening Methods :Several classes of algorithms exist for performing ligand based virtual screening. As illustrated, the classes algorithms can be categorized by the usage of known active ligands v/s both known active and inactive ligands. The categorization of our method, utilizing collaborative filtering is also illustrated in the figure.

Ligand-based virtual screening techniques attempt to exploit prior knowledge of the abilities of ligands to bind to the protein targets to predict the effect of the candidate ligand on the said protein target. The prior knowledge of the abilities of the ligands to bind with the targets are recorded via Assays. Assays in our context are investigative procedures that qualitatively analyze a ligand’s effects on identified protein targets and are typically conducted in laboratory settings.

Several variations of the ligand based virtual screening techniques exists. The various categories of ligand based virtual screening approaches, along with the classification our method is illustrated in Figure 1.2. The technique called Pharmacophore models, begin with a known set of structurally diverse ligands that binds to a receptor, to construct a model of the receptor by deciphering the collective information contained in such set of ligands. The candidate ligand is compared to the receptor model thus constructed to then predict

binding affinities. In the molecular similarity based technique, molecular similarity is assessed between the candidate ligand and the set of ligands with known prior affinities. Alternately, there exist a class of algorithms that utilize the ligands that exhibit both known active and inactive ligands. Typically the knowledge of the prior affinities between a ligand and protein target is recorded in Assays. Within the class of such methods, approaches can be further distinguished based on how the models are trained. In the per target mode, one model per target is trained, while in the across target mode one machine learning model is trained using all targets and ligands. In summary, all virtual screening techniques can be viewed as automated computational methods of filtering candidate ligands based upon their inferred relationship with a given target. Screening virtually has a number of cost-saving advantages. However, these advantages must be reconciled with the ability to accurately represent these ligands in a form that computer programs can understand and, then, find binding ligands from relatively few training examples. We discuss the nuances around accurately representing the ligands in the next section.

### 1.3.1 Ligand Fingerprints

The numeric representation of a ligand is often referred to as a “fingerprint.” The ligand fingerprints attempt to represent the molecular structural and physicochemical features in a format comprehensible by a computer program, such as a binary vector of predetermined length. Featurizing or “fingerprinting” molecules are a necessary precondition for ligand based virtual screening. As is evident from our primer (section 1.2) above, several features describe the molecules. To begin with, this includes the atoms that make up the molecule, their counts, and the types of bonds that exist. The molecular conformation referring to any one of the infinite number of possible spatial arrangements of atoms in a molecule also form key aspects of describing the molecules. As described earlier, many molecules also exist in ‘left-’ and ‘right-handed’ forms (chirality). While the physical forms are similar, they may exhibit different behaviour in practice when attempting to bind with protein targets.

Given these complexities, it comes as no surprise that the process of fingerprinting ligands is an involved research area, with several seminal works [12–18]. Typically the aim of the ligand fingerprint methods are to represent the following:

- Molecular descriptors such as its constituents, counts, or structural fragments.
- Types of pharmacophores, which represent the features of the molecules that are necessary for interactions with protein targets. Examples include hydrogen bond acceptors or donors, positively or negatively charged groups, etc.
- Connectivity pathways, which represent a series of actions that can occur in molecules to result in a change to the protein target with which they interact

The SMILES (Simplified Molecular Line Entry System) [12] is the most popular method of representing ligands. In this representation the molecules are represented using ASCII strings with specific rules governing the encoding of the constituent atoms, bonds, simple chains, rings, and branches that may exist in a ligand. Alternately, dictionary-based fingerprints, where the ligand is inspected for presence(1s) or absence(0s) of certain structural fragments are also used for representing ligands. Other forms such as topological fingerprints, encoding the types of atoms and the paths between them, and circular or radial fingerprints, constructed by iterating through all atoms and picking an atom and including its atomic surroundings within  $N$  bonds, are also in use.

However, all existing works employ the use of explicit structure featurization in order to “fit” the problem into the workflow of traditional computer based modeling. Prior knowledge of the ligand’s physical structure and its chemical formula is a necessary condition for featurization. While knowing the ligand structure is not overly burdensome, especially given the three-dimensional nature of ligands, there also exists a reliance on traditional machine learning algorithms to map from these explicit features to a desired outcome that requires many training examples. More reliable mapping comes at the expense of needing more training assays so that the machine learning model can learn the relevant portions of the features for the desired task. Many state-of-the-art protein-ligand interaction (PLI) models use machine learning that relies on abstract descriptors of compounds/proteins as input features [19–24].

### 1.3.2 Fingerprints and Virtual Screening

Virtual screening in drug discovery relies on the ligand fingerprints to perform any meaningful similarity search among ligands. However the virtual screening techniques that use

these models, while high performing, are not free of deficiencies: the limitations of representing drug compounds and targets abstractly limits our ability to infer their binding properties [25, 26]. We argue that a critical barrier is the lack of a universal fingerprinting model that can amass knowledge about drug compounds, protein targets, and assay characteristics in a shared latent space that can be used by a variety of machine learning models, visualization tools, and compound design tools. We further argue that if the representation is completely abstract, even if it performs well at PLI (Protein Ligand Interaction) prediction, it is fundamentally limited because researchers cannot systematically create candidate compounds based on the featurization of the target ([27–29]). In this research, we propose and evaluate an alternate fingerprinting technique based on collaborative filtering and methods to generate new ligand based upon shared representations. This is further discussed in the section *Dissertation Objectives*. Next, we present a short overview of computer aided drug design, which forms the basis of Dissertation Objective 2.

### 1.3.3 Computer Aided Drug Design

While representing ligands and targets effectively for predicting PLI is an active research area, an equally influential usage of this representation is to design new drugs. Computer-aided drug design uses computational approaches to discover, develop, and analyze drugs with similar biologically active molecules. This facilitates the process by which new candidate medications are discovered [30]. Modern drug discovery involves the identification of screening hits, medicinal chemistry and optimization of those hits to increase the affinity, selectivity (to reduce the potential of side effects), efficacy/potency, metabolic stability (to increase the half-life), and oral bio availability. Once a compound that fulfills all of these requirements has been identified, the process of drug development can continue. If successful, clinical trials are developed [31]. The recent years have seen an uptick in research related to applying generative deep learning algorithms to create new drug like ligands [2, 32–34]. Deep learning based drug design techniques attempt produce novel molecular structures with desired pharmacological properties from scratch. This process has evolved to be an effective method for lead identification in the hit- and lead-finding stages of the drug discovery process.

In our research, we investigate the usefulness of our methods in facilitating the drug discovery process. Our research objective and contributions to the specific area of drug design are described in the subsequent sections.

## 1.4 Dissertation Objectives

In this section, we describe the three related but specific research objectives of this dissertation. First, we aim to discover implicit ligand fingerprints via collaborative filtering, with emphasis on their ability to predict binding. Second, we aim to develop generative methods that allow the creation of new ligands based on their implicit representation. Third and finally, we aim to employ our methods outside of the virtual screening application domain.

### 1.4.1 Aim 1: Implicit fingerprint discovery by means of collaborative filtering, for application to virtual screening

In this thesis, we investigated the use of “implicit fingerprints” that do not rely on any prior knowledge of the ligand structure for the purposes of ligand based virtual screening. Using collaborative filtering, we generated fingerprints for ligands and targets, deriving dense numeric vectors that implicitly encode important characteristics of the ligand-target pair as well as the assay experimental conditions.

Aim 1 attempts to answer the following questions with respect to ligand based virtual screening

- Can we perform ligand based virtual screening without depending on fingerprints that are explicitly featurized based on known molecular descriptors?
- Using these fingerprints, can we predict ligand protein interactions with relatively fewer training examples per protein target?

While some previous works have investigated this approach [35], our evaluation shows that our results are superior. Chapters 3 and 4 of this thesis describes in detail the experiments conducted to answer the aforementioned questions. That is, to mitigate the need for large data sets of dense assay examples by adopting an “implicit” structure model of the ligands. That is, for a given ligand, we use the assay results of other implicitly similar ligands to

help predict if a particular ligand binds to a target. The measure of similarity is only based on the results of the recorded assays, not featurized descriptors of the ligand. By using this implicit similarity, we can more readily predict if a ligand will bind to a target with far fewer training examples per target.

#### 1.4.2 Aim 2: Generative deep learning based drug design using Implicit Ligand fingerprints

As described, Deep learning based de novo drug design techniques refers to the process of producing novel molecular structures with desired pharmacological properties from scratch using generative deep learning networks. This process has evolved to be an effective method for lead identification in the hit- and lead-finding stages of the drug discovery process.

Several recent works have investigated the use of neural embedding on compound structure representations such as SMILES codes, showing this embedding is effective for exploring the chemical properties [36] and generating novel compounds. [37] refer to these embedded fingerprints as implicit representations. However, their methods work upon the raw SMILES textual representation and are therefore limited in their ability to discern more complicated relationships encoded by graphical fingerprints. Recent years have seen a plethora of deep learning based generative models for de-novo drug generation [37–42]. The common theme in these techniques is to provide as input to the deep learning model the molecules only to produce the same or similar molecules as output. The continuous vector representations of the input molecules in the intermediate layers produce a larger chemical property space, which is then sampled to produce novel molecules.

Given this context of de novo drug design and building on the successful results from our previous aim, Aim 2 attempts to answer the following questions:

- Can we design and train deep learning methods that leverage the implicit compound fingerprints to map back to the known physical structure of compounds (i.e, decoding)?
- If this decoding is possible, can we leverage the implicit fingerprints and decoder to generate novel ligands?

The implicit encoding of compounds are a continuous vector-valued representation and thus lend itself to the use of continuous optimization to generate novel compounds. We

further assess the properties of the novel ligands generated in terms of the drug-like physical properties of molecules, chemical complexity, and biological activity in Chapters 5, 6, 7, and 8.

#### 1.4.3 Aim 3: Cross domain applicability of Implicit Fingerprints for Image Generation

Finally, as a part of this research, we also wish to investigate the cross domain applicability of the said methods to an alternate domain, on a different dataset. We aim to investigate the applicability of the implicit fingerprints in generating images from the corresponding implicit image fingerprints from collaborative filtering. Our dataset for this exercise is comprised of images with meta data tags identifying the objects contained within the image. The data is represented in a matrix form, with the meta data forming the columns of the matrix and the images represented in the individual rows and the value of the data cell indicating the applicability of corresponding tag to the image. The dataset in this form is likened to the user-movie matrix or the ligand-target binding affinity matrix over which the collaborative filtering algorithm can then be applied. It is to be noted that the collaborative filtering algorithm is utilized to generate the latent vectors of the images and meta-data tags as opposed to the traditional purpose of imputing a missing data cell values.

Given this context of evaluating CFCGenNets across different domains, Aim 3 attempts to answer the following questions

- Can we leverage the latent features generated for images using Collaborative Filtering as an alternate encoded representation of images?
- Can we design and train deep learning methods that leverage the implicit image fingerprints to map back to the original images?

### 1.5 Dissertation Contributions

In this section, we present the contributions stemming out of our work. The section is organized by the corresponding research aim that resulted in the contribution.

#### 1.5.1 Contributions from Research Aim 1

Our contributions resulting from our work on Research Aim 1 [43] are as follows:



1. An investigation into how collaborative filtering methods can be used to predict binding affinity of ligand-target pairs. We compare collaborative filtering methods to traditional machine learning methods that use explicit fingerprinting from the RDKit package, showing collaborative filtering performs on-par with other methods in terms of all evaluation criteria, including enrichment factor even without the knowledge of a ligand’s explicit physical structure.
2. An evaluation of collaborative filtering categorized by the amount of required training assays needed for a target of interest, showing that collaborative filtering has a significant performance advantage when the number of training assays for a given target is relatively low.
3. An introduction of “Implicit Target and Ligand Fingerprints”, a new type of ligand fingerprinting and target fingerprinting derived from the latent factors employed by the collaborative filtering method.

### 1.5.2 Contributions from Research Aim 2

In particular, our contributions resulting from our work on Research Aim 2 [1] are enumerated below. Additionally, we also note that domain specific semantic evaluation of the novel compounds generated by our methods were done in collaboration with another researcher (Niraj Verma, co-author on [1]). The details of the areas contributed by the collaborator are described in Chapter 8.

- **CFGenNet**: A novel class of generative deep learning algorithms based on the implicit fingerprints obtained from collaborative filtering.
- An evaluation of how the implicit ligand fingerprints from collaborative filtering translate into discrete representations of molecules (SMILES).
- The design of new compounds based upon the latent representation without explicitly optimizing for chemical properties, thereby encoding drug-like properties including binding affinities to known proteins. We evaluate the uniqueness of generated compounds from CFGenNet and if novel compounds have similar binding properties to

closely related compounds. Binding properties are evaluated using multiple computational models.

### 1.5.3 Contributions from Research Aim 3

Our contributions resulting from our work on Research Aim 3 are as follows:

- CFGenNets for Images : A successful investigation into the viability of leveraging the concepts of CFGenNets into the domain of Image processing
- Composite CFGenNets : Successful design and development of methods that combine the powers of Variational Autoencoders and CFGenNets by creating an enhanced version of CFGenNets called Composite CFGenNets.
- Image Manipulation using Composite CFGenNets : Introduction of a novel mechanism to manipulate images by leveraging Composite CFGenNets

The details related to the above contributions are discussed from Chapter 9.

## 1.6 Definitions

Table 1.1: Definitions

Molecules	Basic unit of any chemical compound or ligand, with 2 or more atoms that take part in a chemical reaction.
Molecular Bonds	Forces that hold atoms together within molecules. Can be of type covalent or ionic.
Target / Biological Target / Protein Target	Biological organism against which the drugs are directed thereby resulting in a change in the target's behavior or function
Ligand	Molecule that produces a signal by binding to a site on a target protein
Assays	Investigative procedures that qualitatively analyze a ligand's effects on identified protein targets and are typically conducted in laboratory settings
Virtual Screening	Computational techniques used in drug discovery to search libraries of molecules to identify those structures which are most likely to bind to a drug target
Ligand Fingerprints	Representation of molecular structural and physicochemical features in a format comprehensible by a computer program, such as a binary vector of predetermined length
SMILES	Simplified Molecular Line Entry System : a method of representing ligands using ASCII strings

## Chapter 2

### Collaborative Filtering

In this chapter, we provide an overview of collaborative filtering. It is to be noted that portions of the subsequent sections have been published as a part of our journal article [43], where I am the first author of the publication.

#### 2.1 Overview of Collaborative Filtering

As described in Chapter 1, collaborative filtering algorithms have been historically used in the context of designing recommendation systems such as movie recommendation engines, as well as up-sell and cross-sell recommendation engines for e-commerce sites. In general, collaborative filtering is a method for making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from other users (collaborating) [8]. In general, collaborative filtering methods can be categorized into two groups of methods: the neighborhood methods [44] and matrix factorization methods [45], also known as latent factor models. Neighborhood methods compute the relationship between items and/or users to identify similar items or like minded users to help predict ratings. Latent factor methods try to explain the ratings by characterizing the items and users on 20 to 100 factors, derived entirely from past rating patterns.

##### 2.1.1 Neighborhood-based Collaborative Filtering

Neighborhood methods (also called memory-based methods) evaluate the relationships between items and users by approximating the relative distance between users. In this scenario, there is a large user-item matrix,  $A$ , with users in the rows and items in the columns. A particular rating between a user,  $u$ , and item,  $i$ , is denoted as  $a_{u,i}$ . This matrix is typically sparse, with only a handful of ratings for each item per user. The general concept is to find similar users by taking the similarity among each row of  $A$ . In this method the system evaluates a user's preference for an item based on ratings of similar users that have

also rated that particular item [44]. More formally, we define this process for a particular user,  $u_0$ , and item,  $i$ , as:

$$a_{u_0,i} = \frac{1}{|U|} \sum_{u \in U} a_{u,i}$$

where  $U$  is the set of all similar users that have also rated item  $i$  and  $|U|$  is the total number of similar users. Variants of this measure also exist where the  $a_{u,i}$  measure is weighted, for example, by the relative similarity of users. Similarity of users can be calculated using various distances. Common measures of distance include Euclidean, Cosine, and Pearson Dissimilarity. Variants also exist on the “rules” for judging similar users. Some methods look for the top- $N$  similar users, whereas other methods employ a distance threshold for discerning which users are similar. In general, neighborhood methods tend to work well for a number of different applications, but suffer from computational issues when the user item matrix  $A$  is large or very dense.

### 2.1.2 Matrix Factorization Method

Matrix factorization methods [8] have been one of the most popular implementation techniques of latent factor recommendation systems. These methods find lower dimensional representations of the full user-item matrix. The dimensions of the lower dimensional representations are often called factors. In the context of movie recommendation engines, the discovered factors from matrix factorization methods have been studied extensively. While there is no guarantee that the factors found represent an interpretable quantity, many times the factors can be identified as representing a number of interesting item and user properties (even though the modeling does not explicitly use any features of the user or item). For example, in movie recommendations these factors often “encode” obvious factors such as comedy versus drama, amount of action, or orientation to children. They can also represent less well-defined dimensions such as depth of character development, quirkiness, or they might be completely uninterpretable dimensions. For users, each factor represents how much the user likes movies that score high on the corresponding movie factor [8]. For the target-ligand application, then the factors might encode properties of the binding sites for the target or chemical properties of the ligands. We illustrate the metaphor in Figure 2.1.

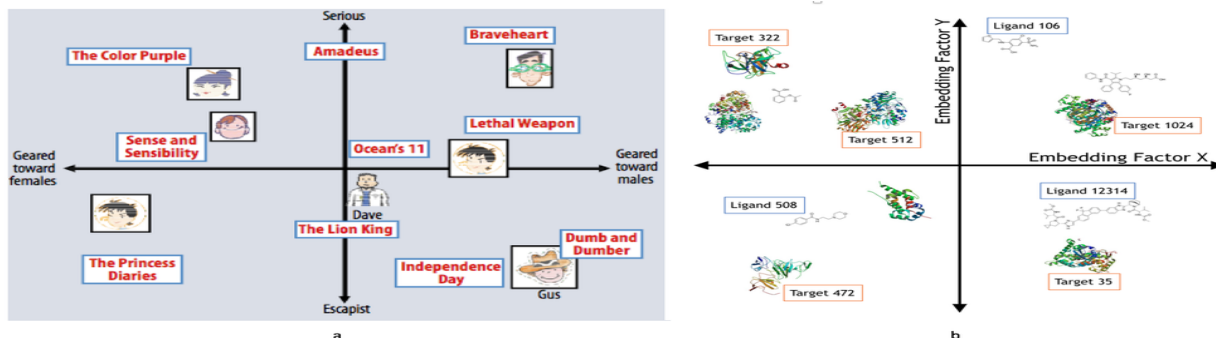


Figure 2.1: Latent Factor Embedding : Sub Figure a illustrates the concept of latent factor the latent factor recommendation for a movie recommendation engine. The latent factor method relies on learning hidden factors from the user-movie ratings alone. In this simplified example the system learns two dimensions from the ratings and places the movies and users in this 2D space. The users predicted rating of the movie would be a dot product of the user’s and movie’s location in the 2D space. Figure b illustrates the same concept for a target-ligand embedding. Here the latent factors correspond to properties that the ligands and protein targets can be jointly modeled with. The properties may correspond to a distinct chemical property, but might also pertain to a factor not well described by traditional cheminformatics.

When applied to the domain of virtual screening, this method involves representing the ligands and targets as vectors of factors with dimensionality  $f$ , to represent the latent factor space, where  $A \approx \hat{A} = P \cdot Q$ . Here the affinity predictions are modeled using the well established singular value decomposition method [46] and optimization procedures to minimize the reconstruction error between  $A$  and  $\hat{A}$ . That is, we model each known affinity  $a_{t,l} \in A$  between ligand  $l$  and target  $t$  as the following dot product of vectors  $p_t$  and  $q_l$

$$a_{t,l} = p_t \cdot q_l$$

where each known ligand  $l$  is associated with vector  $q_l$  and each target associated with vector  $p_t$ . Both  $q_l$  and  $p_t$  contain  $f$  elements. This operation is often represented in matrix form when there are  $L$  unique ligands and  $T$  unique targets in the database:

$$\hat{A} = P \cdot Q = \underbrace{\begin{bmatrix} \leftarrow & p_1 & \rightarrow \\ \leftarrow & p_2 & \rightarrow \\ & \vdots & \\ \leftarrow & p_t & \rightarrow \\ & \vdots & \\ \leftarrow & p_T & \rightarrow \end{bmatrix}}_{\text{target-factor matrix}} \cdot \underbrace{\begin{bmatrix} \uparrow & \uparrow & & \uparrow & & \uparrow \\ q_1 & q_2 & \dots & q_l & \dots & q_L \\ \downarrow & \downarrow & & \downarrow & & \downarrow \end{bmatrix}}_{\text{factor-ligand matrix}}$$

The optimization step involves learning the factor vectors  $q_l$  and  $p_t$  by minimizing the regularized square error on the set of known affinities,  $a_{t,l}$ , using standard optimization techniques such as stochastic gradient descent algorithms [47].

$$\min_{p,q} \sum_{t,l \in \Lambda} \underbrace{(a_{t,l} - p_t \cdot q_l)^2}_{\text{mean square error}} + \underbrace{\lambda \cdot (||p_t||^2 + ||q_l||^2)}_{\text{regularization}}$$

where  $\Lambda$  is set of  $t, l$  for which the affinities  $a_{t,l}$  is known. The optimization function also includes the regularization term with the regularization parameter  $\lambda$  to help minimize overfitting. Figure 2.2 illustrates the method with an example. The highlighted cells in the matrix  $A_{t,l}$  represents the known affinities between a hypothetical set of 6 ligands and 12 targets. The matrix factorization involves employing single value decomposition [46] method to construct matrices  $Q$  and  $P$  with  $f = 3$  factors in this example. The optimization method involves reducing the square error between the known affinities and the predicted values of the known affinities resulting from  $P \cdot Q$ . Variants of the factorization methods also exist where the dot product is approximated in a larger dimensional space using kernels,  $\kappa(p, q) = \phi(p_t) \cdot \phi(q_l)$ , where  $\phi$  is a transformation of the vectors into a higher dimensional space. However, it has been previously shown that this is typically poor performing for target-ligand binding affinity prediction [35].

### 2.1.3 Conclusion

In this chapter, we discussed the concepts of collaborative filtering and its applicability to ligand based virtual screening. In the subsequent chapter, we provide the details of the experiments conducted.

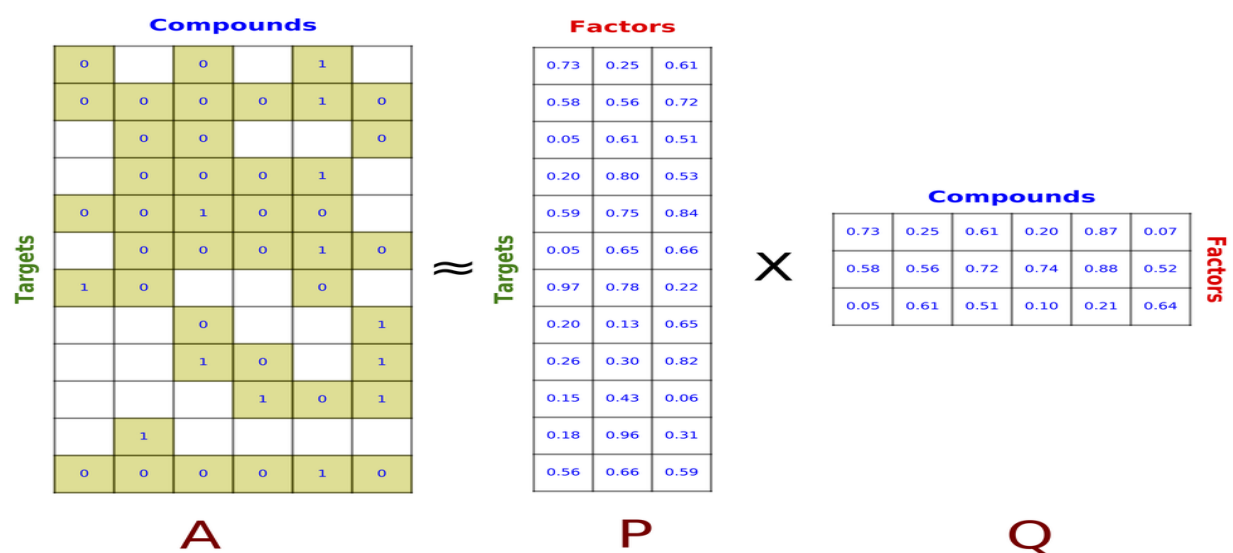


Figure 2.2: Illustration of the matrix factorization method :In this example the highlighted cells represent the known affinities. The SVD method generates the matrices  $P$  and  $Q$ . Optimization methods are employed to minimize the error between the known affinities and their predicted values.



## Chapter 3

### Ligand based Virtual Screening using Collaborative Filtering

In this chapter, we provide extensive details of our investigation into utilizing collaborative filtering for performing ligand based virtual screening. It is to be noted that parts of the subsequent sections have been published as a part of our journal article [43], where I am the first author.

#### 3.1 Data Preparation

Evaluating the applicability of collaborative filtering for performing virtual screening requires exhaustive dataset to work with. For this purpose, We use ligand-target bioactivity data from the ChEMBL database (Version 23). In an effort to keep the evaluations consistent with previous studies [48], we focus exclusively on human targets and three types of binding affinity measures: half maximal inhibitory concentration ( $IC_{50}$ ), half maximal effective concentration ( $EC_{50}$ ), and the inhibitory constant ( $K_i$ ). When more than one binding concentration measure was present in the database, we use the  $K_i$  measure. When  $K_i$  is not present we look at  $IC_{50}$  and then  $EC_{50}$  to categorize the ligand-target pair as active or inactive. To convert data into the binary active-inactive format the following concentration thresholds were applied:  $< 100$  *nM* for “actives” and  $> 1000$  *nM* as “inactives.” We note that many other works apply two thresholds to the dataset and the interactions between the two ranges are discarded, as their classification is subjective [48, 49]. We also note that the thresholds selected in our study are consistent with the standardized activity values of the ChEMBL database.

The data selection methods described above resulted in a bioactivity matrix of size 241,260 (ligands) by 2,739 (targets), with about 0.15% of the matrix containing a real value. The mean inactive:active ligand ratio across targets in the dataset was approximately 7:3.

Figure 3.1 illustrates the distribution of the targets by the number of recorded assays

available for predictions. It was observed that more than half of the targets in the data set had less than 200 recorded assays (including actives and inactives).

We note that there have been recent reports in the literature [49, 50] stressing that the overall model performance is highly dependent on threshold selection for inactive/active as well as the ratio of inactive to active examples. Also, many ligand-target databases are biased in that the experimental data they are comprised of represents only a small, nonuniform portion of the chemical space. This leads to the over representation of certain types of ligand-target patterns. Furthermore, experimental binding affinity measures are often difficult to reproduce [51], which means there is inherent noise in the datasets such that perfect classification of any test set should not be possible (unless models are over-fitted to the problem). Although this has been a common knowledge within the community [52], it nevertheless remains largely un-addressed.

Following recommendations from [50], we computed the bias of our training and test selection to provide an estimate on how trustworthy our evaluation metrics are.

### 3.1.1 Evaluation Criteria

In this study three evaluation criteria are employed: the area under the receiver operating characteristic (AUC), the enrichment factor (EF), and the Boltzmann-enhanced discrimination of the receiver operating characteristic (BEDROC). We briefly describe our rationale in selecting these criteria, as well as an overview of the strengths and weaknesses of each criterion.

Our first evaluation criterion is the area under the receiver operating characteristic curve (ROC [53, 54], the curve that results when the true positive rate is plotted as a function of the false positive rate). This measure ranks ligands based on their predicted probability of being active. An AUC value greater than 0.5 suggests that the classifier is better than chance at assigning an active/inactive target-ligand pair. While widely reported in a number of papers that use machine learning for target-ligand classification, the AUC does not capture important aspects of the virtual screening problem. Specifically, the challenge is to rank active ligands for a given target from the entire dataset. A superior classification model would have a high true positive rate for the highest ranked ligands, which are the ones that

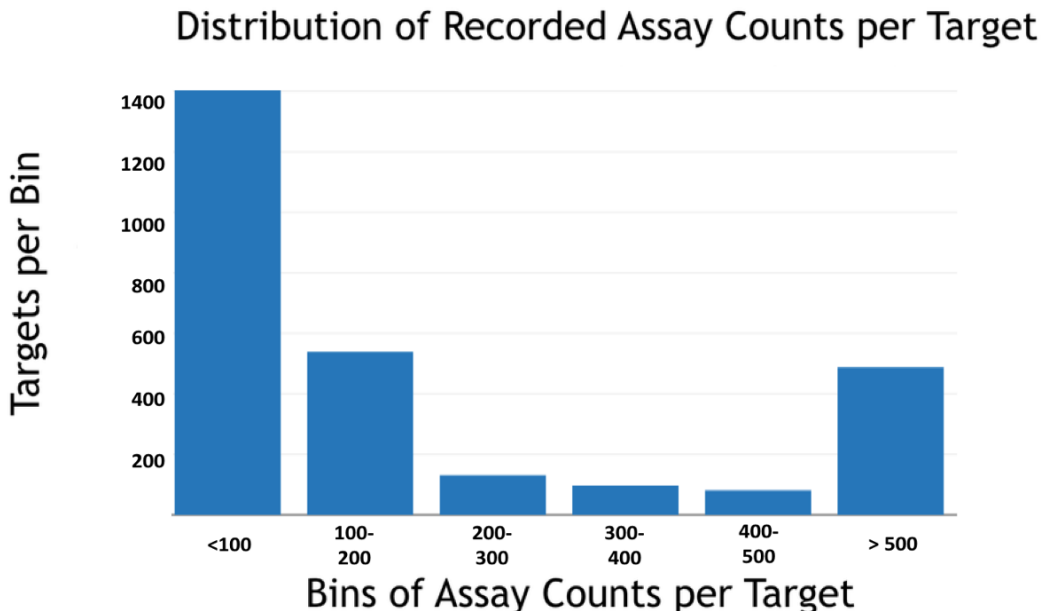


Figure 3.1: Number of targets by known assay counts : Illustration of the distribution of the number of targets by the known assay counts for each target. More than half of the targets have less than about 200 recorded assays. Therefore, it is imperative that the virtual screening methods perform well when the number of assays is relatively sparse.

would be assayed first (the so-called early recognition problem). The AUC does not take into account this early recognition, so it can incorrectly judge a classification model superior if it has an overall high true positive rate, even though the true positives may not occur “early” in the ranking of ligands. Such a model would result in many needless assays before becoming sufficiently accurate. Therefore, more suitable metrics are often sought after that do take into account early recognition – the two popular choices being Enrichment Factor (EF) and Boltzmann-Enhanced Discrimination of Receiver Operating Characteristic (BEDROC).

Enrichment Factor [55, 56] is defined as the ratio of correctly classified active ligands within a predefined early recognition threshold compared to the total ligands selected by the model, further normalized by the expected random selection of the ligands.

$$EF_{X\%} = \frac{Compounds_{selected}/N_{X\%}}{Compounds_{total}/N_{total}}$$

where  $N_{X\%}$  is the number of ligands in the top  $X\%$  of the ranked ligands.  $EF_{1\%}$ , then, is

the ratio of true actives found in the top 1% of ranked ligands from a model normalized by the total number of actives for a specific target. In other words, it gives an estimate on how many more actives can be found within the early recognition threshold compared to a random distribution. While this criterion closely matches the virtual screening problem, it is not appropriate to compare the EF values obtained for different datasets when their number of actives differ. Another disadvantage of the criterion is that it assigns equal weights to the actives within the threshold, without any knowledge that some actives bind extremely well and others have higher  $K_i$  concentrations. Robust Initial Enhancement (RIE) [57] helps mitigate this by comparing two scenarios, (1) when the most active ligands are ranked at the beginning of the threshold and (2) when the most active ligands are ranked closer to the end of the threshold. This is achieved by applying continuously decreasing exponential weight when ranking ligands. The RIE metric is similar in meaning to EF in that it quantifies the superiority (to random) of the exponential average of the distribution generated by the ranking method. Its minimum and maximum value dependance on (apart from the pre-exponent factor  $\alpha$ ) the number of actives and the dataset size contributes to the metric disadvantages. Nevertheless, RIE’s desirable property of differentiating actives within the ordered list serves as a driving force for the development of the BEDROC metric, discussed next.

Bound between 0 and 1, the BEDROC metric [58] is interpreted as the probability that an active in the ordered list will be ranked before a ligand that is drawn from a random probability distribution function. The shape of the distribution is governed by the pre-exponent factor  $\alpha$ , that must be selected by the user. In the words of the original authors: “It is to be noted that  $\alpha$  should not be chosen in such a way that it represents the best performance expected by a ranking method, but rather it should be considered as a useful standard to discriminate better or worse performance in a real problem to which the ranking method will be applied.” [58]. Our study chose an  $\alpha = 20$  based on the previous study by Riniker *et al.* [17] in their benchmarking of fingerprints for ligand-based virtual screening.

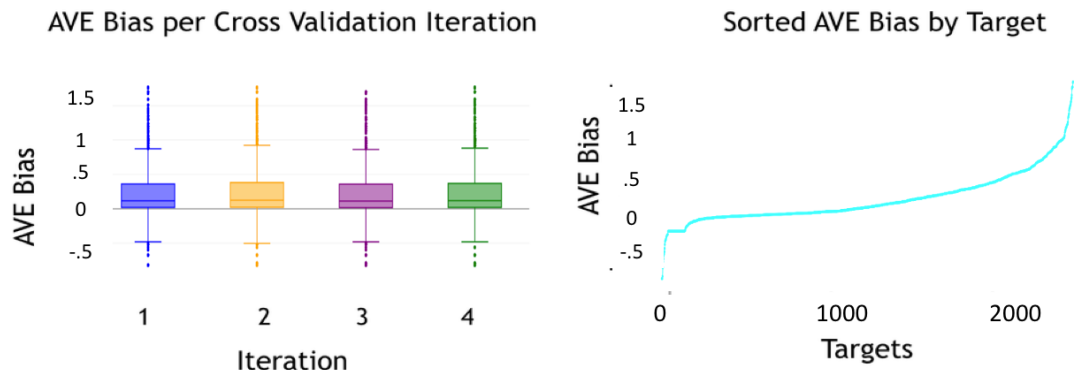


Figure 3.2: AVE Bias : The image to the left visualizes the comparison of AVE Bias across 4 sets of training and validation data. The figure shows that the 4 training and validation sets are randomly split with similar bias measures across the sets, thereby minimizing the impact of variance across our study. The image to the right shows the spread of the AVE Bias for each target in the dataset. The distribution of the AVE Bias from -0.5 to 1.5 (where closer to 0 indicates no bias) across targets facilitates the study of the resilience of the collaborative filtering algorithm to the impact of such a bias.

### 3.1.2 Variance and Bias in Validation Set Selection

The performance of machine learning models can be impacted by variance. A model with high variance performs inconsistently on different validation sets. A model with high bias is one that is well fitted to the training data but fails to generalize well. Building machine learning models by using cross validation to separate training and validation can aid in quantifying variance. In our experiments, the train and the test split was randomly generated by stratifying on the targets, generating a random split with the ratio of 70%:30% of associated ligands by target between the training and the validation sets.

In order to minimize the effects of variance influencing the claims in our study, we use four iterations of our tests including generating four sets of training and validation data and building the above mentioned models using implicit and explicit methods for each set of training and validation data. Modeling bias is another design consideration while building machine learning models to ensure the ability of the models to generalize beyond the training

and validation datasets. In the field of computational chemistry the accuracy in practice is not as good as the benchmark results from previous virtual screening results [8].

Wallach and Heifets [50] introduced a new measure of evaluating the redundancies in the training - validation sets called the Asymmetric Validation Embedding Bias (AVE Bias). The AVE Bias measures the quality of the training and validation sets by measuring the similarities between the actives and inactives in the validation sets with the actives and inactives in the training sets. The Bias is mathematically defined as [50]:

$$B = \left[ \underbrace{H_{(V_a, T_a)} - H_{(V_a, T_i)}}_{AA-AI} \right] + \left[ \underbrace{H_{(V_i, T_i)} - H_{(V_i, T_a)}}_{II-IA} \right]$$

where,  $V_a, V_i$  represent the validation sets with active and inactive ligands,  $T_a, T_i$  represent the training sets with active and inactive ligands respectively, and  $H_{(.)}$  represents a measure of cluster similarity between the sets. The left  $AA-AI$  term of the AVE Bias is a measure of how clumped the validation actives are among the training actives. The right  $II-IA$  measures the degree of clumping among the inactives. The study showed that the performance of the ligand based screening methods strongly and positively correlated with the AVE Bias. Intuitively, it means that an algorithm might perform well because there is an inherent difference in the training and validation sets employed that makes the problem more easily separable for the validation set. Positive values indicate a bias, whereas negative values indicate that the problem is increasingly difficult because the training and validation sets differ in a way that makes the classification problem even more difficult. The AVE Bias can be measured for every target in the dataset. Therefore, it is appropriate to present the AVE Bias results as a histogram or boxplot of the values.

As described, we use four stratified shuffle splits of the data. We computed the AVE Bias for all targets in each of the four iterations of train-validation sets. In our calculations, we employed Tanimoto similarity [13] to compute  $H_{(.)}$ . Figure 3.2 illustrates the AVE Bias scores using a boxplot to summarize the AVE Bias per target (a separate boxplot is used for each iteration of the shuffle split). All four sets of train and validation sets have similar AVE Bias measures. The imbalanced nature of our data set (inactives to actives, 7:3) in addition to the method of classifying molecules as active or inactive based on the  $100nM$  and  $1000nM$

concentration thresholds and the randomized selection of training and validation set resulted in a good distribution of targets with varying AVE Biases across targets. In addition, it was also observed the data set also contained a subset of targets with a negative bias, thus makes the classification of actives and inactives challenging for these sets [50]. Separate plots are shown for each iteration of the validation split. Each box plot represents distribution of the AVE Bias calculated per Target for each iteration of Training and validation sets. Figure 3.2 also illustrates the AVE Bias for each target in the dataset.

### 3.2 Collaborative Filtering Model Selection

In order to identify the best performing collaborative filtering model, we employ two randomized grid searches. The first grid search investigates parameters for neighborhood CF methods. The second grid search investigates parameters for using matrix factorization CF methods. We separate the grid searches because the parameters in each algorithms are quite different. We use collaborative filtering algorithms implemented in the GraphLab API [59].

For the neighborhood CF methods we investigate three separate distance metrics: Jaccard, Cosine, and Pearson. We also investigate a number of threshold values to determine neighborhood size, logarithmically spaced from  $10^{-7}$  up to  $10^{-1}$ . For each combination of hyper parameters, we calculate the mean for each of our evaluation criteria: AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> scores. The best performing neighborhood CF method utilized the Pearson similarity metric with a threshold of  $10^{-2}$ . The mean AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> scores were 0.79, 0.72, and 4.22%, respectively.

Table 3.1 summarizes additional results for the top 5 models, sorted by the EF<sub>1%</sub> score. For the matrix factorization-based CF models, we investigated the number of latent factors, ranging from 5 to 60 in increments of 5. We also investigated the value of the regularization constant,  $C$ , used in the stochastic gradient descent method [47], with values logarithmically spaced from  $10^{-7}$  up to  $10^{-1}$ . Finally, we swept the values of the step size used for updates in the stochastic gradient descent (SGD) optimization, spaced logarithmically from  $10^{-3}$  up to  $10^{-1}$ . We remind that the factorization CF method learns latent factors for each ligand and for each target and uses them to rank ligands according to the likelihood of observing

Table 3.1: Collaborative Filtering hyper-parameter tuning : Tabulated top 5 results from training multiple collaborative filtering models using neighborhood methods and matrix factorization methods.

<b>Distance</b>	<b>N. Threshold</b>	<b>AUC</b>	<b>BEDROC<sub>20</sub></b>	<b>EF<sub>1%</sub></b>
[0.3pt](r0.25em)1-2[0.3pt](r0.25em)3-5Pearson	$10^{-2}$	0.791	0.723	4.225
Pearson	$10^{-5}$	0.792	0.723	4.225
Pearson	$10^{-4}$	0.791	0.723	4.225
Jaccard	$10^{-2}$	0.648	0.647	3.670
Cosine	$10^{-5}$	0.644	0.647	3.670

<b>Num. Factors</b>	<b>SGD Step Size</b>	<b>AUC</b>	<b>BEDROC<sub>20</sub></b>	<b>EF<sub>1%</sub></b>
[0.3pt](r0.25em)1-2[0.3pt](r0.25em)3-550	$10^{-3}$	0.891	0.929	5.476
32	$10^{-4}$	0.860	0.889	5.028
32	$10^{-2}$	0.899	0.872	4.994
25	$10^{-4}$	0.868	0.867	4.765
25	$10^{-2}$	0.892	0.847	4.547

those (target, ligand) pairs. The stochastic gradient descent algorithm was used as the optimization function to minimize the mean square error between the known affinities and their predictions. Table 3.1 summarizes the results of the factorization CF method, sorted by EF<sub>5%</sub> (SGD step size not shown for brevity). From the table, it is clear that the performance of the factorization CF method exceeds that of the neighborhood recommender. The best model was found with 50 latent factors, a value of  $C = 10^{-3}$  for regularization, and  $10^{-3}$  for the SGD step size. The best model had a mean AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> as 0.89, 0.92, and 5.47, respectively.

To more clearly understand how the hyper-parameters change the performance of the factorization CF algorithm, we plot the BEDROC<sub>20</sub> values as each hyper-parameter value changes. Figure 3.3 illustrates the effect of the number of latent factors on the performance of the model. It was observed that the performance plateaus near 50 factors. Similarly Figure 3.3 illustrates the performance of the regularization parameter of the SGD optimizer [47] on the model performance. It is noticed that there is no consistent effect of the value of regularization on the performance of the model.



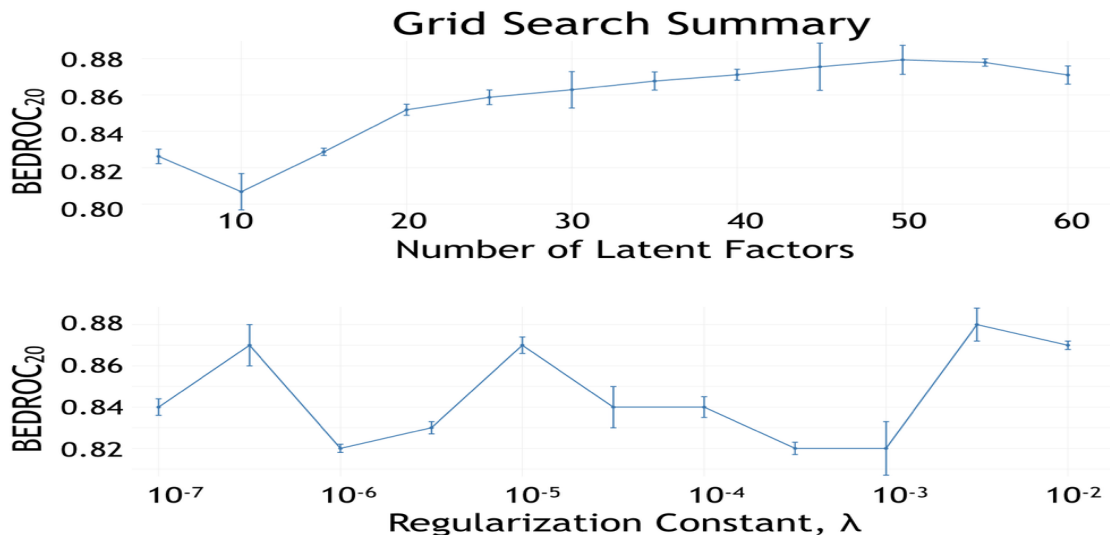


Figure 3.3: Hyperparameter tuning for matrix factorization collaborative filtering : Illustration of the effects of the number of factors,  $f$ , and regularization parameter,  $\lambda$ , on model performance. While model performance improves with more factors, it may lead to over fitting. The regularization parameter does not have a consistent performance, revealing that performance is not sensitive to  $\lambda$ . The error bars represent the standard error of the mean across the BEDROC<sub>20</sub> scores for each Target at the specified parameter value

### 3.2.1 Conclusion

In this chapter, we provided exhaustive details on the methods utilized to train collaborative filtering model on ligand assay data. In the subsequent chapters, we provide an in-depth analysis of the relative advantages of collaborative filtering compared to the traditional virtual screening methods employing explicit featurization.

## Chapter 4

### Performance Analysis of Collaborative Filtering for ligand based virtual screening

In this chapter we provide an in depth analysis of the outcomes observed with collaborative filtering methods when utilized for performing ligand based virtual screening. Furthermore, we compare the outcomes of collaborative filtering with the traditional virtual screening methods that employ explicit featurization.

#### 4.1 Overall Performance of Collaborative Filtering

In the following section, we analyze the comparative performance of the collaborative filtering model (which uses implicit featurization) and the baseline models (which use explicit featurization from structure modeling). We use the evaluation criteria explained previously: AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub>. Figure 4.1 shows boxplots of performance per target across all the algorithms and all evaluation criteria. That is, a performance criterion is calculated for each target and then all values are displayed in each boxplot. Results from all cross validation iterations are combined in each boxplot.



Figure 4.1: Comparison of AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> scores across all algorithms : The collaborative filtering based implicit structure method, based only on assay outcomes performs on par with the other methods across the three evaluation criteria. The across target Random Forest model is the next best performer. All other models perform about the same and poorer than collaborative filtering.

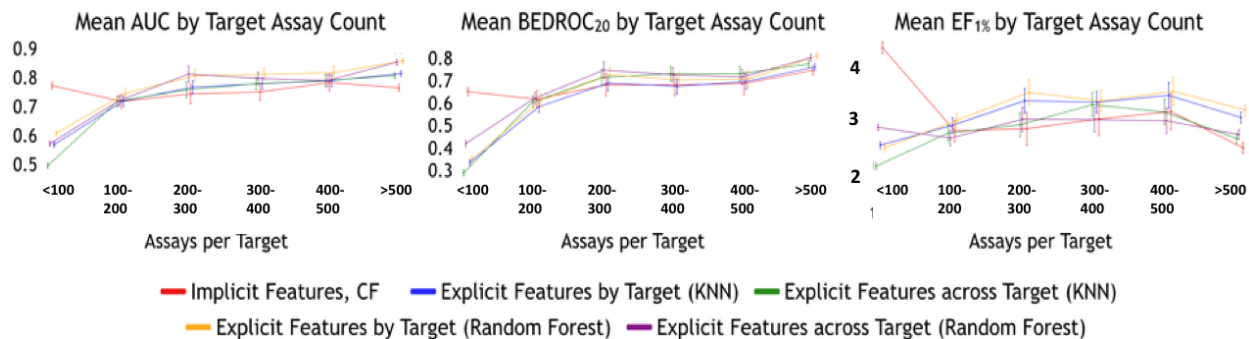


Figure 4.2: Comparison of AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> scores by number of known activities : Illustration of the comparative performance of the algorithms by the number of known affinities per target in the training data. Error bars correspond to standard error. The collaborative filtering based implicit structure methods significantly outperform other algorithms when the training data has 100 or fewer activities. Beyond 100 activities the performance of all algorithms converge.

Across all evaluation criteria collaborative filtering performs similar to the baseline methods, despite the absence of known structural information of the ligands. In general, collaborative filtering is a slightly superior performer, followed by the *across target RF* model, followed by *per target RF*, and *KNN* for *across target* and *per target* rounding out the bottom. When taking the AVE Bias into account, the average AUC scores of the models generated by RF and KNN are consistent with the performance of the same algorithms on the unbiased training and validation tests for the J and J Benchmark reported by Wallach and Heifets [50].

Although we have shown performance for all cross validation iterations combined, similar performance was observed individually for each validation set.

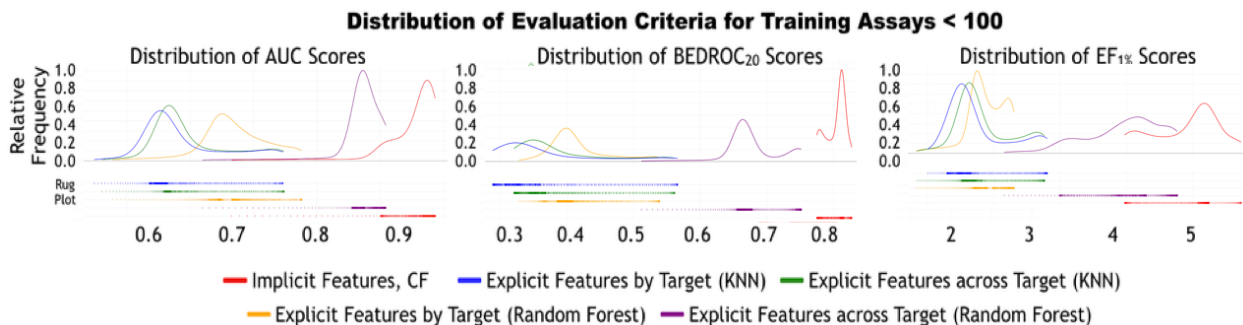


Figure 4.3: Distribution of AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> : Distribution of AUC, BEDROC<sub>20</sub>, and EF<sub>1%</sub> scores across all algorithms when the number of known affinities per target is less than or equal to 100 activities. Based on a two-tailed t-test of the scores ( $p < 0.01$ ), the difference in scores for collaborative filtering with other methods is statistically significant.

## 4.2 Performance Comparisons by Number of known Actives

While the overall performance of collaborative filtering is encouraging, it is still unclear if the implicit features employed help to mitigate the need for large numbers of training assays for each target. To help delineate this, we now focus our attention on grouping the results by how many training assays were used to model a given target. We remind from Figure 3.1 that the number of training assays per target is typically less than 200, which comprises the majority of targets in the ChEMBL database. As such, it is desirable for an algorithm to perform well even when the number of training assays is relatively low. Figure 4.2 shows the performance of the algorithms on the validation sets with results grouped by the number of available training assays. That is, each performance criteria is calculated for each target and, then, the results are grouped by the number of assays used in the training of that target. For example, targets with less than 100 training assays are in the first bin, 100-200 in the second bin, and so on until targets contain more than 500 training assays.

We then clumped together all targets with more than 500 assays into the last bin. This approach enables us to compare the relative performance of the algorithms based on the amount of available training assays. A consistent pattern was observed across all the three evaluation criteria. The predictions from collaborative filtering on the validation set significantly outperform the baseline methods when there were 100 or fewer assays in the training set based on a two tailed t-test ( $p < 0.01$ ). Between 100 - 500 training assays, the implicit feature modeling methods perform statistically no different than the baseline models ( $p > 0.5$ ). Beyond 500 training assays, the traditional baseline method using the Random Forest algorithm outperforms collaborative filtering.

It is interesting to note that the average enrichment factor in the top 1% is the highest across all methods when there are not many training assays available per target. We hypothesize that collaborative filtering becomes less effective (in terms of  $EF_{1\%}$ ) when the number of ligands already tested is relatively numerous because the most likely candidates have already been assayed by chance.

To further evaluate the differences between algorithms when the training assays are relatively sparse, we expand on the performances of the AUC,  $BEDROC_{20}$ , and  $EF_{1\%}$  scores for targets with less than or equal to 100 training assays. Figure 4.3 represents the kernel density plots for the distribution of the three aforementioned evaluation criteria. Kernel density estimation is a means of estimating a smooth distribution from a finite set of points, similar in spirit to a histogram. Below each density estimate is a "rug plot" of the values for each machine learning method, with one "tick" for each observed value (this makes outliers easier to see as they can easily be hidden in kernel density estimates). The distribution of the scores from the implicit structure methods using collaborative filtering has a clear and visual separation from the other baseline methods using explicit structures. With the exception of *across target RF*, no other model has overlap in the performance for any evaluation criteria. Even so, based on a two-tailed t-test of the distributions, collaborative filtering is the significantly best performing algorithm in terms of all evaluation criteria when the number of training assays is less than 100.

#### 4.2.1 Conclusion

In this chapter we provided an indepth look into the performance of collaborative filtering algorithm for the problem at hand . In addition we also analysed the relative differences between our methods with the known traditional methods utilizing the explicit ligand fingerprinting methods. Because of the clear performance separation of collaborative filtering, we conclude that the implicit structure methods demonstrate a consistent and significantly increased performance when the number of known assays is limited to about 100 assays. We also conclude that when the number of training assays is greater than about 500, traditional methods provide an increased performance. Unlike these baseline methods that use molecular fingerprinting, collaborative filtering methods can “learn” about the ligands based on their affinities with other targets and vice-versa, even with fewer numbers of known assays per target. This aspect of the collaborative filtering method contributes to the better performance even with relatively sparse assay counts.

## Chapter 5

### Introducing Implicit Target and Ligand Fingerprints

#### 5.1 Overview

In this chapter, we turn our attention to an initial analysis of how the latent factors computed by collaborative filtering, by virtue of the information encoded within them, can be used to complement the traditional molecular and target fingerprints.

We introduce a new type of fingerprinting technique called *Implicit Fingerprints*, which are the latent factors determined by the collaborative filtering method on the known affinities. From our grid search, we found that 50 latent factors for each ligand and target was the most optimal performing number of factors for the matrix factorization. We further evaluate the applicability of the 50 latent factors to identify and cluster known cancer and thyroid related protein targets. Greater than 20% of industrial cancer drug development programs focus on a small subset of proteins when approximately 20,000 possible proteins are known [60]. While there have been studies investigating the pharmaceutical vulnerabilities of these proteins, the challenges with costs and off-target effects have been a limiting factor in realizing the proteins' clinical potential. We investigated the inherent properties of the implicit fingerprints to identify targets with similar binding affinities based on prior assays results. For the purposes of the tests, we re-trained a collaborative filtering model without discarding the assays with concentration levels 100nM and 1000nM. The experiment was rerun as a multi-label affinity classification problem with the following labels to indicate binding affinities :  $< 1\text{nM}$  was associated with the affinity label of 1, between 1nM and 100nM was labeled as 2, 100nM and 1000nM labeled as 3 and greater than 1000nM labeled as 4.

To visually elucidate the power of *Implicit Target Fingerprints*, we mapped the dimensions of the 50 latent factors of the targets into a 2-dimensional space using t-distributed stochastic neighbor embedding (t-SNE) [61], a powerful method for reducing dimensionality

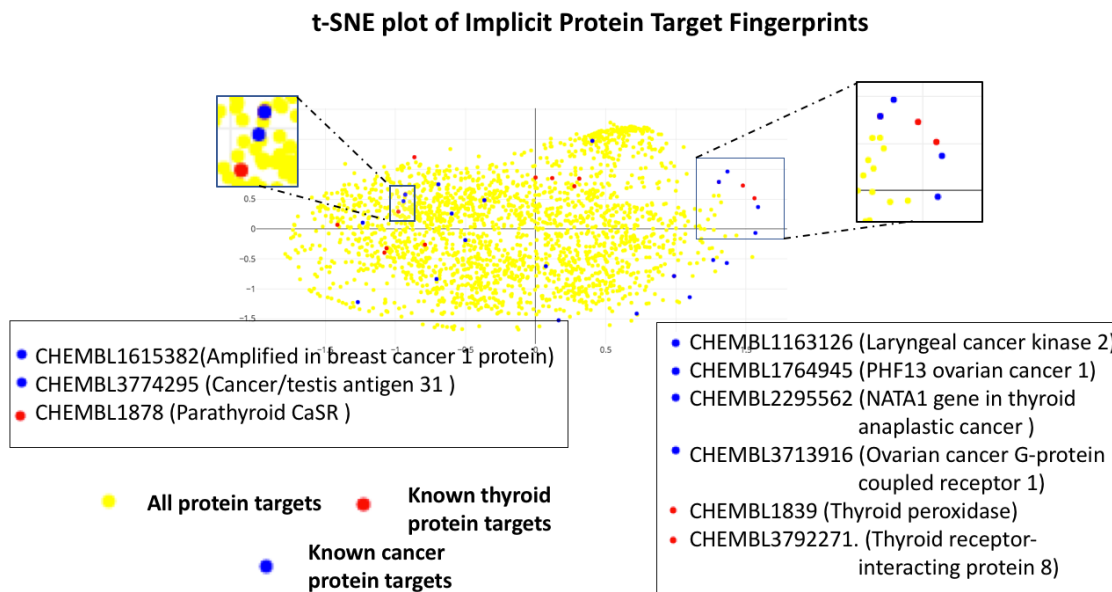


Figure 5.1: Implicit Target Fingerprints : t-SNE plot reducing the dimensions of the 50-dimensional implicit fingerprints into two dimensions for a subset of targets in the ChEMBL database, highlighting known cancer and thyroid related protein targets. The method successfully clusters related targets close to each other, as indicated in the zoomed version of the largest cluster of cancer targets. Interestingly the clusters also contain other targets which are found to have expressions with known cancer/thyroid targets as in the example of Vascular Endothelial Growth Factor Receptor-2 Expression in breast cancer 1 protein.

of high dimensional datasets by minimizing the Kullback-Liebler divergence of distributions in the higher and lower dimensional space. Figure 5.1 illustrates the distribution of the protein targets when mapped into a 2-dimensional space, with the cancer related target proteins highlighted in blue, and thyroid related in red. Interestingly, the graph demonstrates the presence of three potential clusters of known cancer related targets appearing close to each other. Similarly three potential clusters of thyroid related targets also appear close together. These cancer-related targets are visually separated from other biological targets in the latent space. The clusters also contain other targets which are found to have expressions with known cancer/thyroid targets. For example, consider the target clusters containing Vascular Endothelial Growth Factor Receptor-2 Expression (CHEMBL1878) with breast cancer 1 protein (CHEMBL1615382) or Thyroid peroxidase (CHEMBL1839) with PHD13 ovarian cancer 1 (CHEMBL1764945). In the 2D implicit target fingerprints space, the close distances among



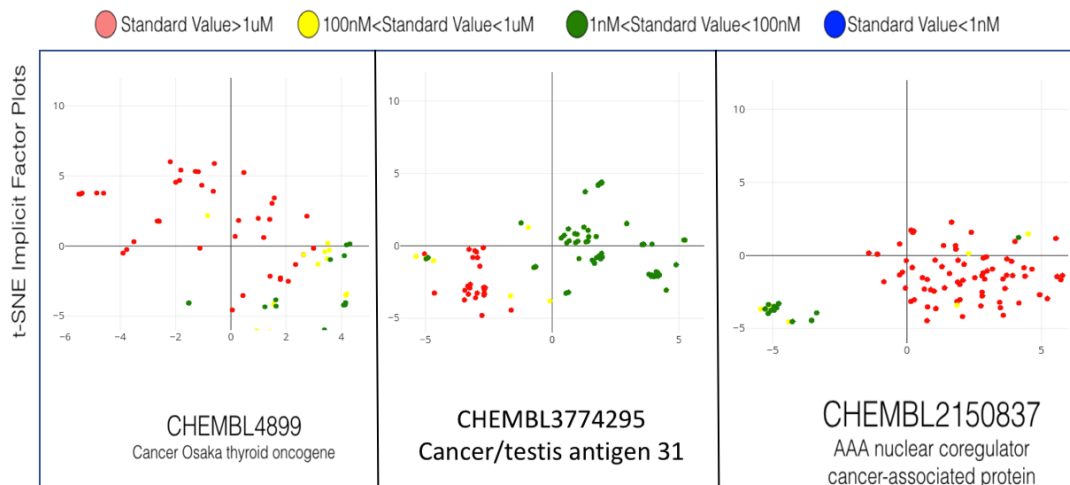


Figure 5.2: t-SNE Plots of Implicit Ligand Fingerprints : Plots for three cancer targets are shown where each point represents a compound assayed from the ChEMBL database. The concentration results of the assays are color-coded. t-SNE plots of the 50-dimensional implicit representations, reduced to 2 dimensions preserving distance.

these targets corresponds to known observations of similarity from prior research [62, 63] . This also engenders a potential for identifying other unexplored relations between the protein targets that appear close to each other in the implicit fingerprint space.

Similar to the biological targets, the the ligands can also be mapped in this latent space, as illustrated next.

To help intuit the compound-protein binding affinity prediction capabilities of the implicit latent space, we randomly selected three cancer related targets from the ChEMBL database. We selected target with IDs CHEMBL4899, CHEMBL3774295 and CHEMBL2150837 as shown in Figure 5.2. Again, the 50-dimensional implicit fingerprints of the compounds are reduced into a 2-dimensional space using the aforementioned t-SNE method. We visualize all compounds with known assays for the three selected cancer related protein targets. The compounds are color-coded on the basis of their standardized concentration levels in the assays, where a decreasing concentration level indicates stronger binding affinity. For the t-SNE plots, the ideal result would be perfect clustering for each concentration level.

The implicit fingerprints from the figure demonstrate a very clear separation between the compounds based on the concentration levels required to trigger binding affinities with the respective targets. This visual separation is striking for assays with excellent binding affinity (standard value below 100nM), indicating that the implicit representation is excellent in its ability to capture properties of similar compounds using Euclidean distance.

From the above evaluations, we conclude that the baseline modeling methods traditionally can be enhanced by the use of *Implicit Target and Ligand Fingerprints*. The models generated by the *Implicit Fingerprints* have better predictive power than their explicit counterparts when the number of known assays for training is less than 100. For the remaining targets, *Implicit Fingerprints* performs about equally to explicit molecular fingerprinting up to about 500 assays. When the number of assays is above 500, traditional methods have a slight, but significant advantage over collaborative filtering.

## 5.2 Limitations

While we conclude our research objective 1 that implicit-descriptor modeling is a promising method for virtual screening, we also point out that our analysis was completed on a large subset of the ChEMBL database, Version 23. That said, we also highlight some of the limitations of the current work and opportunities within the domain. We acknowledge further studies need to be conducted on the cumulative predictive powers of the traditional ligand fingerprinting techniques and the implicit fingerprints generated from collaborative filtering in order to understand if the implicit fingerprints are consistent with other groups of targets and ligands. We also note that a limitation of our approach is that we require ligands to be assayed upon more than one target in order to evaluate them. That is, a ligand must be paired with a target in the training set and paired with another target in the testing set for our method to be able to evaluate binding affinities. We also point out that our implementation of **across target training mode** is not widely used in the cheminformatics community. We only investigated one method of fingerprinting targets using the ProFeat feature generation tool. Other target methods could provide superior performance. Moreover, we note that the Random Forest method trained using combined target and ligand fingerprinting through RDKit was typically a strong performer compared to traditional

per target training (which is most often employed in the virtual screening literature). While the performance of Random Forests was inferior to collaborative filtering, this result does warrant further investigation into techniques for featurizing protein targets. Such an investigation may prove to uncover models that can perform superior to collaborative filtering methods.

### **5.3 Conclusion of Research Objective 1**

We have thus far discussed our research pertaining to our research objective 1 : Implicit Fingerprint based virtual screening by means of collaborative filtering. In the current chapter and preceding chapters, we discussed the approach, algorithms, results and concluded with our novel contribution to this domain, namely Implicit Ligand Fingerprint. In the subsequent chapters, we describe the research methods into our Research Objective 2: Deeplearning based drug design using Implicit Ligand fingerprints.

## Chapter 6

### Introduction to CFGenNet, Collaborative Filtering based Generative Networks

#### Overview

In the previous chapters, we have shown that implicit fingerprints capture ligands and proteins in a shared latent space, typically for the purposes of virtual screening with collaborative filtering models applied on known bio activity data. We extend this further in our research objective 2, where we extend these implicit fingerprints/descriptors using deep learning techniques to translate latent descriptors into discrete representations of molecules (SMILES), without explicitly optimizing for chemical properties . This allows the design of new compounds based upon the latent representation of nearby proteins, thereby encoding drug-like properties including binding affinities to known proteins. The implicit descriptor method does not require any fingerprint similarity search, which makes the method free of any bias arising from the empirical nature of the fingerprint models [43]. We evaluate the properties of the novel drugs generated by CFGenNet using physical properties of drug-like molecules and chemical complexity. Additionally, we analyze the reliability of the biological activity of the new compounds generated using this method by employing models of protein ligand interaction, which assists in assessing the potential binding affinity of the designed compounds. We find that the generated compounds exhibit properties of chemically feasible compounds and are likely to be excellent binders to known proteins. Furthermore, we also analyze the diversity of compounds created using the Tanimoto distance and conclude that there is a wide diversity in the generated compounds. The rest of this chapter and the subsequent chapters will cover the methods investigated and the results obtained. It is to be noted that some portions of the subsequent sections have already appeared our publication [1] for which I am the first author.

## 6.1 Decoders to generate SMILES code

The implicit fingerprints from collaborative filtering is an implicit mathematical representation that allows for a more accurate characterization of the drug compound and protein target in the same numeric latent space, thus narrowing the model-associated bias down to that of the assay, i.e., real clinical (albeit in vitro) environment. [64]. Additionally, to facilitate the ability to discover the physical structure of new compounds, in this work we propose decoding methods that map from the implicit representation of the candidate compounds to their physical structure. We believe this expanded capacity of the fingerprinting model will have significant impact on virtual screening and, consequently, drug discovery, as it will render drug discovery less dependent on costly clinical facilities and services. Moreover, the representation will provide new methods for creating and testing candidate drug compounds.

### 6.1.1 Related Works

The application of deep learning into ChemInformatics has shown tremendous impact in the both replacing traditional ML methods such as QSAR modeling and in the space of de-novo drug design using generative networks. The generative methods of de-novo drug design do not rely on predefined rules of structure generation but generalize the probability of generation process using databases of known structures. These methods rely on sampling within the learned probability distributions. Several recent works in generative drug discovery have borrowed key principles and architectures from the field of image synthesis and natural language processing and successfully applied it to the the domain of de novo drug design. Recurrent Neural and their variants have in the recent past been the mainstays of NLP based applications. The sequential nature of the SMILES code make RNNs a natural choice for adoption in the field of de novo drug design [65]. Additionally, reinforcement learning has also been popular, with special mention of REINVENT [32], which combines RNNs with reinforcement learning. [66] demonstrated that RNNs trained with a set of molecules represented as SMILES strings is able to generate a much bigger chemical space than the training set.

Several recent works have investigated the use of neural embedding on compound structure representations such as SMILES codes, showing this embedding is effective for exploring

the chemical properties [36] and generating novel compounds. The Variational Autoencoder architecture [67] has also shown to generate novel chemicals using courtesy, the latent encoding generated in VAEs. [2] refer to these embedded fingerprints as implicit representations. Another variation was introduced with the Randomized SMILES. [68] leveraged the conditional variational autoencoders by controlling multiple molecular properties simultaneously and imposing them on a latent space. They demonstrated the abilities to generate novel chemical constrained to 5 specific properties. [69] introduced Grammar Variational Autoencoders, where they married the concepts of parse trees from context free grammar with VAEs to mitigate the number of invalid SMILES strings generated from the generative models. [42] introduced a concept of Junction tree Autoencoders, which constructs molecular graphs in two phases, by first generating a tree structured scaffold over chemical substructures, and then combining them into a molecule with a graph message passing network. This approach allows us to incrementally expand molecules while maintaining chemical validity at every step. With their novel method, [42] reported improvements over the state of the art methods at the time of their writing. Additionally ArusPose et al [70] demonstrated the advantages of using different representations of SMILES at every stage of training as opposed to using a unique SMILE representation for each molecule. They successfully demonstrated that the quality of the chemical space generated was much higher with this approach. Besides the SMILES representations, the graph based representations of molecules have also been used in the domain successfully. [71] utilized the graph neural networks to express probabilistic dependencies among a graph’s nodes and edges to learn distributions over any arbitrary graph. Compared to baselines that do not use graph-structured representations, they were able to produce models that often performed far better. [41] proposed a framework based on a type of sequential graph generators that do not use atom level recurrent units. They successfully demonstrated abilities of their approach to generate compounds containing a given scaffold, compounds with specific drug-likeness and synthetic accessibility requirements.

In addition to Variational Autoencoders, Generative adversarial neural (GAN) networks [72] are also a very popular architecture for generating realistic data. A GAN has two components, a generator and a discriminator, that compete against each other during training. The generator generates artificial data and the discriminator attempts to distinguish it from real

data. The model is trained until the discriminator is unable to distinguish the artificial data from the real data [72]. ORGAN [39] and ORGANIC [73] were some of the first demonstrations of the application of GANs for drug discovery. [74] and [75] further combined the concepts of GANs and Reinforcement learning to the application of de novo drug design. [40] introduced LatentGAN, which combined VAEs and GANs for de novo molecular design. They successfully demonstrated the abilities of Latent GAN to generate novel compounds in two scenarios: one to generate random drug-like compounds and another to generate target-biased compounds. Additionally, IBM also demonstrated the viability of deep generative models for antimicrobial discovery [76]. They leveraged classifiers trained on an informative latent space of molecules modelled using a deep generative autoencoder, and screened the generated molecules using deep-learning classifiers along with physicochemical features derived from high-throughput molecular dynamics simulations. With the power of generative learning methods combined with molecular dynamics simulations, they were able to synthesize and experimentally test 28 candidate antimicrobial peptides in record short time frame.

While the list of interesting research in the field of deep learning for de novo drug design is constantly growing, we also note that [77] provides a comprehensive review of research in the advances for the generation of novel molecules with desired properties with a focus on the applications of GANs, RL, and related techniques.

However, to the best of our knowledge all the prior methods, including their variants [38–42, 78] work upon a known representation of the ligands, either the raw SMILES textual representation or graph based representations and are therefore limited in their ability to discern more complicated relationships encoded by fingerprints. The common theme in these techniques is to provide as input to the deep learning model, the molecules only to produce the same or similar molecules as output. The continuous vector representations of the input molecules in the intermediate layers produce a larger chemical property space, which is then sampled to produce novel molecules.

In this work, we design and train deep learning methods that leverage the implicit compound fingerprints obtained from collaborative filtering based on the past bioactivity/assay data to map back to the physical structure of compounds. The implicit encoding of com-

pounds are a continuous vector-valued representation and thus lend itself to the use of continuous optimization to generate novel compounds. We further assess the properties of the novel ligands generated in terms of the drug-like physical properties of molecules, chemical complexity and biological activity. We observed that our compounds exhibit properties similar to the known ligands even though CFGenNet does not explicitly train the neural network for optimizing specific properties. Additionally, we compare our work to the prior work of [37] on a set of chemical compounds with known binding affinities to cancer targets from the ChEMBL23 database [79]. This comparative analysis investigates not only the potential binding affinity of the generated compounds to selected protein targets, but also the diversity of compounds generated. We provide evidence that our method is superior in both binding affinity and compound diversity. In the rest of this chapter, we describe the details of the neural network based decoder that attempts to translate a given ligand’s implicit fingerprints into its corresponding SMILES representation.

## 6.2 Neural Network Architecture

Our method to generate novel ligands is comprised of two steps. (1) We generate implicit ligand and protein fingerprints using collaborative filtering and (2) train a neural network to generate the SMILES string from the implicit representation (i.e., a decoder that can map to a conventional representation from the implicit space). The first step involves generating the implicit fingerprints using known assays by applying the collaborative filtering algorithm [43]. This step yields the implicit fingerprint representations for both ligands and protein targets, as described above. The implicit fingerprints are continuous vectors that represent a point in 50 dimensional space.

The Implicit fingerprints of the ligands are then fed into a Gated Recurrent Unit (GRU) [80] neural network to map the corresponding SMILES string encoding. The neural network is trained to minimize the error in reproducing the relevant SMILES string for each input implicit fingerprints of the ligands. The key aspect of the neural network is to learn the function to map the fixed-length continuous vector representation to the SMILES string. This architecture is illustrated in Figure 6.1. Additional details of the neural network design are discussed in subsequent sections.



As with other methodologies utilizing generative deep learning algorithms, [37] the neural network should ensure that the points in the latent space decode to valid SMILES strings. In order to avoid the latent space from being sparse and resolve to large “dead-areas” (areas in the space that are never trained to decode from and therefore behave unpredictably), we performed input data augmentation. The data augmentation involved adding randomness to the input layer of the neural network (i.e., adding random perturbations to the implicit vector). The data augmentation incentivizes the decoder to more fully represent the areas in the implicit latent space of the ligands, such that they can successfully resolve to the corresponding SMILES string. The intuition is that adding noise to the encoded molecules forces the decoder to learn how to decode a wider variety of latent points and find more robust representations. This approach follows the intuitions made popular by the variational auto-encoders (VAEs) [81] by Bowman et al. The VAEs, instead of decoding from a single point in the latent space, sample from a location centered around the mean value and with spread corresponding to the standard deviation, before decoding. This ensures that a sample from anywhere in the area is treated similar to the original input. Even so, there are differences between the VAE approach and ours. In CFGenNet, the latent space is fixed from the collaborative filtering; it is not trainable like in the VAE. Importantly, this means that the sampling incentivizes the decoder to reconstruct similar SMILES scores from given set of similar points. It does not incentivize the collaborative filtering algorithm to change its implicit representation.

The sequential nature of the output SMILES string required us to consider neural network architectures that are adept at handling such data. The application of neural network architectures such as recurrent neural networks and their enhanced variations such as gated recurrent neural networks (GRUs) for problems involving sequential data such as speech recognition, language translation have been very successful. [80, 82–84] The GRU neural networks, with their innate abilities of learning long-term dependencies in sequences, are especially useful for handling SMILES strings.

## Deep learning based Ligand design using Implicit Fingerprints

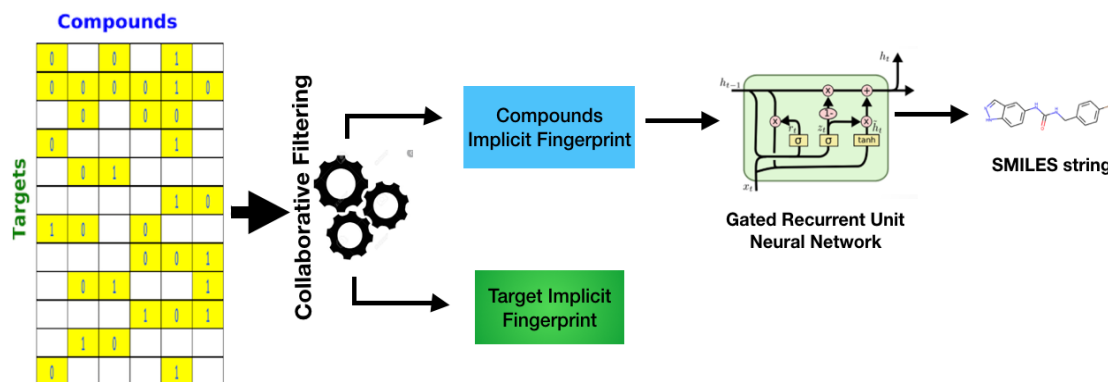


Figure 6.1: Deep learning based Ligand Design using Implicit Fingerprints from Collaborative Filtering - Architecture

### 6.3 Dataset Description

Our method involves translating the implicit ligand fingerprints into its corresponding SMILES string. The implicit fingerprints, however are derived from the ligand - target bioactivity data from the ChEMBL database (Version 23). The bioactivity data, keeping in line with previous studies [85, 86] was focused only on human targets. We restricted bioactivities to three types of binding affinities. The included half maximal inhibitory concentration  $IC_{50}$ , maximal effective concentration ( $EC_{50}$ ), and inhibitory constant( $k_i$ ). Following the precedence with previous works [43, 85, 86], we converted the data into the binary active-inactive using the following conversation thresholds: lesser than 100 nM for “actives” and greater than 1000 nM as “inactives.” Furthermore, in order to be consistent with research objective 1, we retained only ligands which have at least two prior assays. This resulted in a bioactivity matrix of size 241,260 (ligands) by 2,739 (targets). The bioactivity matrix was subjected to the collaborative filtering method as described in research objective 1. The resultant implicit fingerprints were then used as inputs to our deep learning model, with the goal to produce the respective canonical SMILES string as the output. Figure 6.2 illustrates

the data distribution of the number of ligands against the known number of prior assays and known number of prior assays with known positive affinities. As evident from the plots, close to 50% of the ligands have only 2 prior assays. Additionally close to 62% of the ligands have only 1 prior assay with positive affinity. We also wish to note that the number of ligands ( 241k) used to model the deep learning model is comparable to previous works [78].

Considering that CFGenNet relies on the prior assay history to determine the implicit ligand fingerprints, having more numerous examples of prior assays for each ligand may also result in better quality implicit fingerprints. This statement is further evidenced by results from the next section: (1) the ability of the decoder to accurately translate implicit fingerprints into the corresponding SMILES and (2) the abilities of the ligands to yield more novel ligands are both influenced by the number of available assays per ligand, as described next.

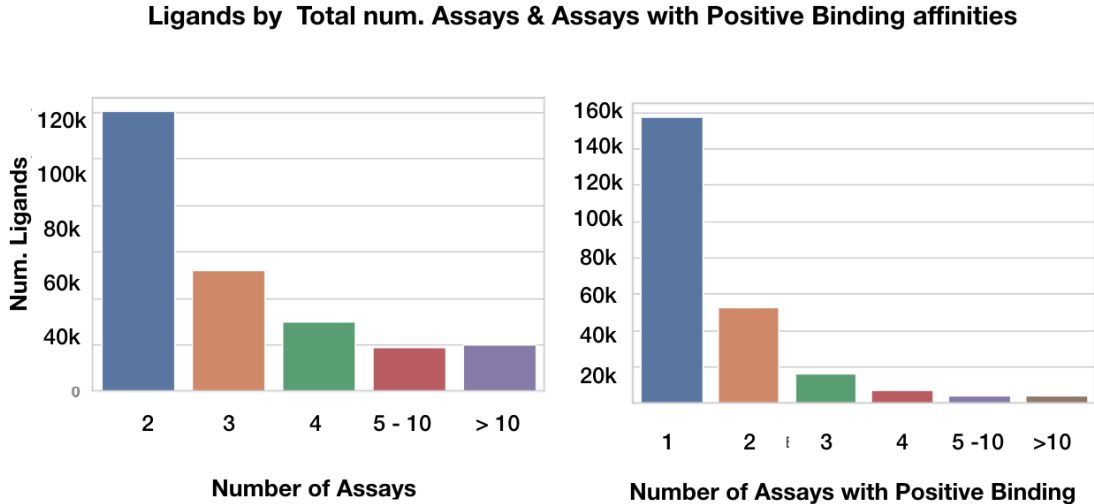


Figure 6.2: Data Distribution : The first figure illustrates the number of molecules against number of assays, binned at specified values or ranges. Close to 50% of the molecules have only 2 prior assays. The second figure illustrates the number of ligands against the number of known assays with positive affinities. It can be observed that 62% of the ligands with only one assay with positive binding affinity.

## 6.4 Construction of the Neural Network

In CFGenNet we use the implicit ligand fingerprints obtained from the prior assay information (collaborative filtering) as inputs to a deep learning model, with the objective of producing the corresponding canonical SMILES representation as the output. This implicit representation can have a number of advantages because it is based solely on the observed behavior of the compound, rather than inherent measures of physical properties. Thus, formulating a decoding procedure from this implicit representation may have distinct advantages over previous methods. The implicit fingerprint, because it is a continuous vector of fixed length (50), also lends itself well to statistical sampling with simple procedures. We employ data augmentation of the input vector by employing a vector of means  $\mu$  and another vector of standard deviations,  $\sigma$ . The input vector (implicit fingerprint vector) serves as the vector of means, which is then added to another vector, which is a random normal distribution centered at 0 with standard deviation  $\sigma$ , to yield a statistically sampled point around the implicit fingerprint. The stochastic sampling process ensures that the actual vector will vary on every single iteration due to sampling, while keeping the mean and standard deviations the same. Intuitively, the mean vector controls where the implicit fingerprint of a ligand is centered around, while the standard deviation controls the “area,” how much from the mean the encoding can vary. The decoder, hence learns that not only is a single point in latent space referring to ligand, but all nearby points refer to the same ligand. The decoder is exposed to a range of variations of the encoding of the same input during training. This process is illustrated in the decoder architecture, as shown in figure 6.3. The approach adopted here is similar to the data augmentation employed with Variational auto-encoders [67].

### 6.4.1 Gated Recurrent Neural Networks

In order to generate the SMILES string from the implicit fingerprint, we are motivated to use recurrent neural networks (RNNs) because of their success in modeling sequential data such as natural language. The SMILES strings lend themselves well to this model considering the sequential nature of the notation. Each unit in the RNN attempts to capture state information of the sequence by transforming all the elements that appeared before it. It does so by encapsulating this information in a hidden state vector, that is passed from

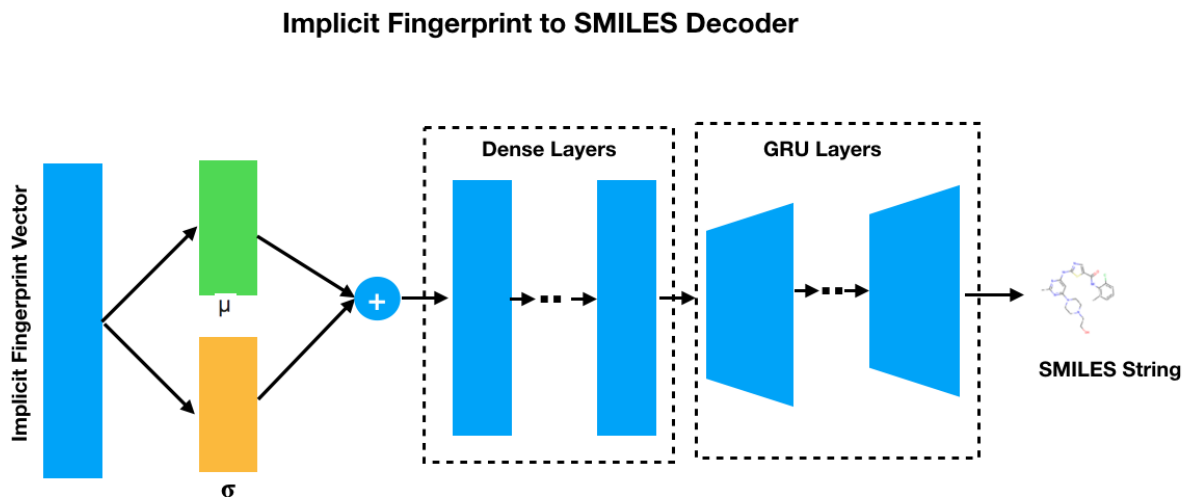


Figure 6.3: Implicit Fingerprints to SMILES Decoder : The deeplearning network learns ligand representations by employing data augmentation technique at the input layer. The continuous representation obtained is then fed into a series of dense layers followed by a Gated Recurrent Unit Neural Network to obtain the corresponding SMILES string.

one unit back into itself, recurrently. The hidden state  $h^t$  of the RNNs can be represented as

$$h^t = \tanh(x^t w^{xh} + w^{hh} h^{t-1})$$

where  $x^t$  represents the input at timestep  $t$ ,  $w^{xh}$  represents the weight from input node to the hidden node,  $w^{hh}$  represents the weight on the feedback loop from the hidden node to itself, and  $h^{t-1}$  represents the previous hidden state. As evident from the equation, the hidden states from the earlier time steps get diluted over long sequences. This problem gets compounded with SMILES considering the long term dependencies (such as matching brackets etc.) that need to be maintained in order to resolve to a valid chemical compound. The Gated Recurrent Neural Network attempts to address this problem by introducing two gates called the “update” gate and a “reset” gate along with a memory which governs how much of the previous state is retained. Each of these units (update gates, reset gate, and memory) have their own trainable weights. The Update Gate at each unit decides the amount

of new information to be added to the hidden states. The reset gate determines the past information to be forgotten or retained at each unit.

$$r = \sigma(x^t w^{xr} + h^{t-1} w^{hr})$$

where  $\sigma$  represents a logistic or sigmoid function. These sigmoid values of the reset gate range from 0 to 1 and determine how much of the previous hidden node value is retained. A value of  $r = 0$  implies the none of the previous node value is retained and a  $r = 1$  ensures the entirety of the previous node is retained. This memory  $m$ , can be signified by the following equation:

$$m = \tanh(x^t w^{xm} + (r \odot h^{t-1}) w^{hm})$$

where  $\odot$  represents the Hadamard (or element-wise) multiplication of two vectors. Additionally the update gate is governed by the following equation:

$$u = \sigma(x^t w^{xu} + h^{t-1} w^{hu})$$

The update gate, with values ranging from 0 to 1, determines if the new hidden state should use the previous value or the new value. Tying all these together, hidden state is governed by the equation:

$$h^t = u \odot m + (1 - u) \odot h^{t-1}$$

Intuitively, the GRUs are better suited than the RNNS to our problem considering the long term dependencies between symbols that must be maintained in the SMILES string. The Ring structures, for example, are represented by matching numeric symbols typically separated by two or more atoms within the SMILES string. The neural network should be able to remember these long term dependencies for effectively decoding to a valid SMILES string. Figure 6.3 illustrates multiple GRU layers that make up the decoder in order to effectively map to the SMILES code. These “layers” shown in the figure are visualized compactly—that is, they are actually two stacked sequences of GRU nodes .

We performed extensive training and validation of the vanilla RNN and GRU models with the continuous implicit fingerprint vector as the input and the one hot encoded SMILES string as the output. We measured the outcome of the models by evaluating the categorical cross entropy loss and the accuracy. As a part of training, we explored a variety of architecture options with respect to the depth and the width of the deep learning models. We also trained with different composition of the training sets based on ligands with varying counts of past assay data. Across all training iterations, we noticed that the GRU based model performed better with lower cross entropy and higher accuracy. We also noticed that the training loss converged faster compared to the validation loss. Our architecture comprised of a series of dense layers followed which consume the 50 vector wide implicit fingerprint representation of the ligands, followed by the GRU layers returning sequential information to map to the SMILES representations. The exact makeup of the deeplearning architecture with the trainable parameters are provided in the supporting information.

Figure 6.4 illustrates the performance of the 3 different models trained with different datasets. The first dataset comprised of all the 241K ligands from the dataset. We additionally trained with training set comprised only of ligands with atleast 1 positive binding affinity (121k ligands) and another iteration comprising of ligands with atleast 5 positive binding affinities (8.5k ligands). As evident from the figure, the training and the corresponding validation loss was lower when trained with the filtered data sets as opposed to the entire population of 241k ligands. This can be attributed to the fact that the implicit fingerprints of the ligands that exhibited positive binding affinities in prior assays tend to encode more information on the ligands, and hence decodable into the explicit SMILES representations. While this implies that approximately half the ligands in our dataset do not resolve back to its corresponding SMILES representation, it does not however dent the utility of CFGenNet. This is due to the fact CFGenNet is able to resolve the implicit fingerprints of the ligands which have demonstrated bioactivities in the past, and hence such ligands are more desirable to be used as anchor ligands from which to generate novel ligands.

The detailed architecture of the neural network is illustrated in figure 6.5. The number of parameters at each layer is illustrated in the figure 6.6.

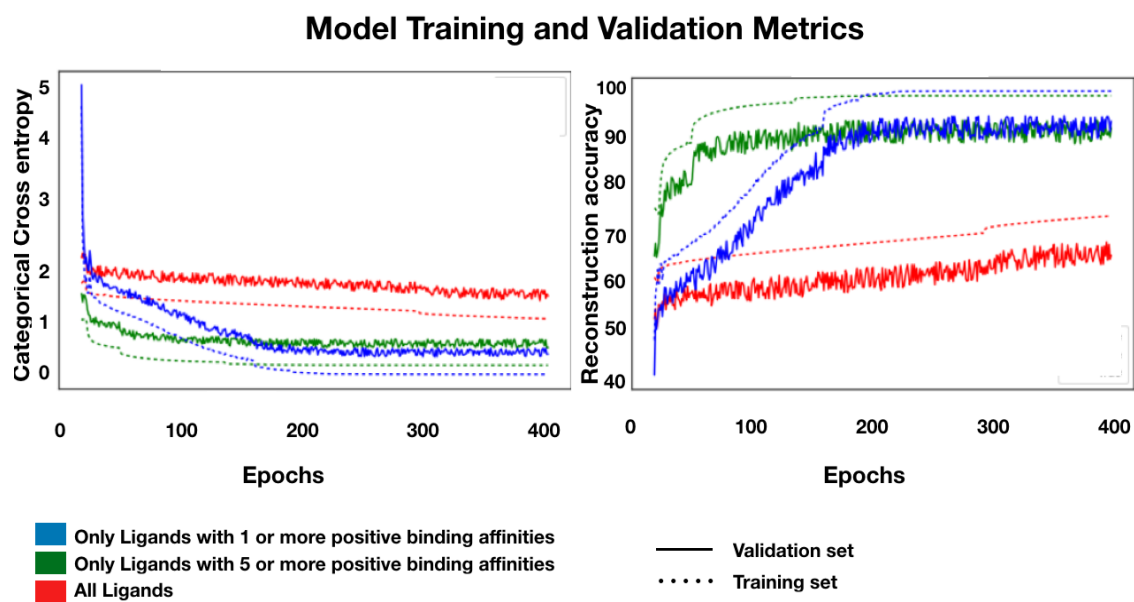


Figure 6.4: Train and Validation Losses of the Neural Network : Training and validation losses across multiple runs of the neural network.



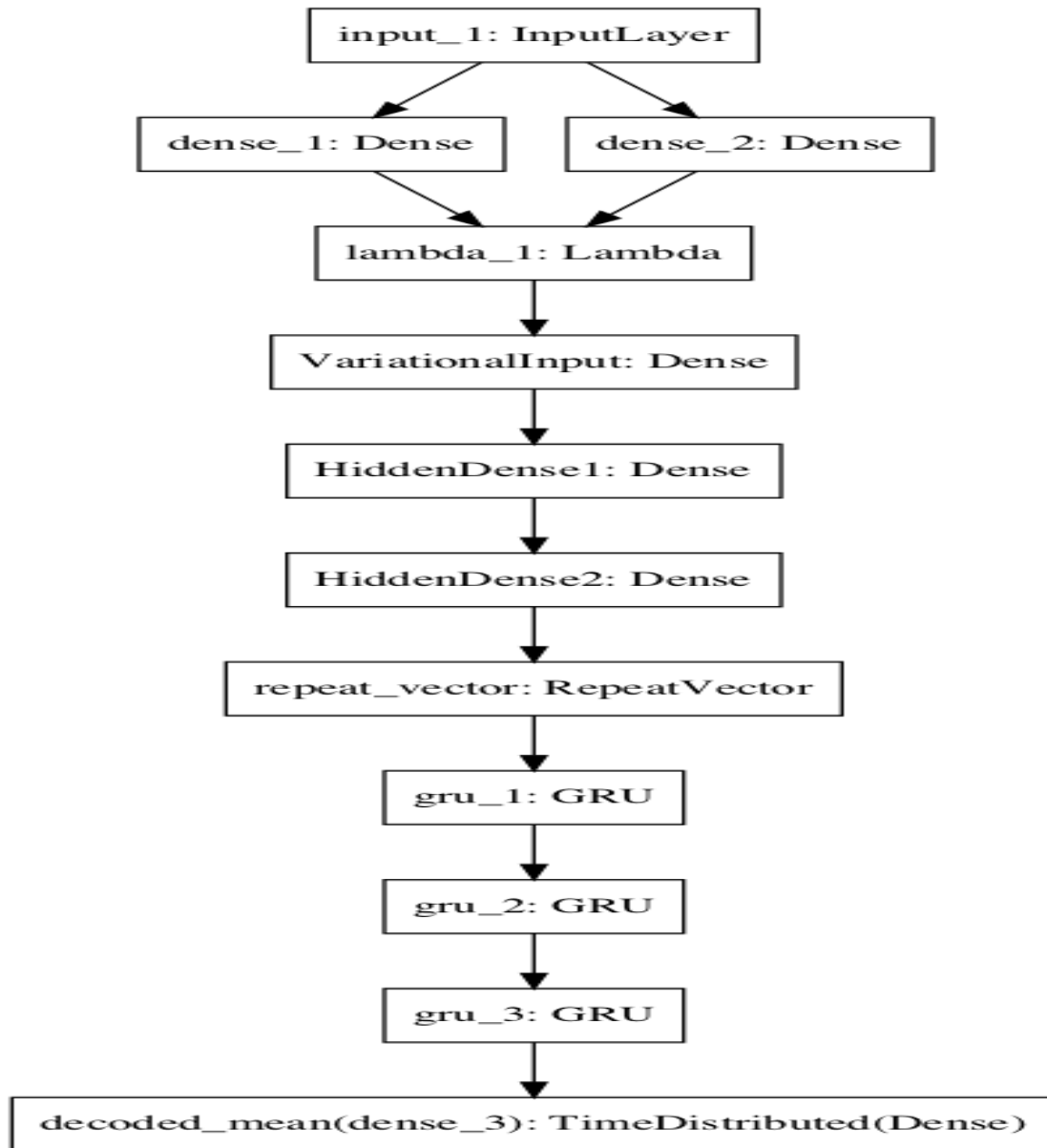


Figure 6.5: Neural network architecture of the deep learning model.

Trainable parameters of the Deep learning architecture

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	(None, 50)	0	
dense_1 (Dense)	(None, 50)	2550	input_1[0][0]
dense_2 (Dense)	(None, 50)	2550	input_1[0][0]
lambda_1 (Lambda)	(None, 50)	0	dense_1[0][0] dense_2[0][0]
VariationalInput (Dense)	(None, 50)	2550	lambda_1[0][0]
HiddenDense1 (Dense)	(None, 80)	4080	VariationalInput[0][0]
HiddenDense2 (Dense)	(None, 120)	9720	HiddenDense1[0][0]
repeat_vector (RepeatVector)	(None, 120, 120)	0	HiddenDense2[0][0]
gru_1 (GRU)	(None, 120, 501)	934866	repeat_vector[0][0]
gru_2 (GRU)	(None, 120, 501)	1507509	gru_1[0][0]
gru_3 (GRU)	(None, 120, 501)	1507509	gru_2[0][0]
decoded_mean (TimeDistributed)	(None, 120, 58)	29116	gru_3[0][0]
Total params: 4,000,450			
Trainable params: 4,000,450			
Non-trainable params: 0			

Figure 6.6: Layers and parameters trained in the deep learning model.

## Chapter 7

### Syntactic & Novelty Analysis of Generated Ligands

#### 7.1 Introduction

In this chapter and the subsequent chapter, we present the results of our methods in the context of its abilities to generate valid and novel molecules from the implicit latent space. The ligands are validated in multiple folds. It is to be noted that some portions of the subsequent sections have already appeared our publication [1] for which I am the first author.

- Syntactic Analysis of generated molecules: We validate if the generated molecules are syntactically correct and can indeed resolve to a valid molecule. The molecules (their corresponding SMILES strings) are validated using the RDKit library [16]. The details are further described in the subsequent sections of this chapter.
- Novelty of the generated molecules : We validate if the generated molecules are novel. We employ multiple criteria to assess novelty , as further described in this chapter.
- Semantic Analysis of generated molecules : We validate if the generated molecules are meaningful and useful. We evaluate the properties of the molecules and further validate its usefulness to be synthesized into a valid drug. The details are further described in the subsequent sections of this chapter.

We wish to note that large parts of the subsequent sections have already been appeared in a journal paper [1].

#### 7.2 Syntactic Analysis of Generated Ligands

In this section, we discuss the outcomes of our method in the context of 5,000 randomly selected ligands from the validation set. Additionally we also present the outcomes of a scaf-

fold analysis from the potentially novel ligands generated from ligands with known affinities to cancer targets from the ChEMBL23 database. To further analyze the practical applicability of CFGenNet, the resulting ligands, specifically from approved cancer related drugs were further evaluated for their viability to be valid compounds with enhanced biological activities. The complete list of ligands is made available as a part of the supporting information of our journal paper [1](section 0.4).

Our method samples around the implicit latent space of the known ligands, or “anchor ligands” to generate (potentially novel) compounds. In our testing, we randomly sampled 100 points across the 50 dimensions in the implicit space around our anchor ligands. Each point was then processed through our neural network to obtain the corresponding SMILES string. The SMILES string was then validated using the RDKit library. This process is discussed in more detail in the next chapter on the syntactic analysis of generated ligands.

We ran the aforementioned sampling and validation exercise on the 5,000 ligands (henceforth referred to as anchor ligands). A total of 4,632 out of the 5,000 (92.64%) anchor ligands resolved to at least one valid ligand, although not all resolved ligands were (potentially) novel. As mentioned earlier, 100 points are randomly sampled for each anchor ligand. Depending on the information encoded in the continuous implicit vector space, multiple points around a given anchor ligand may resolve to the same ligand. Only those ligands that are generated at least twice, and can be resolved to a valid compound using the RDKit library, are considered to be “valid” generated ligands. The frequency constraint of “at least twice ” is enforced to help ensure that the generated ligand is not generated spuriously.

### 7.3 Novelty analysis of Generated Ligands

The practical applicability of the generative deep learning methods are typically measured by the ability of the methods to generate novel ligands with desirable properties. However, despite the plethora of works in the space, the concept of novelty is loosely defined. Several popular recent works [2, 32, 33] just validate if the generated ligand was already present in the training data set. If not found, the generated ligands are deemed as novel. Alternately, Popova et al [34] assessed novelty by checking for the presence of the generated ligands in the training set of 1.5m ligands from ChEMBL21. Additionally they also searched for the

presence of the generated ligand in the ZINC database for 320M synthetically accessible drug-like molecules. It is to be noted that a difference of even a single atom was deemed as a sufficient condition to term a generated ligand as being novel. With the precedence in the aforementioned prior works serving as a baseline, we adopted a series of multi-fold conditions to assess the potential of the generated ligands to be deemed novel:

- Is the generated ligand already present in the training data set?

A total of 2917 generated from the validation set of 5k anchor ligands were not present in the training set.

- Is there any other known ligand in the 1+B ZINC database with a similarity threshold of .85 with the generated ligands.

We further ran the 2917 ligands against ZINC database to look for the most similar known ligand from 1.3 billion compounds from the ZINC database, with a similarity threshold of .85. The similarity between compounds were measured by the Tanimoto Coefficient (TC) which measures a distance between fingerprints resulting in a score ranging from [0,1] (0 corresponds to least similar and 1 to exactly same). [87] We obtained the TC based on Morgan Fingerprints [88] of radius 512 bits . This resulted in a total of 2759 ligands from the previous step.

- Among the ligands from ZINC database with similarity  $<.85$ , are there differences in the scaffolds and/or the number of functional groups between the anchor and generated ligands.

In order to further verify that the generated ligands were meaningfully different from their closest hits from ZINC database, we further compared the scaffolds and functional groups between the generated ligands and their closest hit from ZINC database. Of the 2759 ligands from previous step, only 322 (12%) of ligands had the same functional groups and only 618 ligands (22%) had similar scaffolds as their closest hit from ZINC database. These numbers are summarized in table 7.1 for easier readability. Additionally, Figure 7.1 plots the distribution of the ligands over the differences in the number of functional groups between each pair of potentially novel ligand and their closest hit

from ZINC database. It is seen that 67% (1831 ligands) of the ligands generated had a difference of at least 2 or more functional groups with their closest hit. The figure also illustrates the TC similarity scores between the potentially novel ligands & their closest ZINC database hits. It is observed that 20% of the potentially novel ligands have similarity of .5 or less.

We wish to note that the references of "novel ligands" in the rest of the sections should be read as being potentially novel and in conjunction with the aforementioned set of conditions.

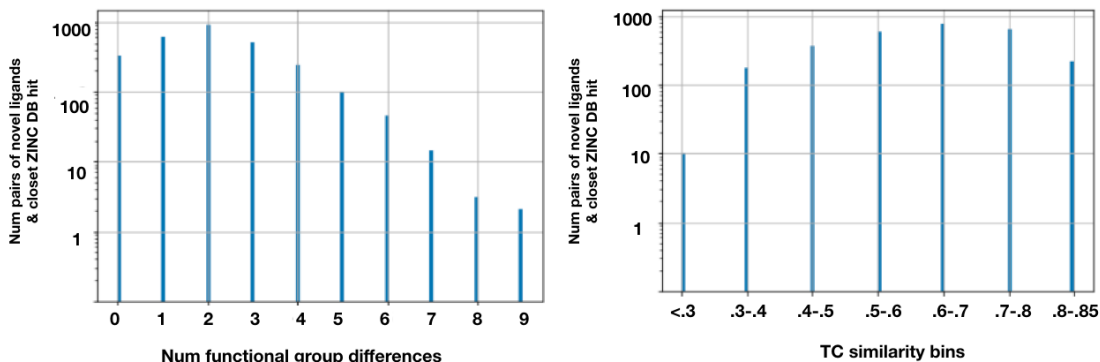


Figure 7.1: Novelty of generated ligands : Distribution of the number potentially novel ligand and their closest hit pairs v/s differences in the functional groups & TC similarity ranges. The figure demonstrates that A degree of novelty could be associated with the generated ligands when compared with the 1.3Billion known ligands from ZINC DB. The differences in the number of functional groups between the anchor and generated ligands range from 0 to 9, with at least 67% ligands with 2 or more differing functional groups. The TC similarity bins help gauge the distribution of the TC similarities between the pairs. It is seen that the lower the similarity, more likely it is for the pair having varying functional groups.

It was also observed that sampling around certain anchor ligands resulted in numerous potentially novel ligands being generated, while sampling around other anchors did not yield *any* novel ligands. To further investigate this phenomenon, we analyzed the abilities of the

Table 7.1: Summary of Generated Ligands

---

Number of anchor ligands from validation set :	5,000
Number of anchor ligands yielding at least 1 valid ligand :	4,632
Number of generated ligands not present in the training data :	2,917
Number of generated ligands with a difference of 2 or more functional groups:	1,831
Number of anchor ligands yielding at least 1 ligand outside training data :	1,332

---

associated anchor ligands to yield potentially novel compounds by grouping according to known prior assays. Figure 7.2 (left) illustrates the relationship between the presence of assays of the anchor ligands and their ability to generate potentially novel ligands. As can be seen, anchors that generated novel ligands tended to have a greater number of known assays. In Figure 7.2 (right), we can also observe that this relationship holds for the number of anchor ligands with positive affinities. Anchors with a more numerous positive assays also tended to generate potentially novel ligands.

This observation is perhaps not surprising considering that the implicit fingerprints are derived from the known assays. This provides evidence that the implicit fingerprints for anchor ligands encode more meaningful information when there are more numerous assays. One potential explanation for this is that the implicit representation can encode many desired properties that are difficult to measure as the number of assays increases, thereby giving a better blueprint for the decoder to generate potentially novel ligands. However, because the implicit representation does not explicitly model chemical or experimental parameters, this hypothesis must be investigated through observation of known ligand properties, as discussed next.

### 7.3.1 Conclusion

In this chapter , we reviewed the methods adopted to validate the syntax and novelty of the molecules generated by CFGenNet. We further evaluated in detail the results from the aforementioned methods. We further evaluated these ligands for their usefulness in being synthesized as drugs. We refer to this as the semantic analysis of the generated results,

### Generation of novel ligands v/s Prior Assay Results

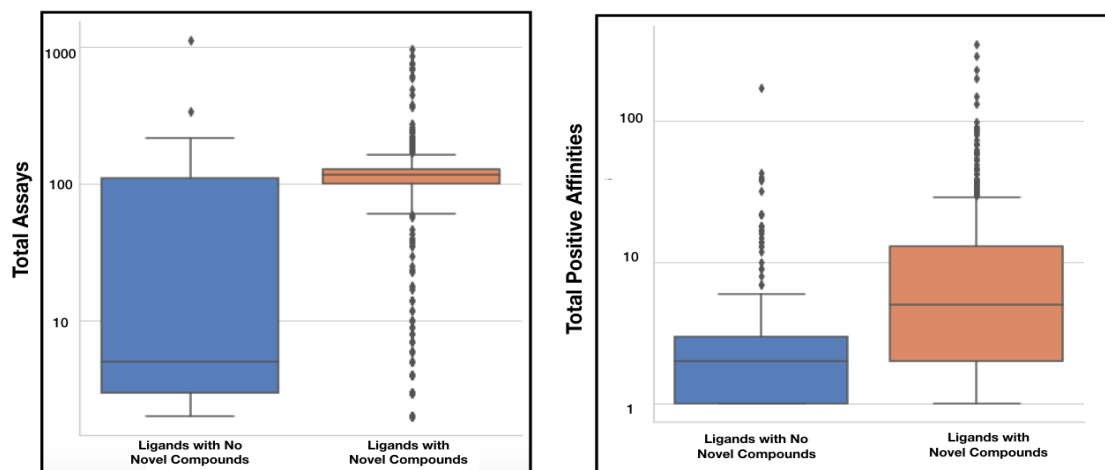


Figure 7.2: Ligand generation v/s total assays : Correlation of the ability to generate potentially novel ligands with prior assays: Box plots show the co-relation between the two sets of anchor ligands - one set from 1 or more potentially novel ligands were generated and the second set which yielded no ligands when sampled in the implicit fingerprint latent space. The first figure visualizes the total number of known assays that exists for each set. The second box-plot visualizes the total number of positive binding affinities already recorded for each assay.

whose details are covered in the next chapter.



## Chapter 8

### Semantic Analysis of Generated Ligands

In this chapter, we provide details of the evaluation conducted to validate the quality of the generated ligands. We term this as the semantic analysis of the generated ligands. As a part of this analysis, we evaluated some of the commonly used physical properties to validate the usefulness of generated ligands from prior works. Additionally, we also evaluated the scaffolds exhibited by the generated ligands along with the abilities of the generated ligands to bind with known protein targets. We wish to note that large parts of the subsequent sections are already published in our journal article (under review) [1] for which I am lead author. It is also to be noted that the analysis of binding affinities, scaffolds, and relative performance comparisons with FDA approved drugs was led by a co-author (Niraj Verma) of the aforementioned paper [1]. However I contributed in the interpretation of these analyses.

#### 8.1 Physical Properties of Generated and Anchor Ligands

In order to explore the similarity of potentially novel and anchor ligands, we evaluated the properties of the compounds using a number of scoring measures. More specifically, we used the quantitative estimation of drug-likeness (QED), n-octanol water partition coefficient (LogP), synthetic accessibility score (SAS), and used the number of benzene rings as an indicator of the chemical complexity. Our approach to ascertain the similarities in the physical properties between the anchor ligands and its corresponding novel ligands considered the following approaches

- Compare the distribution of the populations of property values of the anchor ligands with the distribution of the generated ligands. This comparison is standard practice when evaluating the quality of generated ligands [34, 36].

Table 8.1: Properties of anchor and potentially novel ligands.

		Anchor Ligands	Novel Ligands	t-test
QED	Mean	0.69	0.57	t-stat = .99 p-value = .35
	Stddev	0.20	.22	
LogP	Mean	3.41	3.43	t-stat = 0.33 p-value = .74
	Stddev	1.69	1.95	
Benzene Rings	Mean	3.45	3.12	MW stat = 2.1e7 p-value = 4.76e-18
	Stddev	1.24	1.33	
SAS Scores	Mean	2.67	3.18	t - stat = 21.62 p-value. = 6.7e-99
	Stddev	0.55	0.85	

- Additionally, to investigate similarity of generated ligands with their respective anchor ligand, we evaluated the magnitude of the difference in the values between the ligands for each of the 4 aforementioned properties. A residual value, which is the difference between the property values is calculated for each pair of anchor ligand and its corresponding generated ligand. A mean residual is then obtained for each anchor ligand as described in equation 8.1. The magnitude of the mean residual value was used as a method to determine the deviation of the properties between anchors and its generated ligands.

$$R_m = \frac{\sum_{n=1}^N \sqrt{(p_a - p_n)^2}}{N} \quad (8.1)$$

$R_m$ :mean residual property value for each anchor ligand

$N$ :number unique potentially novel ligands generated for each anchor ligand

$p_a$  : property value (QED,LogP,SAS and NumRings) for the anchor ligand

$p_n$  : property value (QED,LogP,SAS and NumRings) for  $n^{th}$  novel ligand for the corresponding anchor ligand.

The QED ranges between 0 and 1. The ligands with higher value indicate that the molecule is more drug-like. Additionally, the method also claims to capture the abstract

### Property Distributions of Anchor Ligands v/s Generated Ligands

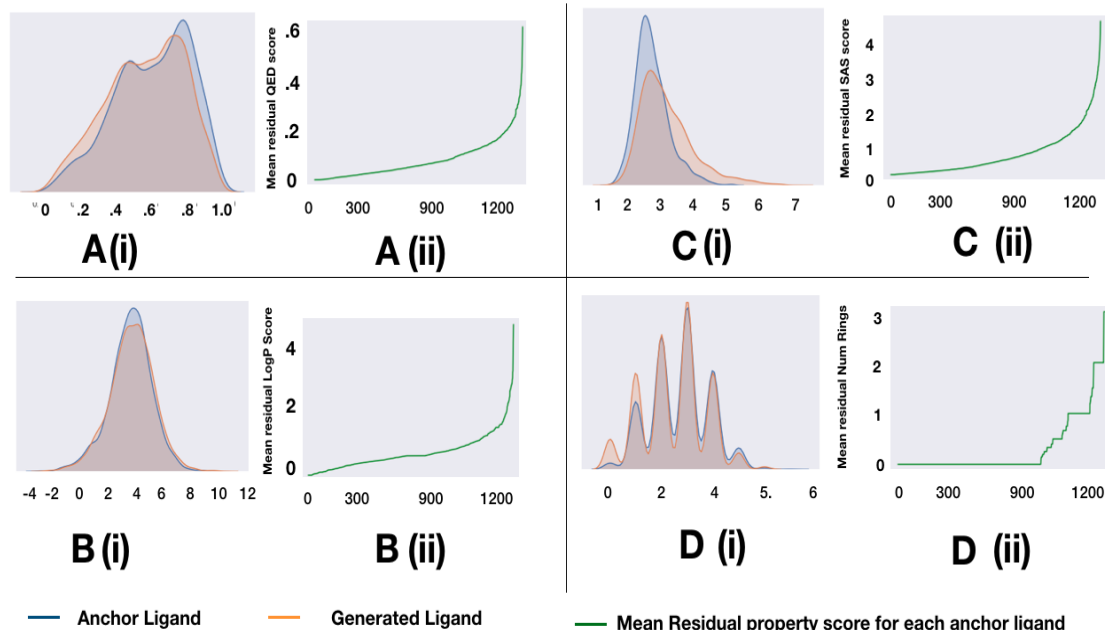


Figure 8.1: Property Distribution between anchor ligands and generated ligands : (A) Quantitative Estimate of DrugLikeness(QED) (B) Partition Coefficient (LogP) (C) Synthetic Accessibility Score (SAS) (D) Number of Benzene rings. The figure demonstrates that the property distributions of the anchor ligands is similar to the potentially novel ligands generated from the corresponding anchors across all 4 properties.

notion of aesthetics in medicinal chemistry [89]. We leveraged the python based RDKit library to determine the QED scores of the generated novel compounds. As illustrated in Table 8.1, the average QED score of the novel ligands was found to be 0.57. Figure 8.1-A(i) illustrates the comparison of the distributions of the QED scores from the potentially novel ligands with their anchors. It can be observed that the two distributions are very similar. The 2 sample student t-test statistic of 0.99 with a p-value of 0.35 also confirms that there exists no statistical difference between the two distributions. Table 8.1 tabulates the mean, standard deviations and t-test scores of all the properties calculated as a part of our experiments. Additionally, figure 8.1-A(ii) illustrates the similarities of the QED scores between the anchor ligands and their respective generated ligands by measuring the mean residual value as described in equation 8.1. It is evident from the plot that a large number of mean residuals are less than .1 units. This indicates the QED scores of close to 80% of

the anchor ligands are within .1 units of their generated novel ligands, and close to 96% of the anchor ligands have QED scores within .2 units of their generated ligands.

The water-octanal partition coefficient (LogP) was an other property used to quantify the physical properties of the potentially novel ligands. LogP describes the propensity of ligand to dissolve in an immiscible biphasic system of lipid (fats, oils, organic solvents) and water [90]. A negative value for logP means the ligand has a higher affinity for the aqueous phase(hydrophilic);when  $\log P = 0$  the ligand is equally partitioned between the lipid and aqueous phases; a positive value for logP denotes a higher concentration in the lipid phase(lipophilic). The potentially novel ligands tended to be more lipophilic with a mean LogP value of 3.43 with a standard deviation of 1.94. Figure 8.1-B(i) illustrates that distributions of LogP scores between the novel and anchor ligands. The two distributions appear to be visually similar and the 2 sample student t-test score of .33 with p-value = .74 also confirms the same. Additionally figure 8.1-B(ii) illustrates the similarities of the LogP scores between the anchor ligands and their respective generated ligands by measuring the mean residual score as described in equation 8.1. It is observed that close to 87% of the anchor ligands have their LogP scores within 1 unit of their generated ligands.

The synthetic accessibility score (SAS), a method that is able to characterize molecule synthetic accessibility as a score between 1 (easy to make) and 10 (very difficult to make) [91] was another property that was evaluated for the potentially novel drugs generated by our method. The mean score was found to be at 3.17 with a standard deviation of .85. While the SAS scores between anchors and their novel ligands appear to be similar visually (8.1-C(i)), the t-statistic score of 21.62 with p-value=6.7e-99 indicate that the two distributions are statistically different. Nevertheless the mean score of 3.17 of the potentially novel ligands indicate that the potentially novel ligands are synthesizable to generate valid drugs. Figure 8.1-C(ii) further compares the individual SAS Scores between the generated ligands and their respective anchor ligands. It is observed that 87.3% anchor ligands have SAS Scores within 1 unit of the generated ligands. This indicates that an overwhelming majority of the anchor ligands share similar SAS Scores with their generated novel counterparts. Additionally the number of Benzene rings was evaluated as a measure of chemical complexity of the potentially novel ligands. Figure 8.1-D(i) demonstrates that the complexities of the

potentially novel drugs are comparable to the complexities of their corresponding anchor ligands. Figure 8.1-D(ii) compares the similarities in the number of benzene rings between the anchor ligands with their respective generated novel ligands. From the figure, it is evident that the distribution of the number of rings do not follow a normal distribution. For this reason, we conducted the Mann Whitney non parametric test [92] to compare the two distributions. The test yielded a statistically significant difference in the 2 distributions . however it was observed that approximately 83% of the anchor ligands had the exact same number of benzene rings as their respective generated novel ligands.

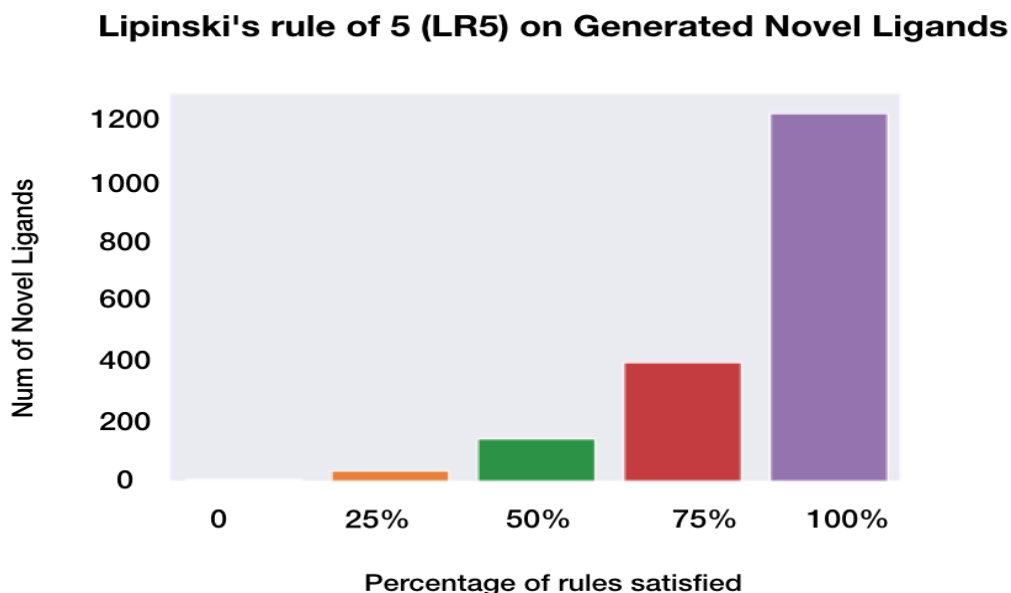


Figure 8.2: Lipinski's Rule of 5 : Lipinski's Rule of 5 valuated on the potentially novel ligands generated from implicit fingerprints. The figure demonstrates that 80% out of the 1,831 potentially novel ligands satisfy 3 or more rules, signifying that the generated ligands have properties to be an effective drug.

We further evaluated the Lipinski's rule of 5 (LR5) for all the generated ligands. [93] The LR5 describes critical properties of a ligand in the human body such as absorption, distribution, metabolism, and excretion. The rule states that a ligand to be effective for

therapeutics should have less than 5 hydrogen bond donors, less than 10 hydrogen bond acceptors, a molecular mass of less than 500 daltons and the LogP less than 5. The LR5 score was computed for all generated ligands based on [94]. We observed that 68% of generated ligands completely satisfies the LR5 rule and 22% of generated potentially novel ligands satisfy at least 3 out of the 4 rules. This is further illustrated in figure 8.2. The percentage of matches to Lipinski’s rule of 5 signifies that the generated ligands have properties to be an effective drug.

Now that it is established that the potentially novel and anchor ligands are likely to have similar and comparable physical properties, we turn our attention to answering whether the novel ligands are also likely to similarly bind to known targets.

## 8.2 Binding Affinity predictions of the potentially novel ligands

The biological activities of the potentially novel ligands were evaluated by inferring their predicted binding affinities with 102 DUD-E protein targets. [95] The DUD-E targets consist of a variety of proteins exhibiting different mechanism of protein-ligand interactions. The relationship of bioactivities within the anchor ligand and generated ligands over the DUD-E targets will highlight the versatility of our model. Thus we used the anchor ligands to test their binding affinities with the DUD-E targets.

In order to validate the similarities of the binding affinity properties of the novel ligands with their respective anchors, the binding affinity scores were determined from SSnet and Smina for the anchor ligands with the 102 DUD-E proteins. Each ligand (anchor and novel ligands) yielded a distribution of binding affinity scores against each target from the set of 102 DUD-E protein targets. The similarities in the binding affinities of the novel and their respective anchor ligands were evaluated by comparing the aforementioned binding affinity distributions. Out of the total 1,332 unique combinations of novel and respective anchor ligands, approximately 84% demonstrated similar binding affinity behaviors. The similarity score or the measure of Intersection over Union (IOU) [96], in this exercise is calculated by evaluating the proportion of DUD-E targets to which both the ligands demonstrate binding or lack of binding. An SSnet score of 0.5 or less is considered lack of binding, and a score greater than 0.5 as binding. Figure 8.3 illustrates this for 1,332 unique pairs of novel ligands and

their anchor ligands. Each data point on the x - axis in figure 8.3 represents a unique anchor-novel ligand combination. The y-axis represents the Intersection over Union score calculated between the 2 distributions of binding affinity scores, the first distribution being binding affinity indicator of anchor ligand with 102 DUD-E proteins and the second distribution, the binding affinity indicator of the novel ligand with 102 DUD-E proteins. The figure further illustrates that a large majority of the anchor-ligand pairs exhibit similar binding affinities. A similar observation was made for Smina shown as Figure S1 by considering ligand similarity based on -7.5 kcal/mol as a threshold for plotting IOU.

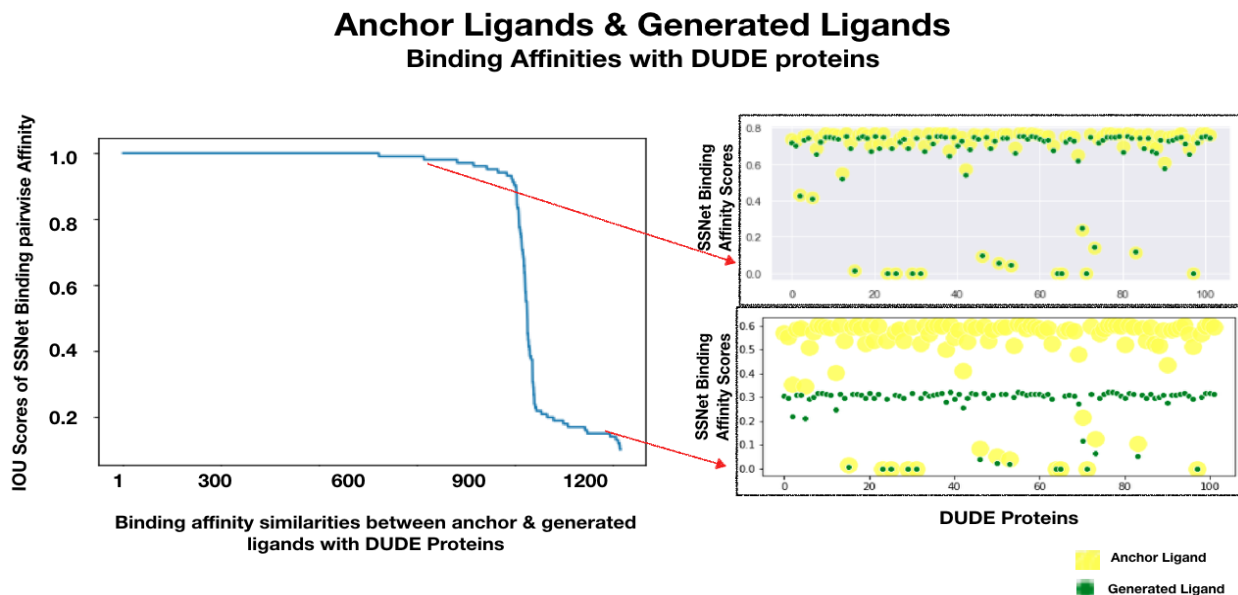


Figure 8.3: Pairwise binding affinity scores : The plot illustrates the similarities in the bio-activity between each pair of anchor ligands and their corresponding generated ligands to the 102 DUDE protein targets. The large regions of dark blue hue in the heat map demonstrate a strong co-relation between the binding affinities for most pairs with the DUDE targets. The scatter plot illustrates two sample pairs, with the top right plot representing a pair with very similar affinity scores, with the bottom right plot illustrating a pair where the affinities differ between the anchor and generated ligand.

While there is a high coherence of the scores obtained from SSnet, we further evaluated

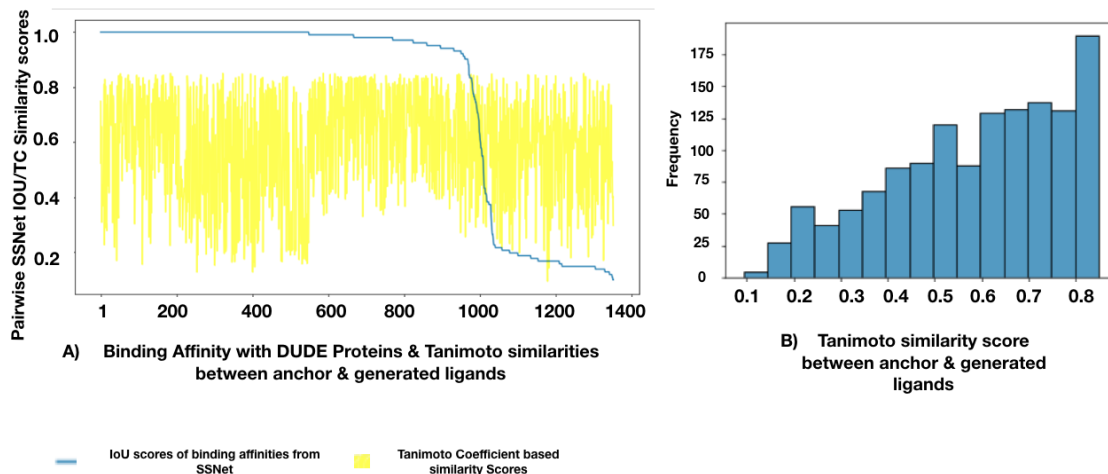


Figure 8.4: IoU scores and the Tanimoto coefficient scores : A) Comparison of the IoU scores and the Tanimoto coefficient scores between the anchor and generated ligands. : The figure illustrates that there is no strong correlation between the anchor and generated ligands. B) Histogram of TC scores across all the pairs : Illustrates the distribution of the similarity scores between the pairs.

the similarities between the anchor and generated ligands. We calculated the Tanimoto Coefficient based similarity scores between each pair of anchor and generated ligands. Figure 8.4 plots the IoU scores and the TC similarity scores for each pair. It is evident from the plot that there is no correlation between the IoU scores and the TC similarities. Despite the lack of correlation, the high coherence in the binding affinities could be explained by the scaffold similarities between the anchor and generated ligands. This is further evidenced by the scaffold analysis using the pseudo-hilbert curve as described in the subsequent section.

The analysis on QED, LogP, and SAS provided an intuitive relationship of generated ligands and drug-likeness. However, for a drug to be effective for specific target and show selectivity among other targets, should preserve the scaffold (core structure of a molecule [97–99]). To analyze if the generated ligands have similar scaffolds, we sorted all the anchor ligands by Tanimoto Coefficient (TC). The sorting was performed by recursively finding next most similar ligand from the anchor ligands starting from a random anchor ligand. The sorted list was then mapped to a pseudo-Hilbert space filling curve. The pseudo-Hilbert



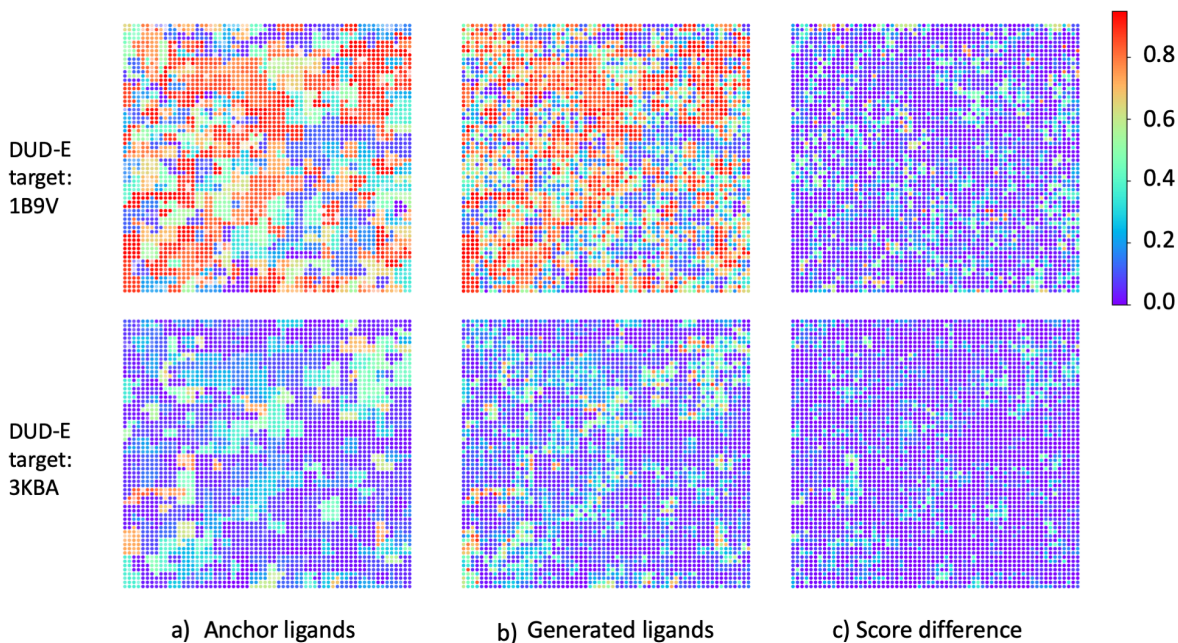


Figure 8.5: Scaffold analysis : A pseudohilbert curve is plotted for anchor ligands and generated ligands. The color denotes SSnet scores. Similarity between anchor and generated pseudohilbert curves and the low difference among them, signifies that our method retains scaffolds from the anchor ligands while also predicting similar bioactivities.

curve was used to observe molecular scaffolds directly from the map as pseudo-Hilbert curve preserves the spatial proximity of the sorted list. The pseudo-Hilbert map for the generated ligands were made similarly. Each anchor ligands were repeated to the same number of generated ligands in order to match one-to-one when comparing pseudo-Hilbert curve for generated ligands and anchor ligands. Figure 8.5a and 8.5b shows the pseudo-Hilbert map for anchor ligands and generated ligands respectively. The pseudo-Hilbert map is colored based on the SSnet scores obtained by docking the ligands with the DUD-E targets with PDB ID 1B9V and 3KBA.

We observe that the clusters are majorly retained for the generated ligands when compared to the anchor ligands. This is further highlighted in Figure 8.5c, that shows the difference in SSnet scores for generated and anchor ligands. The map is mostly blue which represents a mere difference of SSnet scores in generated and anchor ligand of less than 0.1. The results highlight that the novel molecules generated preserves the scaffold that is

essential in protein ligand binding.

### 8.3 Comparison with FDA Approved Drugs

In order to further hone in on the practical applicability of our methods and the potentially novel drugs generated, we conducted analysis on novel drugs generated on known cancer related ligands. For this exercise we shortlisted 10 drugs approved for treating various forms of cancer also available in the ChEMBL23 database. We present detailed analysis of the potentially novel ligands generated around a known cancer drug, DASATINIB. Sampling around the implicit fingerprint space of this anchor ligand yielded 10 novel ligands. Figure 8.6 illustrates the 10 novel ligands. The novelty of the compounds were tested from the ChEMBL23 dataset (1.4 million compounds) and the ZINC dataset (1.3 billion compounds). Across the 10 novel compounds, the maximum similarity score were 0.88 for ligands in the ChEMBL23 dataset and 0.92 for ligands in the ZINC dataset. Table 3(in supporting information [1]) shows the largest TC obtained for each novel compounds. Interestingly, in this particular case, we observe that the scaffold for the anchor ligand is retained in most of the generated ligands. The results are in line with the scaffold analysis performed for the DUD-E protein targets provided in the previous section (Figure 8.5). Retention of scaffold is crucial for ligand binding as the protein pocket in general has confined space for docking. The scaffold provides both size and imperative interactions such as hydrogen bonding,  $\pi$  interactions etc. that contributes to the stability of the protein-ligand complex.

To test the bioactivities for the novel ligands generated, we sorted 9 known targets for the anchor ligand, the details of which is provided in Table 2(in supporting information [1]). We conducted a docking method Smina [100] and a deep neural network based model SSnet [24] for bioactivity score prediction. Figure 8.7 shows the results obtained by the two methods. The first five targets are labeled active and the remaining four as inactive for the anchor ligand used in the ChEMBL dataset. We observe that the generated ligands have similar Smina scores as the anchor ligands. A similar behaviour is observed when comparing the SSnet scores for anchor and generated ligands. It is important to note that both Smina and SSnet are sensitive to ligand and their complex interaction with a protein target. Many factors such as functional group, size of the molecule, molecular weight etc. govern

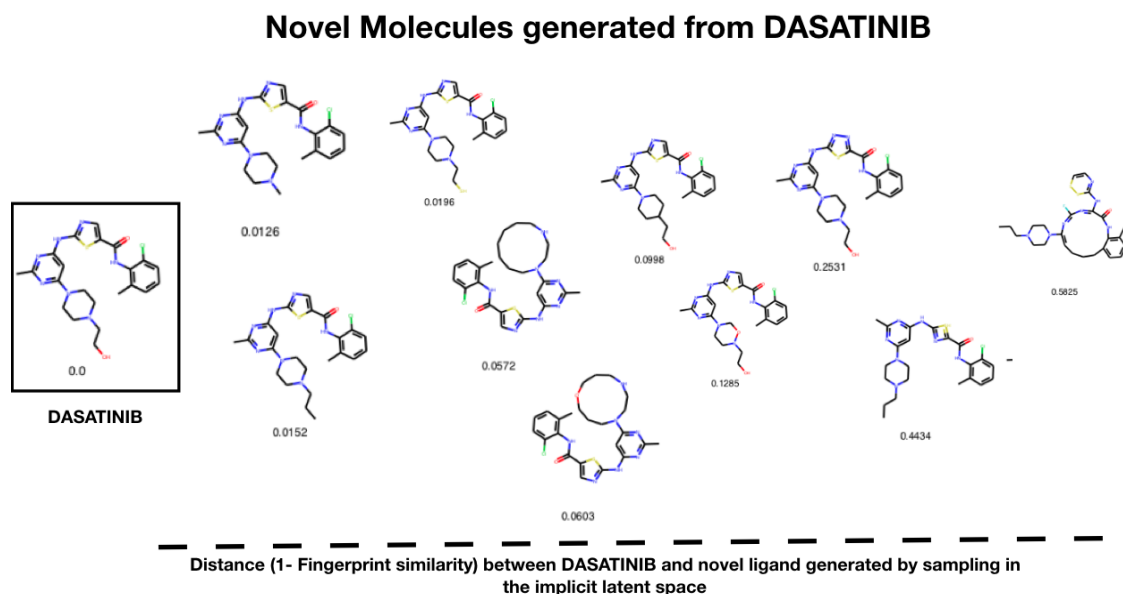


Figure 8.6: Novel ligands generated around known cancer drug, DATASINIB : It is observed that the generated ligands have ring structures similar to the anchor drug compound. In order to further validate if the rings are indeed realistic and ligands pragmatic, we investigated synthesizability, and like-lieness of the ring to bind a protein. Section 0.2.1 from Supporting Information [1] enumerates information about the novelty and the binding affinities exhibited by these generated ligands.

the bioactivity. The fact that all the 10 novel generated ligands have similar bioactivities provides evidence that our ligand generation method produces ligands with similar binding characteristics to the anchor ligand.

We further compared the bioactivities of 6 FDA approved drugs and their corresponding generated ligands from the implicit fingerprint and latent space generated from the variational autoencoder (VAE) work from [36] respectively. Each of the 6 ligands were docked towards their original intended target, the details of which is provided in Table 1(supporting information [1]). We observe high similarity in predicted bioactivities for implicit fingerprints compared to the latent space generated from VAE for both the Smina and SSnet scores shown in Figure 8.8 (Table 5-10) (in supporting information [1], sections 0.2.2 - 0.2.7). A visual inspection of the compounds generated from our method and the latent space from VAE [2] shows that both the scaffold of the original anchor ligand (Figure 2-7, supporting

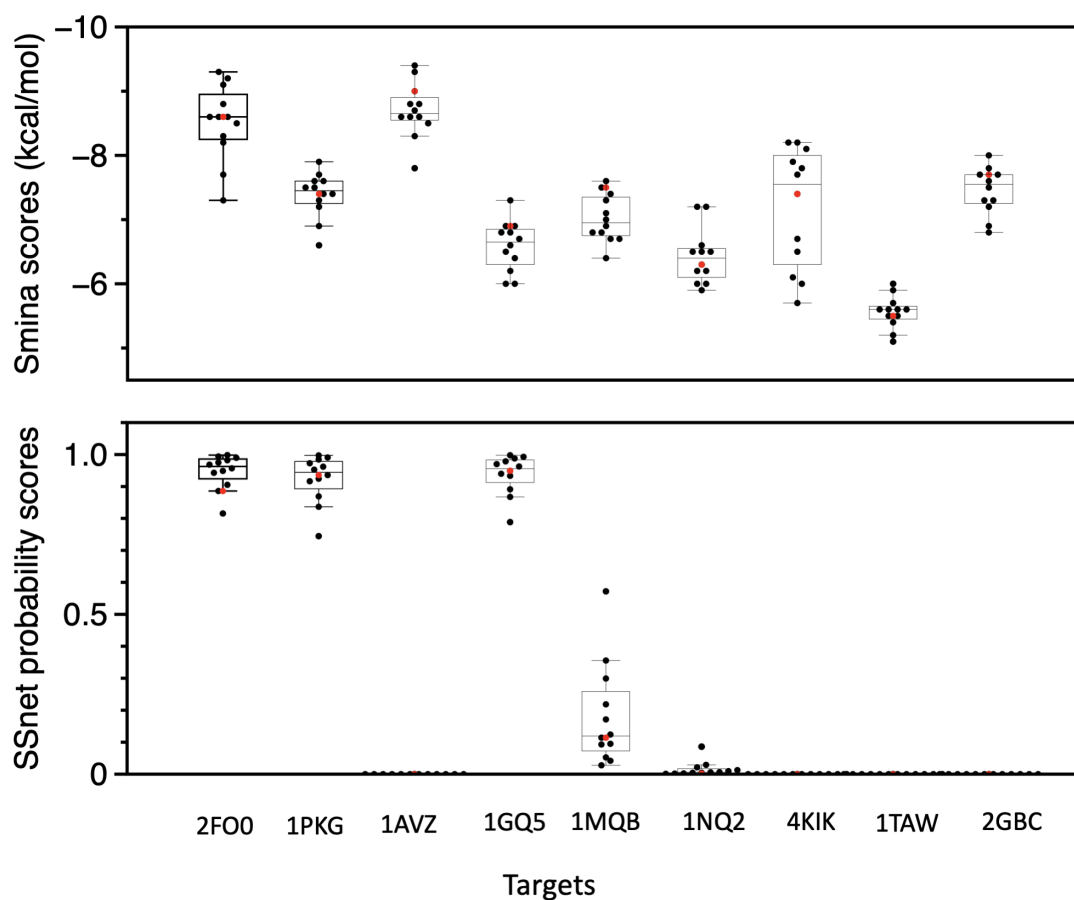


Figure 8.7: SMINA scores for generated ligands around DATASINIB : Sample test on various active/inactive targets for anchor ligands. The first 5 targets 2FO0, 1PKG, 1AVZ, 1GQ5 and 1MQB are active and the rest inactive. The red color denotes the anchor ligands and the black denotes generated ligands respectively.

information [1] section 0.2) and generated ligands are similar. However, bioactivity is sensitive to small changes in the chemical structure such as a functional group. Our method is perceptive towards functional groups due to the way collaborative fingerprints was modeled, i.e. by considering the bioactivities.

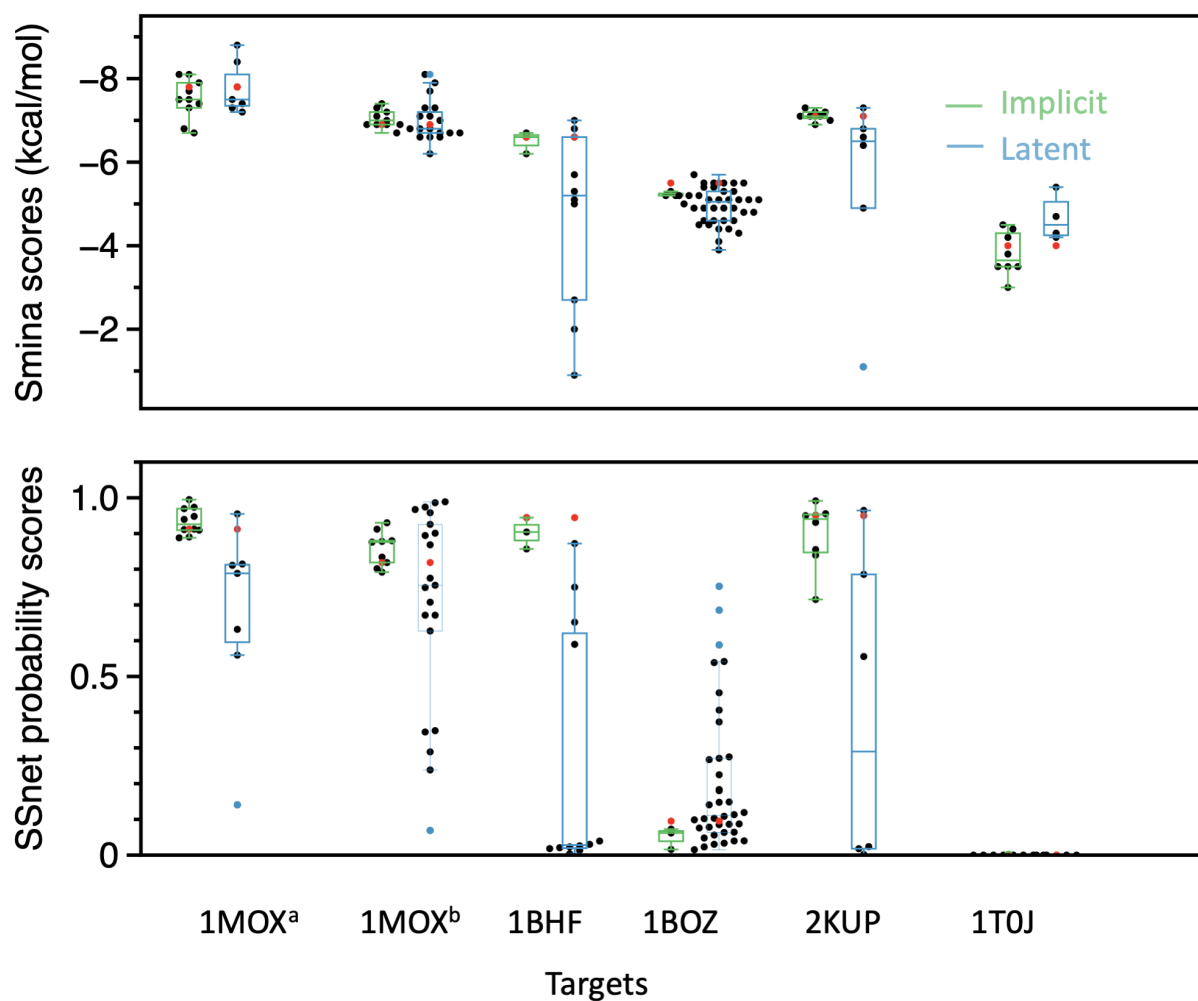


Figure 8.8: Comparison with Variational Autoencoder based fingerprints : Comparison of Implicit and latent fingerprints on FDA approved drugs and their corresponding targets. The red color denotes the anchor ligand and the black color denotes generated ligands respectively. The latent label and implicit label shows the binding affinities for generated ligands from the method developed by [2] (in blue) and our method (in green).

## Chapter 9

### CFGenNets for Image Generation

#### Introduction

Research Aim 1, outlined in the previous chapters demonstrated that implicit fingerprints capture ligands and proteins in a shared latent space, typically for the purposes of virtual screening with collaborative filtering models applied on known bioactivity data. Research Aim 2 further extended this concept and introduced the concept of CFGenNets, a novel generative deep learning algorithm that could not only reconstruct the original ligands but also generate novel ligands from the implicit ligand fingerprint space. Our third and final research objective as a part of this thesis involves evaluating the abilities of CFGenNets to generate images with implicit image fingerprints derived from the corresponding image meta data. That is, we seek to evaluate our generative methodology in a new application area—images. The recent years have seen rapid advances in the domain of caption to image generation [72,101–104]. These methods attempt to generate realistic images that match the captions semantically. In this aim, we propose to generate images, however from a structured set of attributes describing the images, as opposed to the unstructured natural language text. Such a function could have meaningful applications in areas such as image manipulation and gaming. For example, Open AI unveiled DALL-E [105], which was trained to generate images from unstructured text. DALL-E demonstrated the creation of novel objects in images by combining unrelated concepts, e.g., an armchair shaped like an avocado. Going beyond these, the creators of DALL-E also demonstrated more practical applications such as image manipulation by generating shadows of objects within images, fashion design, and more generally product design and conceptualization (e.g., conceptualization of spaces by merely controlling the caption to generate relevant images). While we are focused on structured sets of attributes as opposed to unstructured caption text, with this aim we investigate the

applicability of CFGenNets to provide a framework where the user can manipulate synthetic images by altering structured attributes (e.g., keywords). Hence, in theory some of the same applications described above in the context of caption to image generated are applicable to CFGenNets based image generation capabilities. In general, image understanding and representation are thriving research communities and our methodology may have informative and interesting contributions that complement existing theory.

## 9.1 Motivation for Image Synthesis

While the previous chapters described in detail the applications of CFGenNets in the field of ChemInformatics to generated novel drug compounds, there is still a need to evaluate the cross domain applicability of CFGenNets. In this context, the domain of image synthesis, which aims to generate images given a set of descriptors is naturally aligned to the application of CFGenNets. The images with their shared descriptors can be described in a tabular form, with the rows representing images and the columns representing the attributes. Once in this form, the tabular data is conducive to the application of collaborative filtering to generate latent representations of the images. While technically, the domain of image synthesis is conducive to the application of CFGenNets, there also exists practical applicability as evidenced by the recent research trends in the field of image synthesis.

### 9.1.1 A Primer on Related Work in Image Synthesis

The recent years have seen tremendous progress in the field of deep learning based image synthesis and manipulation using generative neural networks. The Variational Autoencoders [67] (VAEs) are a class of generative models that attempt to learn the underlying distribution of the data to decipher how the data is generated. It aims to identify the underlying distribution of the data which allows for sampling to generate new data, or images in our case. The traditional VAE uses an encoder and decoder pair of neural networks (typically convolutional). The encoder reduces dimensionality and is designed such that the reduced dimensionality space can be sampled from to produce images. In general, VAEs are designed with a loss function that simultaneously reduces error between the image input and the reconstructed image, while also keeping the latent space samples normally distributed. This is typically achieved by placing a KL-divergence constraint for Gaussian samples in



the latent space through the loss function. Several works have attempted to deploy VAEs and their variants to obtain neural embeddings on celebA datasets and manipulate the latent spaces to generate novel synthetic images. [106] demonstrated the use of VAEs not only to reconstruct the original images, but also to extract features correlated to binary labels in the data. Their model, the Conditional Subspace VAE (CSVAE) [106], used mutual information minimization to learn a low-dimensional latent subspace associated with each image attribute, that could be easily interpreted and also used for image manipulation. Li et al. [107], introduced Latent Space Adversarial Auto Encoders, where the latent space from the autoencoders are mapped into a style space and an attribute space. The former represents attribute-irrelevant factors, such as identity, position, illumination and background, etc. The latter represents the attributes, such as hair color, gender, with or without glasses, etc, of which each dimension represents one single attribute [107]. Yan et al. [108] investigated the problem of image generation by modeling images as a composite of foreground and background. They developed a model with disentangled latent variables using VAEs called disentangled Attribute-Conditional generative models. Their model relied on interpreting an image as a composite of a foreground image and background image combined via a gating function with the model training process aimed at learning the latent representations of the background and foreground images. Cai et al. [109] extended the concepts of VAEs by producing Multi-Stage VAEs where the decoder is divided into 2 components where the second component generates refined images based on the coarse images generated by the first component. They decoupled the second component from the VAEs which allows other loss functions beyond the MSE based loss function typically employed in VAEs. Razavi et al. [110] introduced VQ-VAEs, or Vector Quantized VAEs which builds on the ideas of lossy compression in techniques such as JPEGs [111]. Their argument stems from the idea that it is possible to remove more than 80% of the data without noticeably changing the perceived image quality. They compress images into a discrete latent space by vector-quantizing intermediate representations of an autoencoder. In their version of VAEs, the encoders output discrete encodings instead of continuous vectors using the concepts of vector quantization [112]. Besides VAEs, the Generative Adversarial Networks (GANs) have also been prominently used for image synthesis and generation in the recent past. As



a framework of generative model, Generative Adversarial Net (GAN) [113] aimed to generate better synthetic images than previous generative models, and has become a popular research area. A Generative Adversarial Net consists of two neural networks, a generator and a discriminator, where the generator tries to produce realistic samples that fool the discriminator, while the discriminator tries to distinguish real samples from generated ones [72]. This methodology has progressed dramatically in the past decade such that many generated images are quite realistic. As mentioned earlier, the GANs leverage discriminator network is trained to tell apart the synthetic data from the real examples. The difference between the synthetic and the real examples determined during each step of the training process produces a gradient that guides the learning of the two networks. The recent years have seen a number of enhancements to the core idea proposed by Goodfellow et al. [72]. For example, Hjelm et al. [114] proposed Boundary Seeking Generative Adversarial Networks, to handle discrete settings by means of a policy gradient for training the generator which is informed from by the estimated difference measure from the discriminator to compute importance weights for generated samples. Mao et al. [115] proposed the adoption of L2 loss function as an effective measure to overcome the saturation problem in GANs after exhaustive experiments on large image datasets. Zhao et al. [116] proposed Dual-Agent GANs, where they employed an off the shelf 3D face model as a simulator to generate profile face images with varying poses. They specifically attempted to address the problems associated with the unbalanced distribution of poses in available face recognition datasets. They employed two discriminators that the generator played against — the real-fake discriminator and an identity discriminator. Additionally, Mirza et al. [117] proposed the conditional version of GAN, with condition on both the generator and discriminator for effective image tagging. Berthelot et al. [118] proposed Boundary Equilibrium GAN (BE-GAN) framework where they leveraged Wasserstein distance loss for controlling the trade-off between image diversity and visual quality.

Some of the more seminal works in the space of image synthesis is in the domain of Neural Style Transfer. Gatys et al. [119] successfully demonstrated that the neural networks can encode both the content and the style information of images and that it was possible to separate them. Huang et al. [120] enhanced the optimization process to enable a real time style transfers by introducing Adaptive Instance Normalization technique. They modified

the concepts of Instance Normalization where they align the channel-wise mean and variance of the content input to that of the style input. The Adaptive Instance Normalization involves removing the style information from the content image by normalizing and performs style transfer to the content image by transferring feature statistics, specifically the channel-wise mean and variance from the style images to the content images. In this technique the affine parameters are not learnt but adapts the affine parameters from the style input. Karras et al. [121], motivated from style transfer literature redesigned the generator architecture to control the image synthesis process. They introduced a nonlinear mapping network to generate an intermediate latent space, which provides the style information to enable Adaptive Instance Normalization. With GANs as their foundation, another notable advancement is Stack GAN [122], which consisted of two stages of GANs, with Stage One producing the basic shapes for images at relatively low resolution. This is further enhanced by Stage Two GANs, which add realism and resolution to the base images. For this task, the latent space cannot simply be random, but must be discovered from the image caption used to generate the image. To this end, recent years have also seen the advent of Recurrent Adversarial networks to map natural language input to visual images [123]. In this work, recurrent networks are used to transform captions into a latent encoding and this encoding is used to reconstruct an image. This is a similar process to adversarial auto encoding [124], except that the input is a sentence, not an image. Further, alignDRAW [125] was another mechanism that leveraged Recurrent neural networks in lieu of Convolutional Networks to generate images using soft attention mechanisms. Similarly, Zhi et al. introduced PixelBrush [126], a tool to generate art from text description from human written descriptions using Skip-thoughts text embedding (a popular form of sentence embedding). PixelBrush used Generative adversarial networks to produce artistic images conditioned on human-written descriptions. Skip-thoughts were used for text embedding of 4800 dimension and conditional generator and discriminator network for generating images [126].

The recent popularity in the field of image synthesis, as evidenced by the continuous stream of novel research, and the natural alignment of the problem statement to Collaborative filtering for generating the latent image vectors, serve as motivation for us to choose this as an application to evaluate the cross domain applicability of CFGenNets. By illustrating

the versatility of the CFGenNet algorithm to a popular research field, we hope to maximally impact the research community of our peers. Our contribution, therefore, can be thought of as the first work to investigate the use of collaborative filtering with generative deep learning analysis. We hypothesize that the latent space learned by collaborative filtering offers unique qualities that previous works have had difficulty achieving. We therefore explore the space of image synthesis with our CFGenNet algorithm, and manipulate the algorithm to better represent images.

## 9.2 Methodology

We adopted the following steps in order to investigate the applicability of CFGenNets for Image synthesis:

- Constructing a matrix to represent the images from the data and their respective attributes. Here the rows represent each image, while the columns represent the entire set of distinct attributes that capture key aspects and/or objects contained within the images.
- Performing collaborative filtering on the said matrix to derive the implicit image fingerprints for each image. We hypothesize that this implicit latent space will have more utility in synthesizing images, similar to our findings in virtual screening.
- Training a decoder with convolutional layers to generate the images from their corresponding implicit image fingerprints.

This methodology is similar to that employed in CfGenNets for drug discovery, but using images and image attributes to train the collaborative filtering algorithm. Later on, we will show that using this base methodology produces good, but not great, image reconstructions. Therefore, we combine our methodology with VAEs to produce more photo-realistic images (discussed later).

### 9.2.1 Dataset details

We leveraged the CelebFaces Attribute Dataset (CelebA) [3] for our evaluation of CFGenNets for Image processing. CelebA is a large dataset with around 200k celebrity images

with 40 attributes describing the facial features found in the images. The images in this dataset cover large pose variations and background clutter. Figure 9.1, reproduced from the original article [3] in this paper for easy reference, illustrates the a sample set of images and the attributes associated with them. In order for us to be able to apply collaborative filtering algorithm on the dataset, we recast the CelebA faces dataset in terms of a large matrix consisting of 200k rows, each row representing one image and 40 columns, with each column representing the facial attribute descriptors. Figure 9.2 illustrates the matrix with two sample images.



Figure 9.1: Sample Images from CelebA dataset. Reproduced from the original article [3] here for easy reference.

### 9.2.2 Collaborative Filtering on CelebA dataset

The next step in our process (once the matrix described in Figure 9.2 is constructed) is to run the collaborative filtering algorithm. In our dataset, there are 200,000 rows (one for each image) and 40 columns (one for each image attribute). We leveraged the same set of steps



	Smiling	BlondHair	Male	..	Wearing Glasses
	1	0	1	....	1
	1	1	0	....	0

Figure 9.2: CelebA dataset recast as a matrix with each row representing one image from the dataset and the columns representing the attributes described for the image.

as described in Chapter 3 to generate two sets of fingerprints. One set of fingerprints for the images called Implicit Image Fingerprints and another set of 40 fingerprints representing the 40 attribute descriptors. We chose the latent representation to be 500 dimensional for the collaborative filtering. Thus, the fingerprints are continuous vectors of size 500 dimensions.

### 9.3 Analysis of Implicit Fingerprint

As described in the previous section, our method generates implicit fingerprints for images and the 40 features that describe the images. By definition of collaborative filtering, as outlined in Chapter 2, the implicit image and the feature fingerprints share the same latent space. For visualization, we reduced the dimensions of the implicit fingerprints from 500 dimensions to 2 dimensions using the t-SNE [61] algorithm. The t-SNE algorithm attempts to retain the same mutual distances of the data points even in a lower dimensional space. Figure 9.3 illustrates the images and the 40 features in a reduced 2 dimensional space. It can be seen that the features and the images share the latent implicit fingerprint space with the features interspersed amidst the images. In this way, one could reduce an image into this latent space or construct a set of 40 attributes for an image and map into the same latent space. This property is key to understanding why the collaborative filtering latent space might be advantageous compared to other methods—the shared latent space might allow for increased understanding because we can “probe” values from the image attributes or the

image itself.

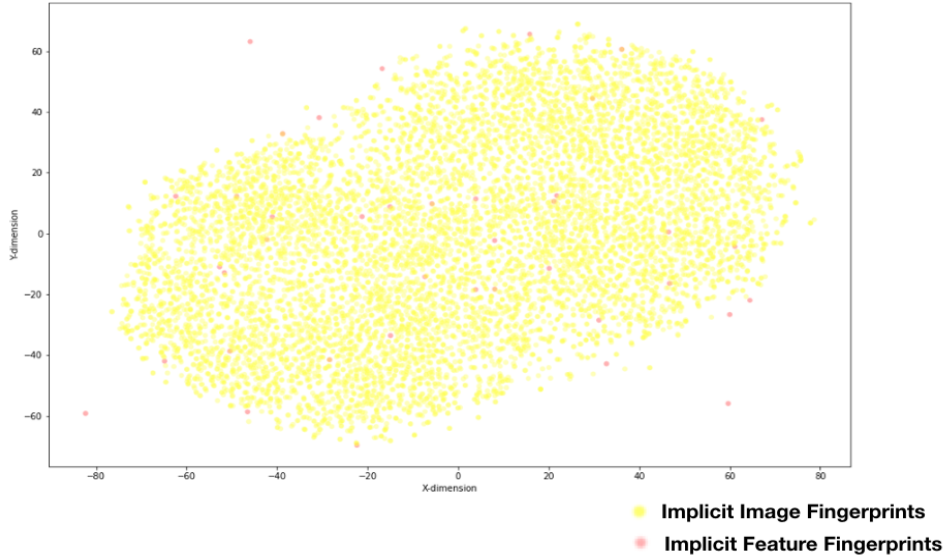


Figure 9.3: Representation of the implicit image and feature fingerprints in a 2 dimensional space using t-SNE algorithm. The images and the features both share the same latent space, with the features interspersed amidst the images.

We further evaluated the predictive powers of the implicit image fingerprints for the task of Facial Attribute Recognition, ie facial attribute prediction. We trained a multi-class classifier using a neural network with fully connected layers. The input to the network was the continuous implicit image fingerprints and the network was trained to predict the presence or absence of 40 facial attributes contained in the respective images. We used 160k images for training the network with 40k images as the validation set. The results of the training is illustrated in Figure 9.4. One can think of this classifier as a “reverse encoder” for the latent space. That is, it maps from the latent space back to the 40 dimensional image attributes. By performing this classification, we can further understand if the latent space is converged and reliable at separating images with similar attributes.

The classifier was trained to minimize the binary cross entropy loss and we measured

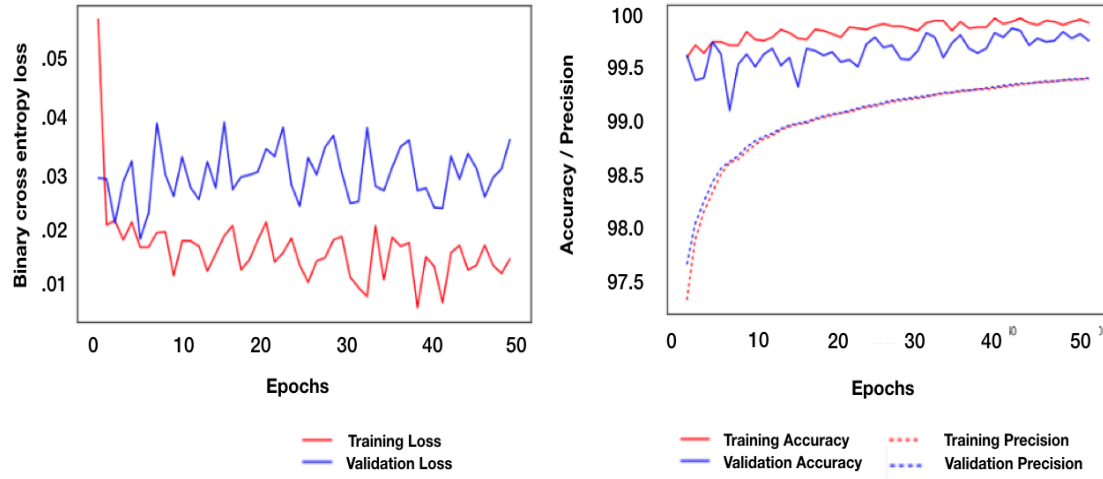


Figure 9.4: Performance Metrics of facial attribute prediction classifier trained on the implicit image fingerprints.

the accuracy and the precision of the predictions. We see the classifier performing with very high accuracy and precision, both hovering around 99%. This indicates that the implicit image fingerprints indeed encode the information on the distinct facial attributes contained in the images. We hypothesis that these patterns encoded in the implicit fingerprint space would enable the CFGenNet decoders for images learn the patterns in order to reconstruct the original images.

### 9.3.1 Implicit Feature Fingerprints

As we described earlier, the collaborative filtering exercise produces latent encoding for both the images and the 40 facial features/attributes that describe the images. The latent space encoding of the facial features or the Implicit Feature Fingerprints also share the same latent space as the images. This is property is also captured in the figure 9.3, where we see the features interspersed with the images in the reduced 2-dimensional implicit fingerprint space.

We further analyzed the abilities of the Implicit Feature Fingerprints to be able to successfully predict the corresponding facial attribute. We used the classifier trained on the



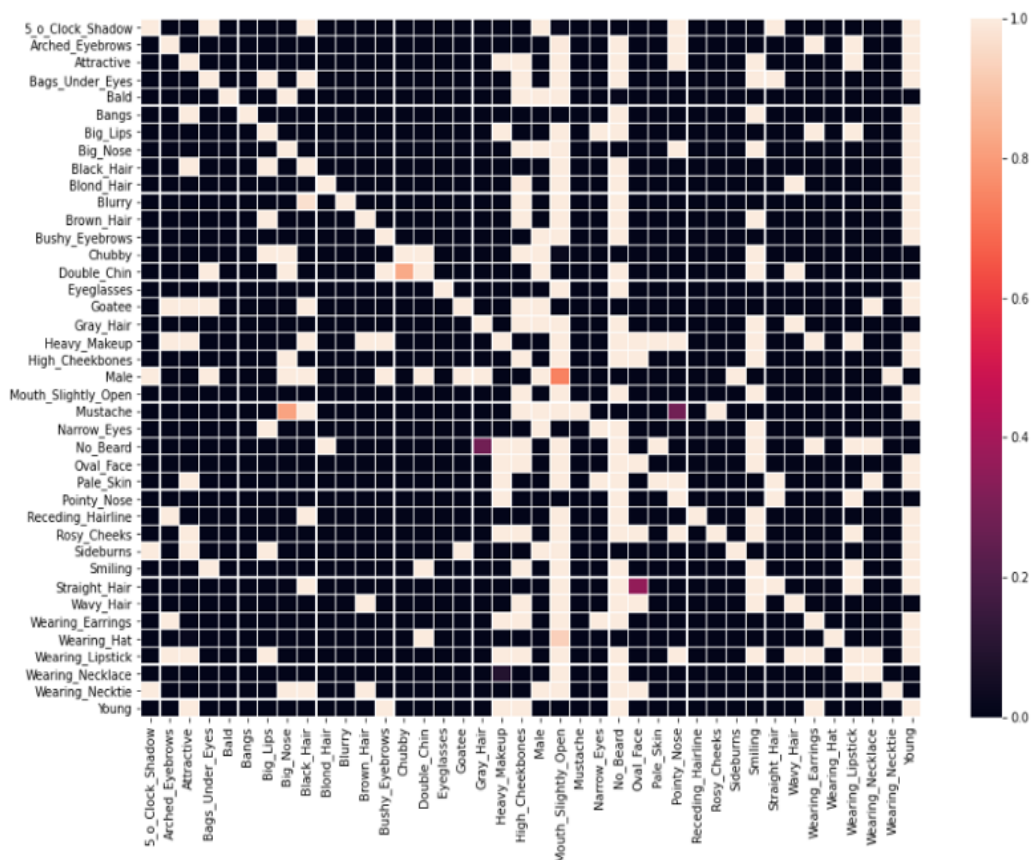


Figure 9.5: The hash map illustrates the abilities of the Implicit Feature Fingerprints to predict the presence of respective facial descriptor attributes in the images. The Implicit feature fingerprints generate accurate predictions even though they are fed into the classifier trained solely on the Implicit Image Fingerprints. This is possible because the features and images share the same latent fingerprint space when generated via Collaborative Filtering.

Implicit Image Fingerprints, described in the previous section, to generate the predictions using the Implicit Feature fingerprints. The implicit features were able to predict with close to a 100% accuracy. Figure 9.5 further illustrates the predictions from the implicit feature fingerprints. The rows in the figure represent the fingerprint vectors for each of feature and the columns represent each facial attribute. The cell at the intersection of the rows and columns is representative of the prediction generated from the Facial Attribute classifier. As is evident from the hash map, the implicit feature fingerprints encode information about the attributes they represent. In addition, they also encode information on other correlated



attributes. For example, we notice that the fingerprint for 'Wearing Makeup' predicts with high confidence the presence of other facial attributes such as attractive, wearing lipstick, arched eyebrows, and female. While wearing makeup does not necessarily indicate these other attributes, the CelebA dataset clearly has correlations that reflect these similarities. As illustrated in Figure 9.5, the Implicit Feature Fingerprints predict the presence of respective facial descriptor attributes in the images with very high accuracy. The Implicit feature fingerprints generate accurate predictions even though they are fed into the classifier trained solely on the Implicit Image Fingerprints. This is possible because the features and images share the same latent fingerprint space when generated via Collaborative Filtering.

### 9.3.2 Conclusion

In this chapter, we introduced the concept of CFGenNets for image processing and described in detail the data set used and the methods applied to generate the implicit fingerprints. We also presented the analysis conducted on the predictive powers of the Implicit Image and Feature fingerprints. Given the predictive capability of the implicit fingerprints, we are comfortable moving forward with additional analyses to produce images from the implicit fingerprints. We describe our experiments to around building CFGenNets for image generation in the subsequent chapter.

## Chapter 10

### CFGenNets for Image Generation : Architecture

In this chapter, we extend the utility of CFGenNets to the domain of image generation. We explore the abilities of CFGenNets to reconstruct images from their implicit fingerprint space. The previous chapter described the methods adopted to generate the implicit image fingerprints. In this chapter, we describe in detail the experiments conducted on CFGenNets for image generation and synthesis.

#### 10.1 Decoders to generate images

While it is observed that the implicit fingerprints for Images encode attributes and features of the images in the latent space, we propose the usage of the fingerprints to reconstruct the corresponding images. We believe this can form the foundation for enhanced applications such as generation of synthetic images and image manipulation by manipulating the corresponding implicit fingerprint space. Unlike the previous related works described in Chapter 9, in our research, we start from the latent space obtained from collaborative filtering instead of generating the latent space from a network that has processed images in some form. The subsequent sections describe our network architecture and the results. Additionally, we also demonstrate in subsequent chapters that the Implicit Feature Fingerprints obtained from Collaborative Filtering itself encodes the style information along with the Implicit Image Vector encoding the content information, thereby providing a framework for image manipulation.

#### 10.2 Decoder Architecture

From our experiments outlined in the previous chapter, it is evident that the implicit fingerprints from collaborative filtering encapsulate certain aspects of the underlying images and the features that describe the images. Here, we design deep neural networks to decode the implicit image fingerprints and map back to the original image. Recall that the implicit

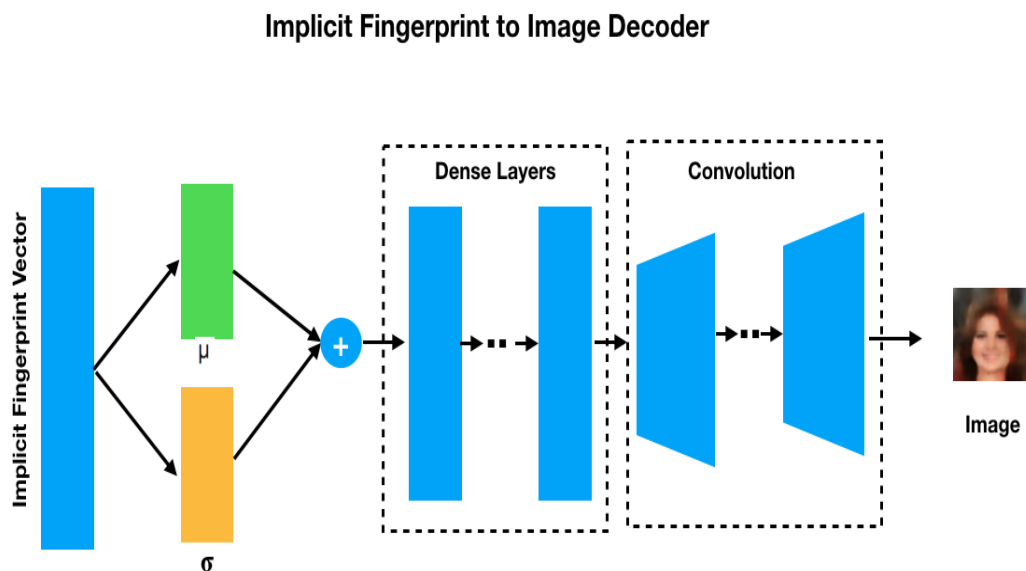


Figure 10.1: CFGenNets architecture for generating images from implicit image fingerprints generated from collaborative filtering.

fingerprints of the images are continuous vectors that represent the images in a space with 500 dimensions. This is fed as an input to our custom decoder which consists of a series of fully connected layers followed by a variant of two dimensional convolution layers. These layers up-sample the image and provide details based upon the latent space representation. Ideally, the images would look perceptually similar to their original forms.

As outlined in the CFGenNets for drug discovery in the preceding chapters describing Aim 2, we adopt a mechanism of input data augmentation. Similar to our previous work in ChemInformatics, this is done to avoid the latent space from being sparse. We perform data augmentation by adding randomness to the input layer of the neural network. This stochastic noise added to the decoder input ensures that the decoders learn to decode from a wider variety of latent points to find more robust representations. This is similar to the process by which Variational Autoencoders (VAEs) sample in the latent space before reconstruction. Our network consists a series of fully connected layers processing the input implicit image vector followed by convolution layers to upscale the dense representations and map to valid images. The up-sampling operation performed was evaluated with two

competing convolutional techniques that are each common in generative deep learning with images (discussed more in the next section). The general flow of the process is shown in Figure 10.1.

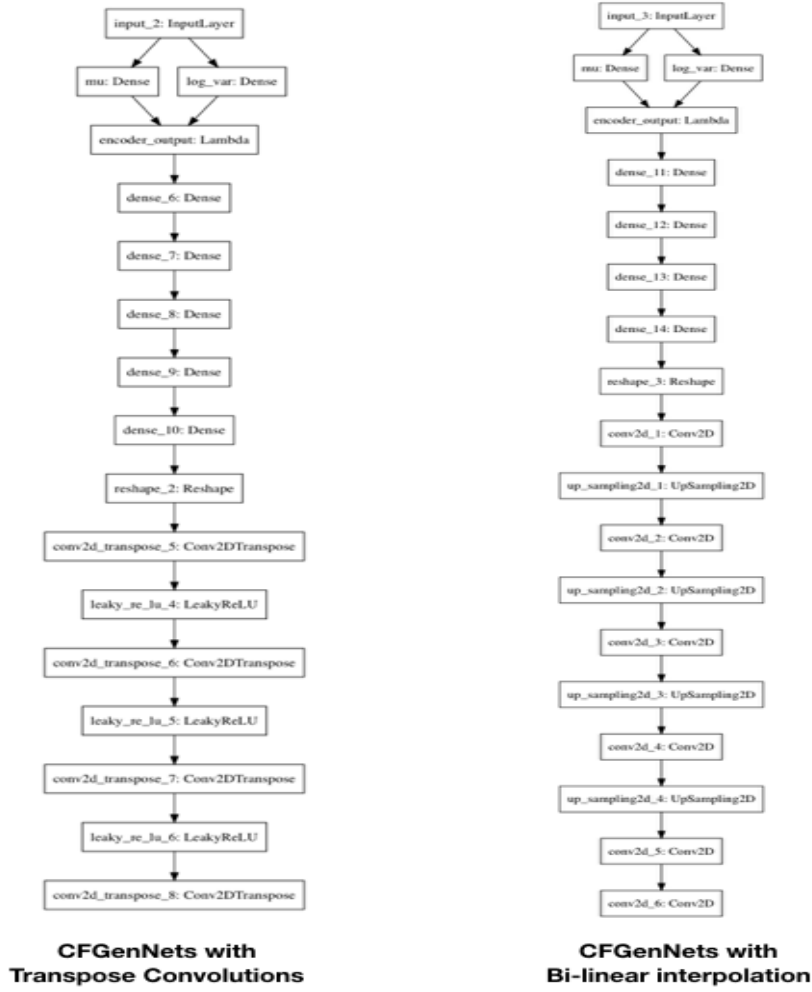


Figure 10.2: CFGenNets for Images : Network design using transpose convolution operations and bi-linear upsampling.

### 10.2.1 Convolution Layer Design

Convolution layers are standard components of neural networks that deal with images. The convolutional layers facilitate representational learning, a method that aims to learn relevant features for a given task without manual feature engineering. In the context of image related tasks, convolutional networks exploit the facts that nearby pixels in any given image are likely to be strongly related compared to farther pixels, and that objects represented in an image are made up of smaller parts. The convolutional network uses filters, which are moved from the top left of the image to bottom right using a standard two-dimensional convolution operation. Each point on the image is represented in the filter by a value calculated by the convolutional operation. The convolutional operation typically results in down-sampling, i.e., the spatial dimensions of the output layer is lesser from the input layer. However, for our purposes of producing an output image of dimensions 128x128 from a dense input vector of 500 dimensions, we apply the filters as a means of up-sampling operations in the neural networks. That is, the learned filters in the network function as image interpolating filters. The up-sampling operations can either be conducted via interpolation such as bi-linear interpolation or transposed convolution [127] methods. In bi-linear interpolation, the input to the convolutional layer is doubled in size using bi-linear interpolation and then filters are learned that smooth out this interpolation process and add detail. Bi-linear interpolation also is one of the re-sampling techniques used in computer vision and image processing applications [128]. The bi-linear interpolation extends the concepts of linear interpolation in the 2 dimensions. It is basically a method of curve fitting using linear polynomials to construct new data points within the range of a discrete set of known data points. Several recent works in the realm of deep-learning based image processing applications [129–132] have successfully demonstrated the viability of simple interpolation techniques as opposed to parameter learning methods for up-scaling images.

In the transposed convolution methods [127], the network attempts to learn the most optimal method of up sampling by parameters that are learnt during the training process. This is similar to adding “zeros” on the rows and columns of the input and then the convolutional filter learns to interpolate these “zeros” to add detail to the image. The deep convolutional generative adversarial networks (DCGAN) [102] successfully employed the transpose convo-

lutional operations to generate images from random sampled values, following which several variants of generative adversarial networks employed similar approach. The application of transpose convolution can also be found in the image based variational auto encoders [109], where convolutional layers are deployed to extract features in the encoder and then decoders restore the original image size by applying transpose convolution operation.

We experimented using both variants (bi-linear and transpose convolution) in our decoder. We present the details of the experiments conducted and the results obtained. Figure 10.2 shows the architecture employed for each convolutional layer type investigated.

### 10.3 Results

We performed extensive training and validation of the CFGenNets with the continuous implicit image fingerprint vector as the input and corresponding image as the output. We measured the outcome of the models by evaluating the mean square error loss and the reconstruction accuracy. For images, reconstruction accuracy is the percentage of pixels that match exactly with the original image, which is a commonly reported evaluation metric. As a part of training, we explored a variety of architecture options with respect to the depth and the width of the deep learning model, include the two variants of up-sampling - transpose convolutional operations and bi-linear interpolation up-sampling. In order to compare the performance of CFGenNets across the 2 up sampling methods, we trained the two variants of the neural networks as illustrated in Figure 10.2 on a sample set of randomly selected 10,000 images and their corresponding implicit fingerprints. Figure 10.3 illustrates the performance metrics from the two models after training for 300 epochs. As is evident, the MSE loss of the 2 networks are quite comparable, hovering around .01. The reconstruction accuracy of the networks the end of 300 epochs is a 76% and 75.5% for network with transpose convolutions and bi-linear interpolation, respectively.

While the quantitative metrics of the two variants were very comparable, we also qualitatively evaluated the output of the models with visual inspection. A sample of output images is presented in Figure 10.4 for illustrating the nature of outcomes from the two networks in comparison with the original images. We observe that both pairs of images, from transpose convolutions and up-sampling methods can generate clean images that are clearly human

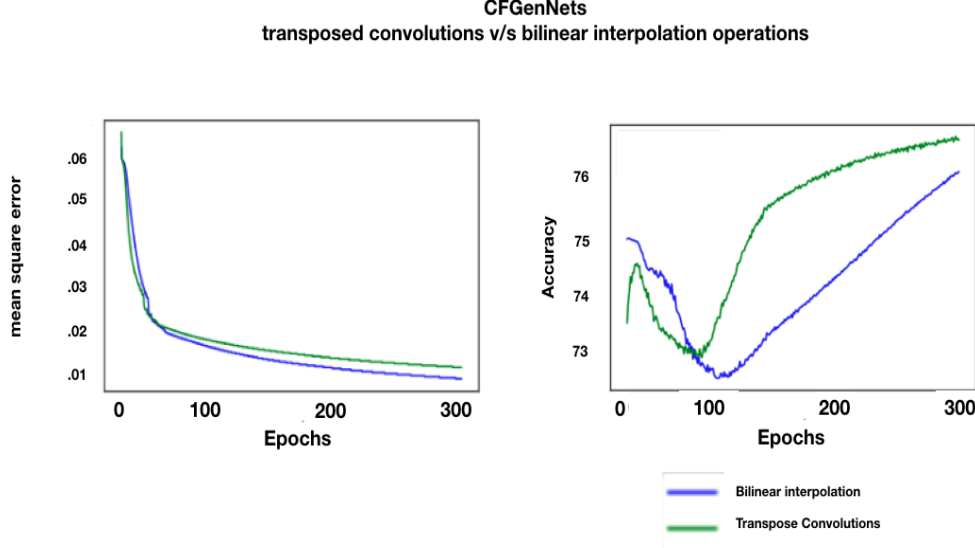


Figure 10.3: Performance metrics across the 2 networks trained on celebA dataset. After 100 epochs, the network encompassing transpose convolution layers has a lower mean square error and higher reconstruction accuracy.

faces. The images have preserved the overall face and object structures, however we also notice that the images are blurry and not sharp. As discussed above, our network aims to minimize the per pixel reconstruction loss between the input and output images. We hypothesize this effect to be similar to prior works with generative image models [133], where the images tend to get sharper when trained to minimize the feature reconstruction loss as opposed to the pixel reconstruction loss. However, we observe that CFGenNets for Images are indeed able to generate output images consistent with their respective inputs with reasonable and perceivable facial features and backgrounds. For some images, the reconstructions appear “more generic” than the input image. That is, the output appears to have lost some of the distinct facial structures and the head pose appears straighter toward the camera. Mouth shape sometimes appears distorted, but eyes tend to take the same position as in the original images. Both the networks were able to map the implicit image fingerprints obtained from collaborative filtering back to the original image with approximately 75% accuracy and little difference between each method can be observed. In summary, blurring appears to be

the most noticeable artifact, as well some replacement of specific facial features with more “general” facial features—however, the essential aspects of the images like face shape, mouth and eye position, are captured observably well.

An additional set of images are presented in Appendix A12.3.



Figure 10.4: A small sample of generated images in comparison with the original images. Appendix A12.3 contains a larger sample of images generated from CFGenNets.

In order to evaluate if the models are indeed generalizable, we trained the CFGenNet variant with transpose convolution layers on the entire dataset of 200k images, with 80%-20% train and test split for 100 epochs. Figure 10.5 illustrates the performance of the network in terms of the mean square error loss and the reconstruction accuracy. The mean square error for this network was found to be hovering around .03 with image reconstruction accuracy around 75%. We note that there is some flattening of the performance on this large dataset, but the y-axes are considerably Zoomed, so performance remains relatively constant. As for the computational time, the models were built using the Keras library with Tensorflow



backend. It took approximately 24 hours to train models for 100 epochs. During testing, it took approximately .01 seconds to process a single image of dimensions 128X128 thru the network. These training and test times are benchmarked on the standard compute nodes on the SMU Maneware systems,consisting of 36 cores, 256 GB of memory, and 100 Gb/s networking.These nodes contain dual Intel Xeon E5-2695v4 2.1 GHz 18-core “Broadwell” processors with 45 MB of cache.

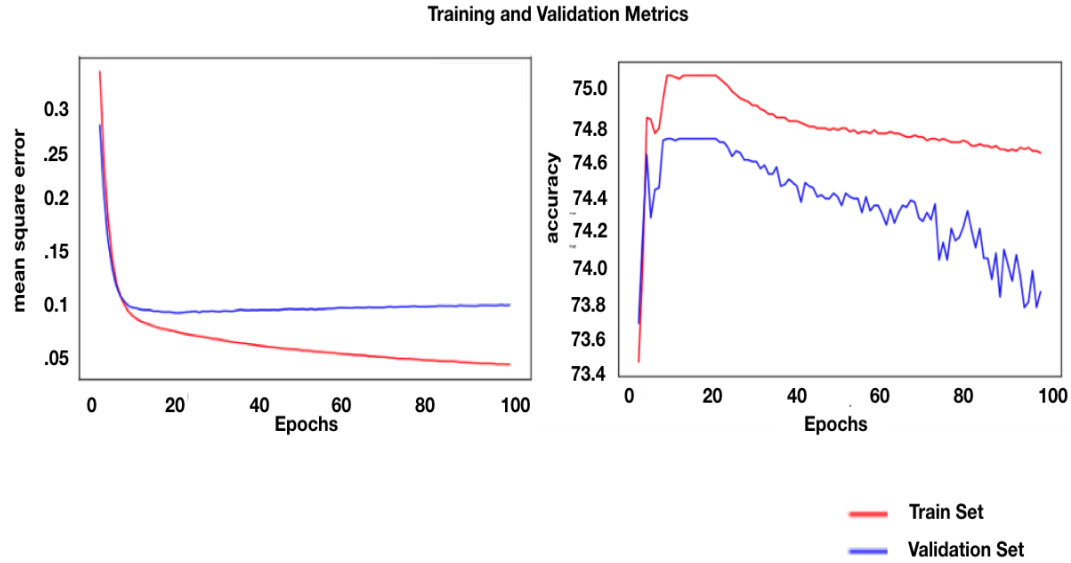


Figure 10.5: CFGenNets for Image generation : Performance metrics at the end of 100 epochs on the final model trained on 160k images and evaluated on 40k images.

### 10.3.1 Limitations and Next Steps

In this chapter, we demonstrated the abilities of CFGenNets to be able to generate images from the implicit fingerprints obtained from collaborative filtering. We observed that the network was indeed to able to reconstruct the images with 75% accuracy and low mean squared error. The manual inspection also demonstrated the ability to visually perceive the output when compared with the original image. While the outcomes affirm the original

research objective, i.e., to evaluate the applicability of CFGenNets in image generation, we also note that the output quality of the images could be better, potentially with additional hyper parameter tuning and tweaks to model architecture. In the next chapter, we explore this further, conducting additional experiments to evaluate how the CFGenNets trained on implicit fingerprints compares with traditional variational autoencoders using a similar architecture and trained on the same dataset splits. The details of the experiments conducted and results obtained are outlined in the subsequent chapter.

## Chapter 11

### CFGenNets and Variational Autoencoders

We demonstrated the abilities of CFGenNets to reconstruct original images from Implicit image fingerprints obtained from collaborative filtering in the previous chapter. We observed that the reconstruction accuracy was approximately 75%. The outcomes validate the hypothesis that the implicit fingerprints indeed capture large aspects of the underlying images which allow for image reconstruction,—however there is still considerable room for improving the quality of reconstructed images. In particular, the implicit fingerprints exhibited strong blurring and other image artifacts. In comparison with other popular generative frameworks, we note that CFGenNets are different from Variational Autoencoders(VAEs) [67] or Generative Adversarial Networks (GANs) [72] where the original images are leveraged in the generative process. In this chapter, we compared the outcomes from CFGenNets with another generative network, Variational AutoEncoders [67]. We describe the details of the experiments conducted and the reasoning behind our choice of networks for comparison with CFGenNets in the next section.

#### 11.1 CFGenNets and Variational Autoencoders

Typically generative models such as Variational Autoencoders [67,134] are trained with a given dataset, and are used to generate data having similar properties as the samples in the dataset. VAEs learn the internal essence of the dataset and “store” all the information in the limited parameters(latent features) that are significantly smaller than the training dataset. Notwithstanding the inputs, we deem VAEs to be comparable to CFGenNets due to some commonalities. For instance the CFGenNet is essentially the same as the decoders in VAEs. The only difference being, in VAEs, the latent space is learnt from training and is constrained to resemble a standard normal distribution with mean of 0 and standard deviation of 1. While there are several variants of variational autoencoders introduced by researchers in the

field in the recent past, we limit our study to comparing the latent space encoded by “plain variational autoencoders” or often referred to as “vanilla variational autoencoders.” In the context of image reconstruction via plain VAEs, the network aims to minimize pixel-by-pixel mean square loss between the original image and the reconstructed image. We deem this as appropriate for comparative studies considering the same loss is leveraged for training CFGenNets, as described in the previous chapter. We note that all references to VAEs in the context of the experiments we describe in this thesis refer to the VAEs trained to minimize the per pixel reconstruction loss. In our experiments, we investigate a composite neural network that uses the VAE representation and implicit representation to construct an image. We theorize that the composite network can better use the implicit fingerprints for image attributes, while using the image representation to fine tune the generated images with information that is not captured by the implicit fingerprint.

Additionally we also trained a composite CFGenNet decoder that intakes the implicit image fingerprint from Collaborative filtering and the latent image vector obtained from variational autoencoders. Figure 11.1 illustrates the network architecture for composite CFGenNet decoders. As illustrated in the figure 11.1, the composite CFGenNet decoder takes two vector inputs, the latent vector obtained from Variational autoencoder (trained on the image data) and the Implicit image fingerprints obtained from collaborative filtering. These inputs feed in parallel to a series of fully connected layers and are averaged to obtain an “aggregated latent space” encoding information. The network is trained to minimize the mean square with the original images. The intuition behind the network is to augment the latent space obtained from variational autoencoders with the implicit image fingerprints from collaborative filtering to improve the reconstruction accuracy of the images. We also hypothesize that the network can separate its usage of the latent spaces in a way that general attributes are more influenced by the implicit fingerprints and other variables that are not encoded in the image attributes are “offloaded” to the VAE encoding. This dichotomy of latent representation encourages the model to learn disentangled latent variables. However, there is some redundancy between the implicit fingerprints and the VAE, so its unclear exactly how the latent space will be represented.

Figure 11.2 illustrates the technical metrics obtained by training the Variational autoen-

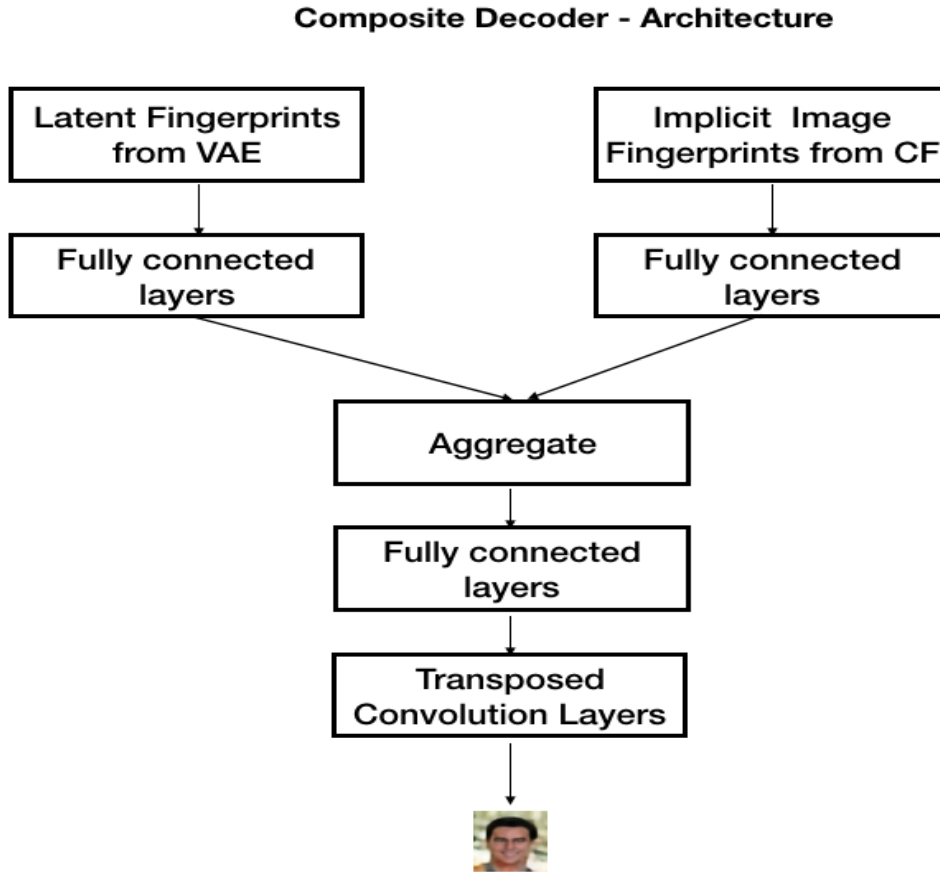


Figure 11.1: Composite CFGenNet decoders Architecture: These decoders consume the latent fingerprints from Variational Autoencoders and Collaborative Filtering as inputs. These are eventually averaged in the network to obtain the corresponding image.

coder and the composite CFGenNet decoders. The figure also plots metrics obtained by training the CFGenNets for purposes of comparison. As evidenced in the plot, the composite model trained with the encoding obtained from VAEs and collaborative filtering has the best performance out of the three models. We observe that the reconstruction accuracy for the CFGenNet model is at 75% and the plain VAE model at 81% while the composite CFGenNet decoder model demonstrated a reconstruction accuracy of 84%.

While the technical metrics indicate better outcomes in the composite CFGenNet decoder, we also visually inspected the outcomes to validate the results. Figure 11.3 illustrates a set of random images generated from each of the 3 networks presented side by side. We

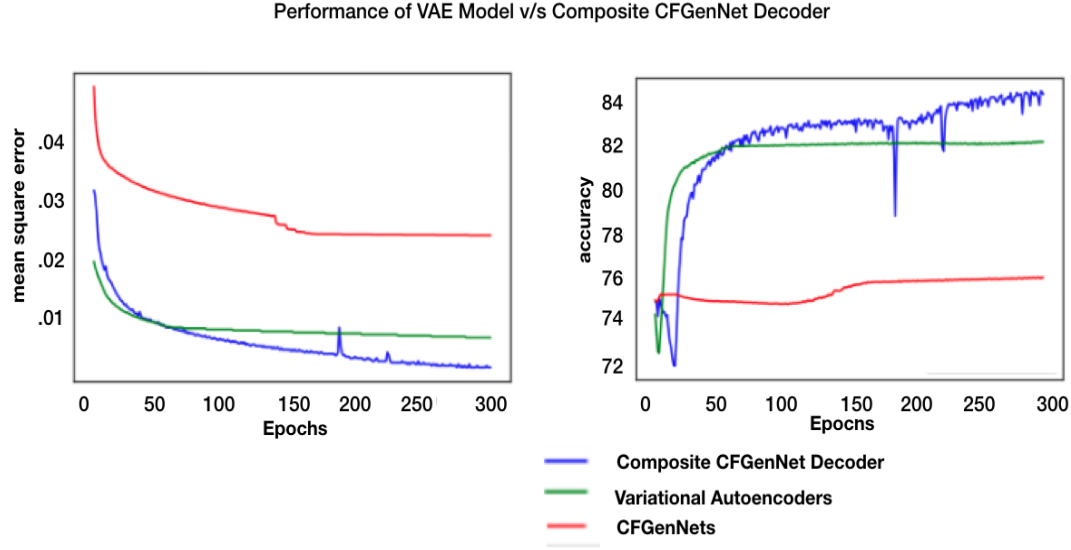


Figure 11.2: VAEs v/s vanilla CFGenNets v/s Composite CFGenNets: a random sample of images processed via 3 different types of networks. The reconstruction accuracy is highest for the Composite CFGenNet Decoders. They take advantage of the information encoded within the latent spaces generated by Variational Autoencoders and Collaborative Filtering

notice that the images generated by the composite decoder are a better quality compared with the other two models. While both VAEs and Vanilla CFGenNets generate clean images, the image details are blurred and in some cases slightly distorted, as similarly described in the previous chapter for CFGenNets. While the details and structures of images are preserved, the images are not as sharp as their input counterparts. However, we notice that in the case of the composite CFGenNet decoders, the images are comparatively sharper and more consistent. The faces also have much clearer noses, eyes, and other features, besides maintaining the background and foregrounds. As observed in figure 11.3, the images, we note that the VAEs maintain the image structures and the backgrounds reasonably well. This can be especially well perceived in the 2nd and the 5th images in the set of images on the left side. For example, the red background in 2nd image is well maintained in the VAE output, while the details of the red color are not fully captured by CFGenNets. However we notice that the details of the facial features considerable sharper in CFGenNets as opposed to VAEs. This phenomenon can again be observed in the 2nd image on the left most section in

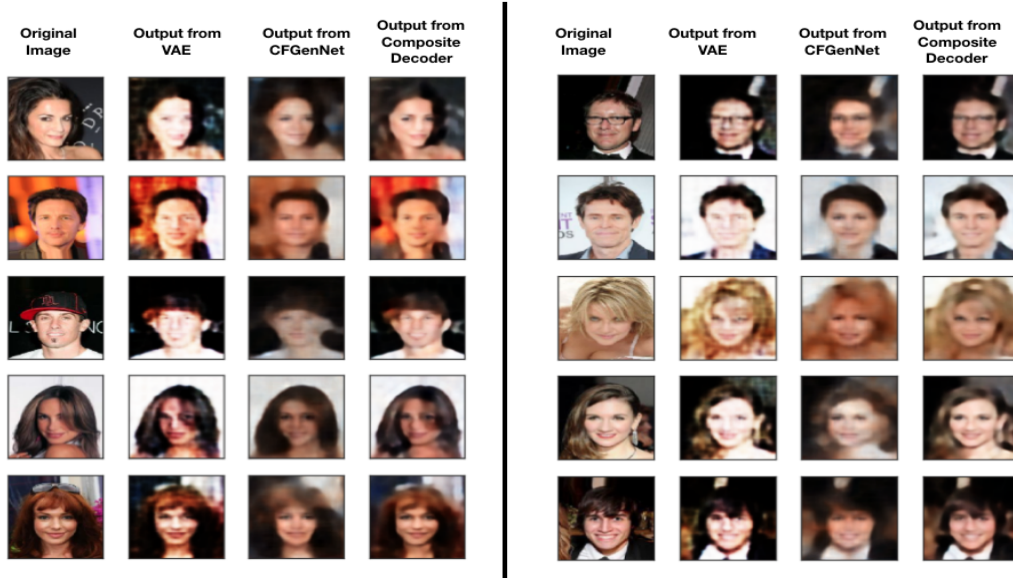


Figure 11.3: VAEs v/s vanilla CFGNets v/s Composite CFGNet Decoders : a random sample of images processed via 3 different types of networks. Visually, it is evident that the outcomes from the Composite CFGNet Decoders are more aligned to the original image then the other 2.

Figure 11.3. We notice that the output from the Composite CFGNets takes advantage of both these qualities from the VAEs and Vanilla CFGNets to produce better quality images than their individual counterparts. In summary, visual inspection reveals that combining the two latent representations is helpful and reduces many of the “generic” artifacts caused by the CFGNet model. Moreover, compared with the Vanilla VAEs, the combined model reduces many ringing artifacts around edges that would typically be observed in wavelet compression.

#### 11.1.1 Visualization of latent vectors

The results presented in figure 11.3 indicate better reconstruction in the images generated by the Composite CFGNet decoder. We further analyzed the latent representations from VAEs and Implicit fingerprints to validate if the better outcomes of the Composite Decoders is truly because of the complementary nature of the information contained in the two encodings. In this analysis, we visualized the images based on the similarity of their

latent representations. Here we define similarity by their L2-Distance in the latent space. We randomly chose 400 images from the data set and reduced both their latent encoding from VAEs and Implicit fingerprints into two dimensional space using the t-SNE [61] algorithm. The t-SNE method preserves higher dimensional space distances such that images having similar encoding in the higher dimensions are similar to one other in the two dimensional t-SNE space as well.

Figure 11.4 and Figure 11.5 illustrate the two sets of images. In each Figure, “more similar” images are positioned closely on a 2-D plane. We observe that in Figure 11.4, images having similar backgrounds (dark or light) tend to appear together. This is evident in the contrast in the images appearing the top left of Figure 11.4 versus the images appearing in the bottom right. Similarly images with similar poses appear together in smaller clusters. For example, the series of images in Figure 11.4 in the red box all contain images that are looking to the right, while the set of images in the purple box are all looking to the left. We can also see a presence of many images with raised hands among the images contained inside the purple box. From this perspective, the VAE encodings tend to be overpowered by background color and head pose.

We can contrast this representation with the images in Figure 11.5, where the images with similar descriptor properties appear together (due to the implicit fingerprints). These features tend to be more semantic to the face, rather than to the background and head pose. For example, the series of images under the red box in the first row contain images of older males with glasses, while the cluster of images enclosed in the green box contain an overwhelming majority of images with older looking males wearing ties. Similarly the images under the purple box contain images of females with blonde and straight hair. The contrast between the two fingerprints in terms of the images that appear together indicate that the VAE based encodings focus for a large part on the global features of the images such as backgrounds, poses, etc., while the implicit fingerprints encode attributes describing the features of the face contained in the image. This further supports our hypothesis that the two encoding methods are complementary with each other and, therefore, when combined help enhance the output image quality. Thus, we observe that Composite CFGenNet decoders might be a popular method of image understanding that bridges the advantages of



collaborative filtering with the less structured attributes in VAEs.

### 11.1.2 Conclusion

In this chapter, we introduced the concept of Composite CFGenNets Decoder which leveraged the latent encoding from Variational Autoencoders and Implicit Fingerprints from Collaborative Filtering. We observed the benefits in terms of better image reconstruction on Composite CFGenNet decoders. We also analyzed the two latent encodings to help explain the improved results from Composite CFGenNets. In the next chapter, we extend the Composite CFGenNets to perform image manipulation based on Implicit Feature Fingerprints.



Figure 11.4: Visualization of 20 x 20 face images based on VAE latent vectors by t-SNE algorithm



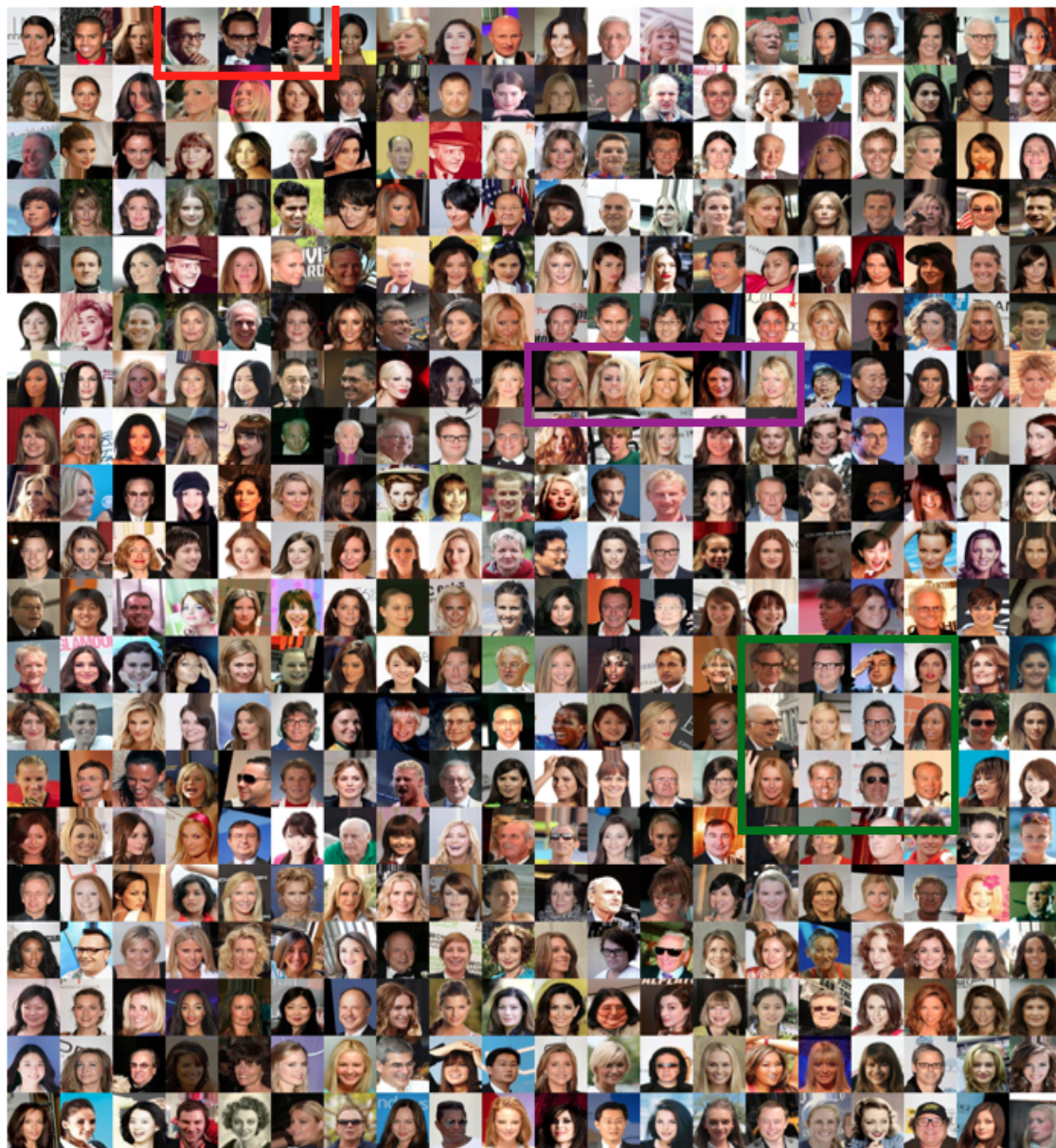


Figure 11.5: Visualization of 20 x 20 face images based on Implicit Image Fingerprints by t-SNE algorithm

## Chapter 12

### Image Attribute Manipulations using CFGenNets

One of the more interesting applications demonstrated with latent encoding both in the domain of image and language is the semantic manipulations performed in the latent space. For example [135] shows that  $\text{vector}(\text{"King"}) - \text{vector}(\text{"Man"}) + \text{vector}(\text{"Woman"})$  generates a vector whose nearest neighbor is the vector("Queen") when evaluating learned representation of words. On similar lines (in the domain of image processing), Radford et al. [102] demonstrated that visual concepts such as face pose and gender could be manipulated by vector arithmetic. Hou et al. [136] also demonstrated the abilities to perform linear interpolation between two latent vectors. They also demonstrated how the interpolation could yield effects of image manipulation. For example transitioning from an image with short hair to an image with long hair, "without glasses" to "with glasses". While the results can be compelling, these dimensions must be found through manual exploration of the latent space. That is, the dimensions of the VAE representation are uncontrolled and therefore two different optimizations of the same VAE architecture can result in completely different structured latent spaces. There have been several recent style based generative models that have demonstrated abilities to generate realistic yet synthetic images. For instance, recent result allow for the use of freeform text to manipulate the images. We discuss in detail some of the recent works in Neural style transfers under the Related Work section in Chapter 9. In our work, we attempt to perform image feature manipulation using the Implicit Feature Fingerprints. The subsequent sections describe in detail the steps undertaken.

#### 12.1 Attribute Manipulation using Composite CFGenNet Decoders

We further conducted analyses on the latent implicit fingerprints to evaluate the presence image manipulation properties. In Chapter 9, we evaluated the shared latent implicit fingerprint space, where we saw that our method produces two sets of implicit encodings

- the implicit Image fingerprints that encode the images and the Implicit Feature fingerprints that encode the 40 individual properties that describe each image. Chapter 9 Figure 9.5 further demonstrated the abilities of the implicit feature fingerprints could accurately predict the presence of the respective facial descriptor attributes when fed into the "facial attribute recognition classifier" trained on implicit image fingerprints. Given this context, we hypothesize that the Implicit Feature Fingerprints encode attribute information that could be visually perceived when fed into CFGenNets decoders. This sort of an operation could provide a mechanism to control the specific attribute of the facial images. We describe in the details of the experiments conducted as an algorithm for easy readability (Algorithm 1 below), in addition to discussing in detail the steps and the results.

---

**Algorithm 1:** Attribute Manipulation using Composite CFGenNets Decoder

---

```

Step 1 : Function TrainCompositeCFGenNetsDecoder;

CompositeDecoder = model.train(inputs=[LatentVAEFingerprints,
    ImplicitImageFingerprints], output = OriginalImages);



---



Step 2 : Manipulate an image to show ‘Wearing Lipstick’ ;

r = random_image_index;
 $L_{VAE}$  = LatentVAEFingerprints[r];
 $L_{imp}$  = ImplicitImageFingerprints[r];
feature_index = get_index_of('Wearing Lipstick');
 $L_{wearing\_lipstick}$  = ImplicitFeatureFingerprints[feature_index];
reconstructedImage = CompositeDecoder.predict(inputs=[ $L_{VAE}$ ,  $L_{imp}$  ]);
manipulatedImage = CompositeDecoder.predict(inputs=[ $L_{VAE}$ ,  $L_{wearing\_lipstick}$ 
    ]);

```

---

We hypothesize that the Composite CFGenNet Decoder, described in Chapter 11 provides a framework to affect such manipulations. In our experiment, we fed as inputs to the Composite CFGenNet, the latent encoding obtained from VAEs and the desired feature specific fingerprint to obtain manipulated images as the output. For example, we attempted to manipulate images to have the facial images “wearing lipstick”— i.e., we utilized the latent VAE fingerprints of the sample image and the Implicit Feature Fingerprint for the attribute

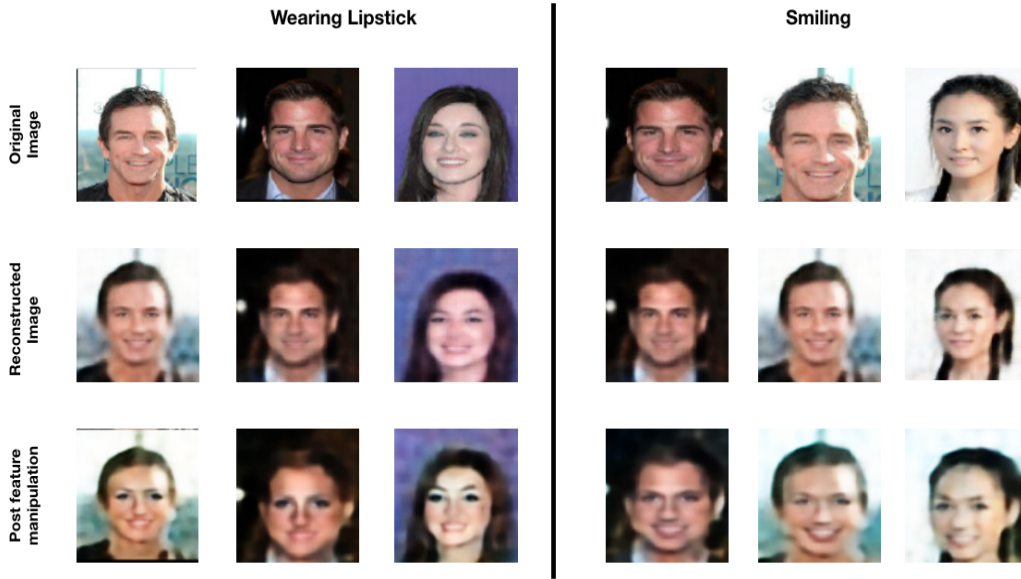


Figure 12.1: Image manipulation via Composite CFGenNet decoders : Images are generated by feeding the latent encoding obtained by VAEs and Implicit Feature fingerprint obtained from Collaborative Filtering. The first set of images are fused with the feature vector representing the attribute 'Wearing Lipstick' and the second set of images with the vector representing the attribute 'Smiling'.

“Wearing Lipstick” as the inputs to the Composite CFGenNet Decoder (Figure 11.1). This results in the input images undergoing the said manipulations. This is demonstrated in the samples presented in Figure 12.1. The first row in figure 12.1 represents the original image, while the second row represents the reconstructed image, without applying attribute manipulations. The third row represents the out images after having undergone the said manipulations. In Figure 12.1, we observe qualitatively that the output images from Composite CFGenNets are indeed manipulated when the Implicit Feature fingerprints are input along with the latent encodings from VAEs. We also observe that the degree of manipulation varies by the feature and the underlying image. Figure 12.1 illustrates a sample of images that have undergone feature manipulation based on our approach of using the Composite CFGenNet decoders. The two sets of images are contained in 12.1, the first set of images are fused with the implicit feature fingerprints representing the attribute “Wearing Lipstick”. We observe that the presence of lipstick is perceivable in the output images. In fact, we also



notice that the faces in the first 2 images begin to show feminine features along with the lipstick being perceivable. Additionally, we also observe that the faces seem to exhibit presence of makeup as well. These manipulations of the images to indicate presence of Makeup and Feminine features are consistent with the facial attribute recognition predictions obtained from Implicit Feature Fingerprints for the attribute “Wearing Lipstick” in Figure 9.5. Figure 9.5 shows that the attribute recognition classifier for the implicit feature fingerprints for “Wearing Lipstick”, also recognized the presence of the attribute “Heavy Makeup” and the absence of attribute “Male.” This is attributed to the fact that these features are highly correlated in the CelebA dataset. This follows some of the common conceptual relationships that exist between different facial attributes. As an example, we would assume bald and gray hair are associated to old people. Additionally, The technical correlated metrics measuring the correlated in CelebA the dataset have been evaluated and reported by Hou et al. [136]. It is not surprising that we do see effects of such correlated features in the first set of output images in figure 12.1. The second set of images in figure 12.1 have been fused with the fingerprints representing the attribute “Smiling.” We notice that the manipulated images in the second set have pronounced expressions indicating smiles. We also see that these images retained many attributes from their original counterparts. It could be explained by the fact that smiling seems to have no correlation with most of other attributes and is a very common human facial expression, especially in photographs. While we present limited examples in this section, Appendix 12.3 lists many more examples of image manipulations attempted via Composite CFGenNet decoders. In general, we observe that the manipulations result in changed images related to the attribute, but that some combinations of images and attribute manipulations results in visually anomalous artifacts that are difficult to describe. We therefore propose a competing architecture for attribute manipulation in the next section that also employs VAEs and implicit fingerprints.

## 12.2 Selective Attributes Manipulation using CFGenNets

The results obtained from the previous section validate that the Implicit Feature Fingerprints indeed contain visually perceivable encoding which, when processed via Composite CFGenNets, can be used to manipulate the images attributes. We akin each Implicit Feature

Fingerprint to a ‘style vector,’ encoding specific attribute properties. The composite CF-GenNets, however, only allow for attribute manipulations of one dimension at a time—i.e., the Composite CFGenNets decoder, as described in Figure 11.1, consumes a given image’s latent VAE fingerprint and an Implicit Image/Feature fingerprint obtained from Collaborative Filtering as its input. In the previous section, we demonstrated the outcomes of fusing the latent VAE fingerprints and the Implicit feature fingerprints to affect image attribute manipulations. However, the architecture supported manipulations at only one dimension at time, as in the case of fusing the image’s VAE fingerprint with either Implicit feature fingerprint the attribute “Wearing Lipstick” or “Smiling”, but not both concurrently. We note that it is possible to average the factors for “Wearing Lipstick” and “Smiling” but this results in images with highly visual and unnatural distortions.

We also observed that the manipulation using the previously proposed method may introduce *competing* representations. That is, when we manipulate the implicit fingerprint, the VAE fingerprint remains unchanged. Therefore, encodings that are redundantly described in the VAE and implicit fingerprint may cause unexpected artifacts. We attempt to address this constraint by introducing the Feature Gated Composite CFGenNets architecture, shown in Figure 12.2. The gating mechanism in the Gated Composite CFGenNets refers to the mechanism of controlling the exact features that can be fed into the CFGenNets at any given point in time. In this way, we can combine the VAE and original implicit fingerprints into an aggregated vector. Then, we use the gating mechanism of the network to manipulate the features. We hypothesize that this will mitigate competing representations when redundancy is present.

Figure 12.2 describes the architecture of the Feature Gated Composite CFGenNets. This is an enhancement over the Composite CFGenNets, where, in addition to the Latent VAE encodings and Implicit Image Fingerprints, the decoder also received a gated input from the Image Feature Maps and the Implicit Feature Fingerprints. The Image Feature Maps here refers to a binary vector of size 40, with one flag for each facial attribute that is contained in the image. The flag, when set, indicates that the attribute feature is present in the image and when unset indicates it is absent. The matrix multiplication operation between the implicit feature map and Implicit Feature Fingerprints serves as the gating mechanism. At



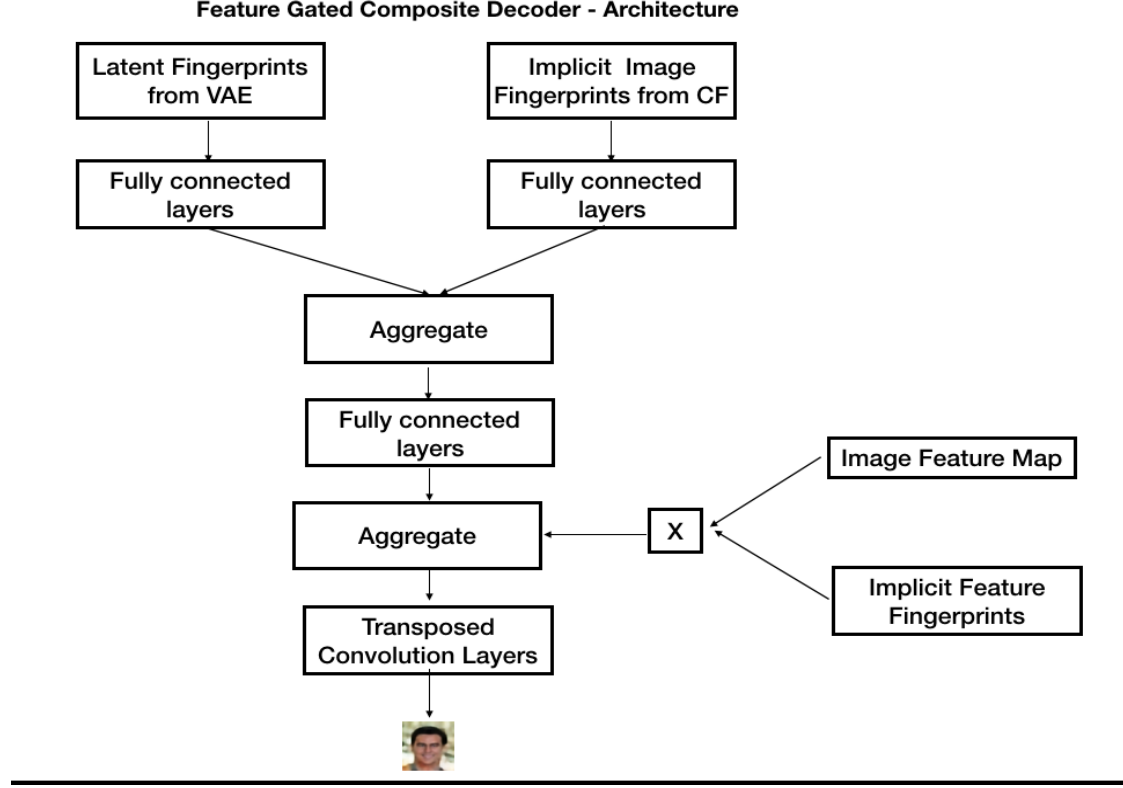


Figure 12.2: Feature Gated Composite CFGenNets : Architecture.

the time of training, the gating mechanism ensures that the Implicit Feature fingerprints of only the features present on the images are being leveraged to reconstruct the original image. Once trained, the gating mechanism provides a framework for controlling any set of features that can be concurrently manipulated on a given image. The pseudo code describing the manipulations using Gated Composite CFGenNets is described in Algorithm 2 below.

Figure 12.3 illustrates a sample set of images that have undergone image manipulation via Gated Composite CFGenNets. The single dimensional manipulation is achieved by setting only the desired feature to TRUE in the Image Feature Map input to the decoder. In Figure 12.3, three different input images have undergone various manipulations. In the first image, we apply the Implicit Fingerprint for “Wearing Hat.” We see that the output image produces what resembles a blue headgear. The same image when manipulated with the Implicit feature fingerprint for “Male” results in an image that has facial features with a more blocked jaw and wider lips. We observe appropriate modifications occurring in all

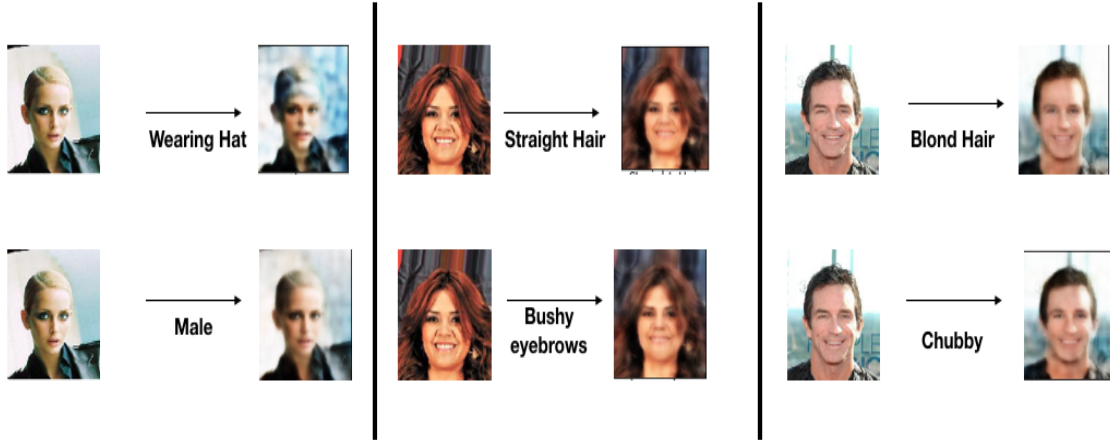


Figure 12.3: Feature Gated Composite CFGenNets: sample images with 1 dimensional feature manipulations

the images represented in Figure 12.3. While the quality of the image could be improved, we observe that the desired changes take effect on the output image in a visually consistent manner. Appendix 12.3 illustrates a other example images which have undergone single dimensional feature manipulations across all the 40 feature dimensions.

As mentioned previously, another advantage rendered by the Feature Gated Composite CFGenNets is the ability to affect more than one desired feature manipulation at a given time. This is illustrated in Figure 12.4, where multiple manipulations are performed concurrently. The first row in figure 12.4 represents the original image, while the second row represents the reconstructed image, without applying attribute manipulations. The third row represents the out images after having undergone the said manipulations.

The first two images of female celebrities are manipulated by gating in the Implicit Feature Fingerprints of the features “Male” and “Bushy Eyebrows.” We can clearly perceive these changes applied on the resultant images where the resultant faces in fact appear to have more traditionally masculine features with straighter jawlines and reduced eye makeup. Similarly, in the second set of images, the Implicit Feature Fingerprints corresponding to “Wearing Lipstick” and “Eye Glasses” are gated in. These features are evidently perceived in the resultant images where facial hair is reduced and the rims of eye glasses are also perceivable—although some ringing artifacts in the face and background are also introduced. While we are able

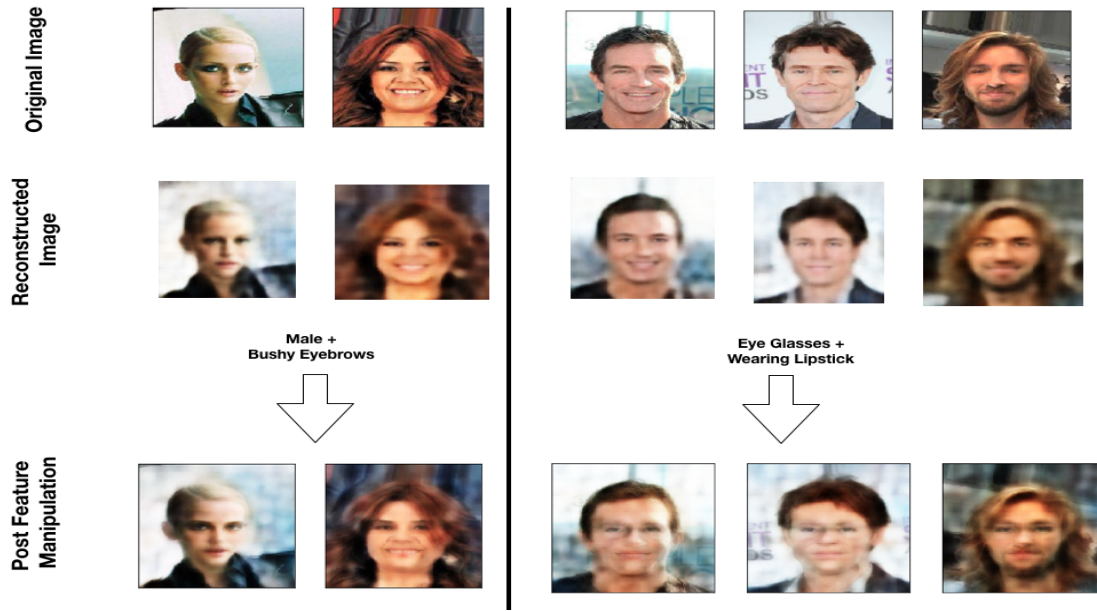


Figure 12.4: Feature Gated Composite CFGenNets : sample images with 2 dimensional feature manipulations

to perceive the desired manipulations on the resultant image, we acknowledge that there is more work to be done to improve the image resolutions from Composite CFGenNet decoders. We discuss these limitations in the subsequent section. Even so, these experiments support our hypothesis that combined collaborative filtering and other encodings (such as the VAE) are advantageous for research in image latent representations.

---

**Algorithm 2:** Attribute Manipulation using Gated CFGenNets Decoder

---

Step 1 : Function PrepareImageFeatureFingerprint;

ImageFeatureFingerprints = ImageFeatureMap X

ImplicitFeatureFingerprints;

where

$ImageFeatureMap \in R^{NumImages \times 40}$  &  $ImplicitFeatureFingerprint \in R^{40 \times 500}$

---

Step 2 : Function TrainGatedCFGenNetsDecoder;

GatedDecoder = model.train(inputs=[LatentVAEFingerprints,

ImplicitImageFingerprints, ImageFeatureFingerprints], output =

OriginalImages);

---

Step 3 : Manipulate an image to show 'Wearing Lipstick' & 'With Eyeglasses';

r = random\_image\_index;

$L_{VAE} = \text{LatentVAEFingerprints}[r]$ ;

$L_{ImpImageFP} = \text{ImplicitImageFingerprints}[r]$ ;

$L_{ImpImageFeatureFP} = \text{ImageFeatureFingerprints}[r]$ ;

CustomImageFeatureMap= init\_vector\_zeros(shape=(1,40));

CustomImageFeatureMap[index\_WearingLipstick]=True;

CustomImageFeatureMap[index\_WithEyeglasses]=True;

$L_{ManipFeatures} = \text{CustomImageFeatureMap} \times \text{ImplicitFeatureFingerprints}$ ;

reconstructedImage = GatedDecoder.predict(inputs=[ $L_{VAE}$ ,

$L_{ImpImageFP}, L_{ImpImageFeatureFP}$  ]);

manipulatedImage = GatedDecoder.predict(inputs=[ $L_{VAE}$ ,

$L_{ImpImageFP}, L_{ManipFeatures}$  ]);

---

### 12.3 Conclusion & Future work

Aim 3 of our research demonstrated the abilities of CFGenNets to be applied to the domain of Image Synthesis. The results discussed in Chapter 10 ascertain the fact that the

implicit fingerprints indeed capture patterns that help in reconstructing the original images. Chapter 11 further demonstrated the combined efficacy of latent encodings obtained from Variational autoencoders and Implicit Image Fingerprints. From the results presented above, it is evident that the combining the latent fingerprints obtained from VAEs and Collaborative filtering has positively influenced the outcome. In the case of Variational Autoencoders, the latent embeddings are obtained by learning to reconstruct the input image using an encoder/decoder combination. In the case of CFGenNets, we leverage the metadata of the images to obtain an encoding. This is different from VAEs in that the images are not used in the training. The collaborative filtering method leverages only the descriptors of the images. The approach learns the encoding from all the images and mutually similar/dissimilar properties represented in a matrix form. From the results presented above, we see that each of the methods VAEs and CFGenNets encode certain nuances of the underlying images. The enhanced performance of the Composite decoder indicates the advantages of combining these two encodings of the underlying images. We further demonstrated the flexibility offered by the Composite CFGenNet decoders to manipulate images by merely swapping the Implicit Image fingerprints with the Implicit Feature Fingerprints obtained from Collaborative Filtering.

While our work engenders the interpretability of the latent encodings obtained from collaborative filtering, we also acknowledge that there could be additional enhancements to CFGenNets architecture to improve the reconstruction accuracy of the images. As an example, in the current design, the process of generating implicit fingerprints via collaborative filtering and the generative networks to affect reconstruction are conducted as 2 separate but related steps. There is further scope to combine these 2 steps into a single multi-objective based neural network. This could further refine the latent encodings produced to facilitate improved reconstructions. We also note that the objective function of the CFGenNet decoders could be further enhanced on the lines of the advancements made in the objective functions of other works , such as mutual information minimization from Conditional Subspace VAE [106] or feature perceptual loss leveraged by Deep Feature Consistent Variational Autoencoder [136].

While we demonstrate a method of leveraging the Implicit Feature Fingerprints for in-

fluencing image manipulation, we also acknowledge that the current results demonstrated in this work could be further improved upon with better resolution decoders. We have largely stayed independent of the advancements made around Neural Style Transfers. We believe borrowing some of the underlying concepts driving style transfer literature such as Adaptive Instance normalization, specifically evaluating applicability of leveraging Implicit Feature Fingerprints as inputs for AdaIn, could provide further breakthroughs in image manipulation.

Finally, we believe with this work we have only scratched the surface of the exciting possibilities of leveraging the encoding generated by Collaborative Filtering by successfully demonstrating the cross domain applicability of our methods. We believe we have contributed a novel area of research to the community and believe the possibilities can be improved by the research community, where countless additions can be investigated.



## APPENDIX A : Sample Images generated from CFGenNets

This appendix sample images generated from the 2 variants for CFGenNets , one with Transpose Convolution layers and the one with Bi-linear Upsampling layers



Figure 12.5: Sample images generated from CFGenNets with Transpose Convolution layers and Bi-linear Upsampling layers

## APPENDIX B : Image Manipulation via Composite CFGNet Decoders

This appendix contains 2 randomly selected images whose latent encoding from Variational Autoencoders were processed with the Implicit feature fingerprints for each of the 40 features from the CelebA dataset. The original image and the outcomes when fused with each of the 40 features is illustrated in figure 12.6 and figure 12.7



Figure 12.6: Random Image 1 manipulated using Implicit Feature fingerprints via Composite CFGNet decoder





Figure 12.7: Random Image 2 manipulated using Implicit Feature fingerprints via Composite CFGNet decoder

## APPENDIX C : Image Manipulation via Feature Gated Composite CFGGenNet Decoders

This appendix contains 2 randomly selected images whose latent encoding from Variational Autoencoders were processed with the Implicit feature fingerprints for each of the 40 features from the CelebA dataset. The original image and the outcomes when fused with each of the 40 features is illustrated in figure 12.8 and figure 12.9

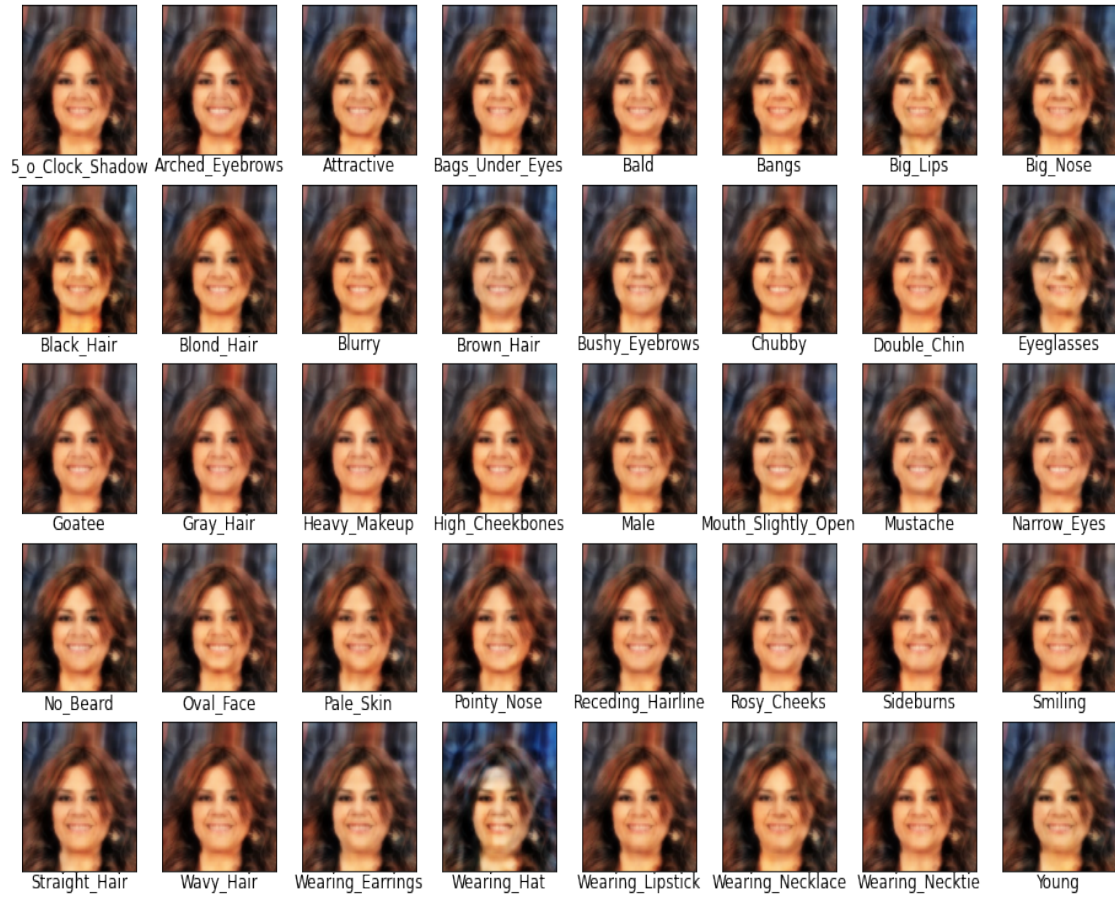


Figure 12.8: Random Image 1 manipulated using Implicit Feature fingerprints via Feature Gated Composite CFGGenNet decoder

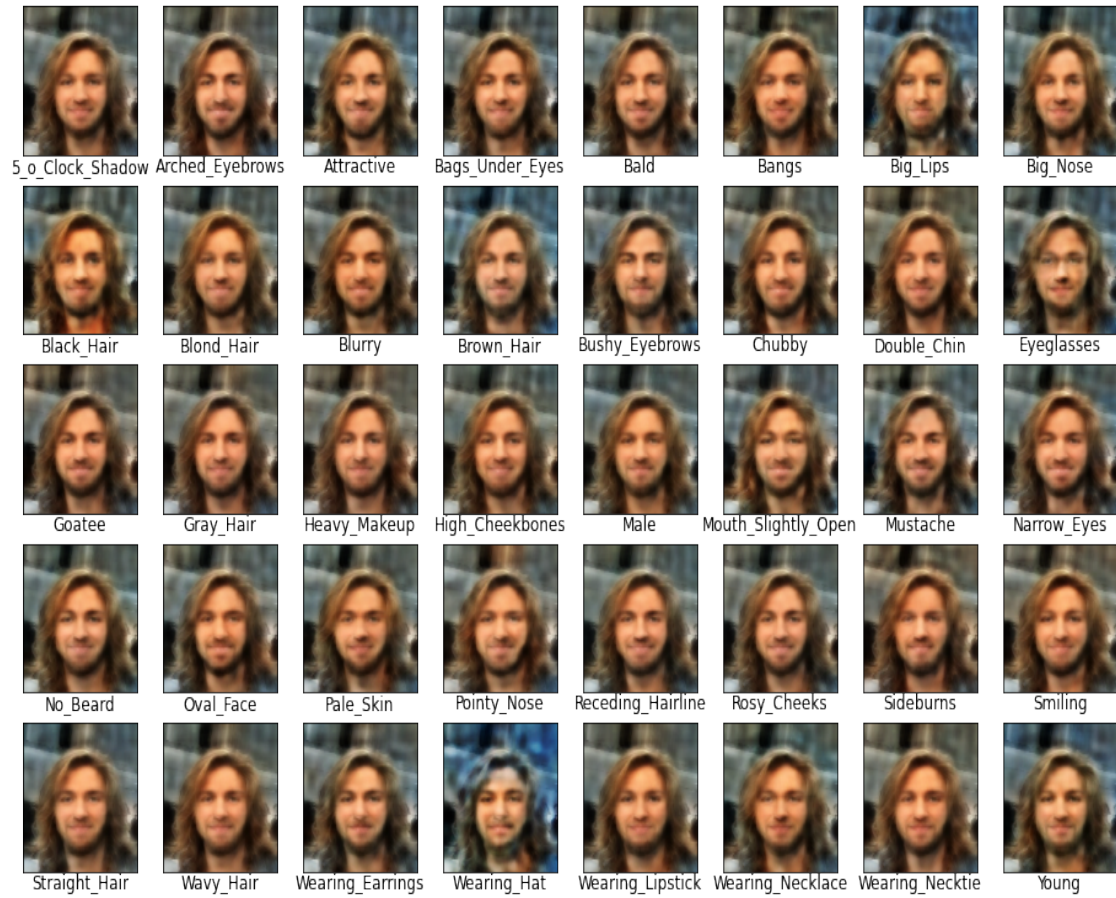


Figure 12.9: Random Image 2 manipulated using Implicit Feature fingerprints via Feature Gated Composite CFGenNet decoder

## BIBLIOGRAPHY

- [1] R. Srinivas, N. Verma, E. Kraka, and E. C. Larson, "Deep learning-based ligand design using shared latent implicit fingerprints from collaborative filtering," *Journal of Chemical Information and Modeling*, vol. 61, no. 5, pp. 2159–2174, 2021. [xiii](#), [12](#), [39](#), [54](#), [55](#), [60](#), [69](#), [70](#), [71](#)
- [2] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, "Automatic chemical design using a data-driven continuous representation of molecules," *ACS Cent. Sci.*, vol. 4, pp. 268–276, 2018. [xiii](#), [8](#), [41](#), [55](#), [70](#), [72](#)
- [3] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. [xiii](#), [78](#), [79](#)
- [4] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The adaptive web*. Springer, 2007, pp. 291–324. [1](#)
- [5] P. Melville, R. J. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations," *Aaai/iaai*, vol. 23, pp. 187–192, 2002. [1](#)
- [6] B. Kitts, D. Freed, and M. Vrieze, "Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditionally independent probabilities," in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2000, pp. 437–446. [1](#)
- [7] R. Khaleghi, K. Cannon, and R. Srinivas, "A comparative evaluation of recommender systems for hotel reviews," *SMU Data Science Review*, vol. 1, no. 4, p. 1, 2018. [1](#)
- [8] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009. [Online]. Available: <http://dx.doi.org/10.1109/MC.2009.263> [1](#), [15](#), [16](#), [25](#)
- [9] "Simon Funk SVD," <https://sifter.org/simon/journal/20061211.html>, [\[.\]](#). [2](#)
- [10] "What is a molecule," <https://www.britannica.com/science/molecule>, [\[.\]](#). [3](#)
- [11] "What is Virtual Screening," [https://en.wikipedia.org/wiki/Virtual\\_screening](https://en.wikipedia.org/wiki/Virtual_screening), [\[.\]](#). [4](#)
- [12] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of chemical information and computer sciences*, vol. 28, no. 1, pp. 31–36, 1988. [6](#), [7](#)



- [13] D. Bajusz, A. Rácz, and K. Héberger, “Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?” *Journal of Cheminformatics*, vol. 7, no. 1, p. 20, May 2015. [Online]. Available: <https://doi.org/10.1186/s13321-015-0069-3> 6, 25
- [14] G. Maggiora, M. Vogt, D. Stumpfe, and J. Bajorath, “Molecular similarity in medicinal chemistry,” *Journal of Medicinal Chemistry*, vol. 57, no. 8, pp. 3186–3204, 2014, pMID: 24151987. [Online]. Available: <http://dx.doi.org/10.1021/jm401411z> 6
- [15] R. Todeschini, V. Consonni, H. Xiang, J. Holliday, M. Buscema, and P. Willett, “Similarity coefficients for binary chemoinformatics data: Overview and extended comparison using simulated and real data sets,” *Journal of Chemical Information and Modeling*, vol. 52, no. 11, pp. 2884–2901, 2012, pMID: 23078167. [Online]. Available: <http://dx.doi.org/10.1021/ci300261r> 6
- [16] “RDKit: Open-source cheminformatics,” <http://www.rdkit.org>. 6, 54
- [17] S. Riniker and G. A. Landrum, “Open-source platform to benchmark fingerprints for ligand-based virtual screening,” *Journal of Cheminformatics*, vol. 5, no. 1, p. 26, May 2013. [Online]. Available: <https://doi.org/10.1186/1758-2946-5-26> 6, 23
- [18] I. Muegge and P. Mukherjee, “An overview of molecular fingerprint similarity search in virtual screening,” *Expert Opinion on Drug Discovery*, vol. 11, no. 2, pp. 137–148, 2016, pMID: 26558489. [Online]. Available: <http://dx.doi.org/10.1517/17460441.2016.1117070> 6
- [19] L. Jacob and J.-P. Vert, “Protein-ligand interaction prediction: An improved chemogenomics approach,” *Bioinformatics*, vol. 24, pp. 2149–2156, 2008. 7
- [20] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, “Prediction of drug-target interaction networks from the integration of chemical and genomic spaces,” *Bioinformatics*, vol. 24, pp. i232–i240, 2008. 7
- [21] I. Wallach, M. Dzamba, and A. Heifets, “Atomnet: A deep convolutional neural network for bioactivity prediction in structure-based drug discovery,” 2015, arXiv:1510.02855. 7
- [22] M. Tsubaki, K. Tomii, and J. Sese, “Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences,” *Bioinformatics*, vol. 35, pp. 309–318, 2018. 7
- [23] L. Li, C. C. Koh, D. Reker, J. B. Brown, H. Wang, N. K. Lee, H. haw Liow, H. Dai, H.-M. Fan, L. Chen, and D.-Q. Wei, “Predicting protein-ligand interactions based on bow-pharmacological space and bayesian additive regression trees,” *Sci. Rep.*, vol. 9, p. 7703, 2019. 7
- [24] N. Verma, X. Qu, F. Trozzi, M. Elsaied, Y. Tao, E. C. Larson, and E. Kraka, “SSnet - secondary structure based end-to-end learning model for protein-ligand interaction prediction,” *bioRxiv*, 2019. 7, 69

- [25] G. Schneider, “Virtual screening: an endless staircase?” *Nature Reviews Drug Discovery*, vol. 9, no. 4, pp. 273–276, 2010. 8
- [26] B. K. Shoichet, “Virtual screening of chemical libraries,” *Nature*, vol. 432, no. 7019, pp. 862–865, 2004. 8
- [27] A. Giordano, G. Forte, L. Massimo, R. Riccio, G. Bifulco, and S. Di Micco, “Discovery of new erbb4 inhibitors: repositioning an orphan chemical library by inverse virtual screening,” *European journal of medicinal chemistry*, vol. 152, pp. 253–263, 2018. 8
- [28] G. Lauro, A. Romano, R. Riccio, and G. Bifulco, “Inverse virtual screening of antitumor targets: pilot study on a small database of natural bioactive compounds,” *Journal of natural products*, vol. 74, no. 6, pp. 1401–1407, 2011. 8
- [29] X. Xu, M. Huang, and X. Zou, “Docking-based inverse virtual screening: methods, applications, and challenges,” *Biophysics reports*, vol. 4, no. 1, pp. 1–16, 2018. 8
- [30] “Drug Development Process,” <https://www.fda.gov/patients/learn-about-drug-and-device-approvals/drug-development-process>, []. 8
- [31] “Clinical Research,” <https://www.fda.gov/patients/drug-development-process/step-3-clinical-research>, []. 8
- [32] M. Olivecrona, T. Blaschke, O. Engkvist, and H. Chen, “Molecular de-novo design through deep reinforcement learning,” *J. Cheminf.*, vol. 9, no. 1, pp. 1–14, 2017. 8, 40, 55
- [33] J. Yasonik, “Multiobjective de novo drug design with recurrent neural networks and nondominated sorting,” *J. Cheminf.*, vol. 12, no. 1, pp. 1–9, 2020. 8, 55
- [34] M. Popova, O. Isayev, and A. Tropsha, “Deep reinforcement learning for de novo drug design,” *Science advances*, vol. 4, no. 7, p. eaap7885, 2018. 8, 55, 60
- [35] D. Erhan, P.-J. L’Heureux, S. Y. Yue, and Y. Bengio, “Collaborative filtering on a family of biological targets,” *Journal of Chemical Information and Modeling*, vol. 46, no. 2, pp. 626–635, 2006, pMID: 16562992. [Online]. Available: <http://dx.doi.org/10.1021/ci050367t> 9, 18
- [36] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, “Automatic chemical design using a data-driven continuous representation of molecules,” *ACS central science*, vol. 4, no. 2, pp. 268–276, 2018. 10, 41, 60, 70
- [37] —, “Automatic chemical design using a data-driven continuous representation of molecules,” *ACS central science*, vol. 4, no. 2, pp. 268–276, 2018. 10, 43, 44

- [38] T. Blaschke, M. Olivecrona, O. Engkvist, J. Bajorath, and H. Chen, "Application of generative autoencoder in de novo molecular design," *Molecular informatics*, vol. 37, no. 1-2, p. 1700123, 2018. [10](#), [42](#)
- [39] G. L. Guimaraes, B. Sanchez-Lengeling, C. Outeiral, P. L. C. Farias, and A. Aspuru-Guzik, "Objective-reinforced generative adversarial networks (organ) for sequence generation models," *arXiv preprint arXiv:1705.10843*, 2017. [10](#), [42](#)
- [40] O. Prykhodko, S. V. Johansson, P.-C. Kotsias, J. Arús-Pous, E. J. Bjerrum, O. Engkvist, and H. Chen, "A de novo molecular generation method using latent vector based generative adversarial network," *Journal of Cheminformatics*, vol. 11, no. 1, p. 74, 2019. [10](#), [42](#)
- [41] Y. Li, L. Zhang, and Z. Liu, "Multi-objective de novo drug design with conditional graph generative model," *Journal of cheminformatics*, vol. 10, no. 1, p. 33, 2018. [10](#), [41](#), [42](#)
- [42] W. Jin, R. Barzilay, and T. Jaakkola, "Junction tree variational autoencoder for molecular graph generation," *arXiv preprint arXiv:1802.04364*, 2018. [10](#), [41](#), [42](#)
- [43] R. Srinivas, P. V. Klimovich, and E. C. Larson, "Implicit-descriptor ligand-based virtual screening by means of collaborative filtering," *Journal of Cheminformatics*, vol. 10, no. 1, p. 56, 2018. [11](#), [15](#), [20](#), [39](#), [43](#), [45](#)
- [44] C. C. Aggarwal, *Neighborhood-Based Collaborative Filtering*. Cham: Springer International Publishing, 2016, pp. 29–70. [Online]. Available: [https://doi.org/10.1007/978-3-319-29659-3\\_2](https://doi.org/10.1007/978-3-319-29659-3_2) [15](#), [16](#)
- [45] Y. Koren and R. Bell, *Advances in Collaborative Filtering*. Boston, MA: Springer US, 2015, pp. 77–118. [Online]. Available: [https://doi.org/10.1007/978-1-4899-7637-6\\_3](https://doi.org/10.1007/978-1-4899-7637-6_3) [15](#)
- [46] G. H. Golub and C. Reinsch, *Singular Value Decomposition and Least Squares Solutions*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1971, pp. 134–151. [Online]. Available: <https://doi.org/10.1007/BF02163027> [17](#), [18](#)
- [47] L. Bottou, *Large-Scale Machine Learning with Stochastic Gradient Descent*. Heidelberg: Physica-Verlag HD, 2010, pp. 177–186. [Online]. Available: [https://doi.org/10.1007/978-3-7908-2604-3\\_16](https://doi.org/10.1007/978-3-7908-2604-3_16) [16](#), [18](#), [26](#), [27](#)
- [48] D. Reker, P. Schneider, G. Schneider, and Brown, "Active learning for computational chemogenomics," *Future Medicinal Chemistry*, vol. 9, no. 4, pp. 381–402, 2017, pMID: 28263088. [Online]. Available: <https://doi.org/10.4155/fmc-2016-0197> [20](#)
- [49] E. B. Lenselink, N. ten Dijke, B. Bongers, G. Papadatos, H. W. T. van Vlijmen, W. Kowalczyk, A. P. IJzerman, and G. J. P. van Westen, "Beyond the hype: deep neural networks outperform established methods using a chembl bioactivity benchmark set," *Journal of Cheminformatics*, vol. 9, no. 1, p. 45, Aug 2017. [Online]. Available: <https://doi.org/10.1186/s13321-017-0232-0> [20](#), [21](#)

- [50] I. Wallach and A. Heifets, “Most Ligand-Based Benchmarks Measure Overfitting Rather than Accuracy,” *ArXiv e-prints*, Jun. 2017. [21](#), [25](#), [26](#), [30](#)
- [51] D. Fourches, E. Muratov, and A. Tropsha, “Trust, but verify ii: A practical guide to chemogenomics data curation,” *Journal of Chemical Information and Modeling*, vol. 56, no. 7, pp. 1243–1252, 2016, pMID: 27280890. [Online]. Available: <http://dx.doi.org/10.1021/acs.jcim.6b00129> [21](#)
- [52] R. P. Sheridan, “The relative importance of domain applicability metrics for estimating prediction errors in qsar varies with training set diversity,” *Journal of Chemical Information and Modeling*, vol. 55, no. 6, pp. 1098–1107, 2015, pMID: 25998559. [Online]. Available: <http://dx.doi.org/10.1021/acs.jcim.5b00110> [21](#)
- [53] T. M. Mitchell, *Machine Learning*, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 1997. [21](#)
- [54] N. Triballeau, F. Acher, I. Brabet, J.-P. Pin, and H.-O. Bertrand, “Virtual screening workflow development guided by the “receiver operating characteristic” curve approach. application to high-throughput docking on metabotropic glutamate receptor subtype 4,” *Journal of Medicinal Chemistry*, vol. 48, no. 7, pp. 2534–2547, 2005, pMID: 15801843. [Online]. Available: <http://dx.doi.org/10.1021/jm049092j> [21](#)
- [55] D. A. Pearlman and P. S. Charifson, “Improved scoring of ligand-protein interactions using owfeg free energy grids,” *Journal of Medicinal Chemistry*, vol. 44, no. 4, pp. 502–511, 2001, pMID: 11170640. [Online]. Available: <http://dx.doi.org/10.1021/jm000375v> [22](#)
- [56] C. Empereur-mot, H. Guillemain, A. Latouche, J.-F. Zagury, V. Viallon, and M. Montes, “Predictiveness curves in virtual screening,” *Journal of Cheminformatics*, vol. 7, no. 1, p. 52, Nov 2015. [Online]. Available: <https://doi.org/10.1186/s13321-015-0100-8> [22](#)
- [57] R. P. Sheridan, S. B. Singh, E. M. Fluder, and S. K. Kearsley, “Protocols for bridging the peptide to nonpeptide gap in topological similarity searches,” *Journal of Chemical Information and Computer Sciences*, vol. 41, no. 5, pp. 1395–1406, 2001, pMID: 11604041. [Online]. Available: <http://dx.doi.org/10.1021/ci0100144> [23](#)
- [58] J.-F. Truchon and C. I. Bayly, “Evaluating virtual screening methods: Good and bad metrics for the early recognition problem,” *Journal of Chemical Information and Modeling*, vol. 47, no. 2, pp. 488–508, 2007, pMID: 17288412. [Online]. Available: <http://dx.doi.org/10.1021/ci600426e> [23](#)
- [59] Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, C. Guestrin, and J. Hellerstein, “Graphlab: A new framework for parallel machine learning,” in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, ser. UAI’10. Arlington, Virginia, United States: AUAI Press, 2010, pp. 340–349. [Online]. Available: <http://dl.acm.org/citation.cfm?id=3023549.3023589> [26](#)



- [60] S. C. Bull and A. J. Doig, "Properties of protein drug target classes," *PLoS One*, vol. 10, no. 3, p. e0117955, 2015. [34](#)
- [61] L. Maaten and G. Hinton, "Visualizing high-dimensional data using t-sne," vol. 9, pp. 2579–2605, Nov 2008. [34](#), [80](#), [99](#)
- [62] K. J. Higgins, S. Liu, M. Abdelrahim, K. Yoon, K. Vanderlaag, W. Porter, R. P. Metz, and S. Safe, "Vascular endothelial growth factor receptor-2 expression is induced by 17 $\beta$ -estradiol in zr-75 breast cancer cells by estrogen receptor  $\alpha$ /sp proteins," *Endocrinology*, vol. 147, no. 7, pp. 3285–3295, 2006. [36](#)
- [63] "Phf13 protein," pHF13 PHD finger protein 13 [ Homo sapiens (human) ]. [Online]. Available: <https://www.ncbi.nlm.nih.gov/gene/148479> [36](#)
- [64] I. Wallach and A. Heifets, "Most ligand-based benchmarks measure overfitting rather than accuracy," *arXiv preprint arXiv:1706.06619*, vol. 20, 2017. [40](#)
- [65] C. Voss, "Modeling molecules with recurrent neural networks," 2019. [40](#)
- [66] M. H. Segler, T. Kogej, C. Tyrchan, and M. P. Waller, "Generating focused molecule libraries for drug discovery with recurrent neural networks," *ACS central science*, vol. 4, no. 1, pp. 120–131, 2018. [40](#)
- [67] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013. [41](#), [47](#), [74](#), [94](#)
- [68] J. Lim, S. Ryu, J. W. Kim, and W. Y. Kim, "Molecular generative model based on conditional variational autoencoder for de novo molecular design," *Journal of cheminformatics*, vol. 10, no. 1, pp. 1–9, 2018. [41](#)
- [69] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato, "Grammar variational autoencoder," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1945–1954. [41](#)
- [70] J. Arús-Pous, T. Blaschke, S. Ulander, J.-L. Reymond, H. Chen, and O. Engkvist, "Exploring the gdb-13 chemical space using deep generative models," *Journal of cheminformatics*, vol. 11, no. 1, pp. 1–14, 2019. [41](#)
- [71] Y. Li, O. Vinyals, C. Dyer, R. Pascanu, and P. Battaglia, "Learning deep generative models of graphs," *arXiv preprint arXiv:1803.03324*, 2018. [41](#)
- [72] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014. [41](#), [42](#), [73](#), [76](#), [94](#)
- [73] B. Sanchez-Lengeling, C. Outeiral, G. L. Guimaraes, and A. Aspuru-Guzik, "Optimizing distributions over molecular space. an objective-reinforced generative adversarial network for inverse-design chemistry (organic)," *ChemRxiv*, vol. 2017, 2017. [42](#)

- [74] E. Putin, A. Asadulaev, Y. Ivanenkov, V. Aladinskiy, B. Sanchez-Lengeling, A. Aspuru-Guzik, and A. Zhavoronkov, "Reinforced adversarial neural computer for de novo molecular design," *Journal of chemical information and modeling*, vol. 58, no. 6, pp. 1194–1204, 2018. 42
- [75] E. Putin, A. Asadulaev, Q. Vanhaelen, Y. Ivanenkov, A. V. Aladinskaya, A. Aliper, and A. Zhavoronkov, "Adversarial threshold neural computer for molecular de novo design," *Molecular pharmaceutics*, vol. 15, no. 10, pp. 4386–4397, 2018. 42
- [76] P. Das, T. Sercu, K. Wadhawan, I. Padhi, S. Gehrmann, F. Cipcigan, V. Chenthamarakshan, H. Strobelt, C. Dos Santos, P.-Y. Chen *et al.*, "Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations," *Nature Biomedical Engineering*, pp. 1–11, 2021. 42
- [77] Q. Vanhaelen, Y.-C. Lin, and A. Zhavoronkov, "The advent of generative chemistry," *ACS Medicinal Chemistry Letters*, vol. 11, no. 8, pp. 1496–1505, 2020. 42
- [78] E. J. Bjerrum and B. Sattarov, "Improving chemical autoencoder latent space and molecular de novo generation diversity with heteroencoders," *Biomolecules*, vol. 8, p. 131, 2018. 42, 46
- [79] "ChEMBL23," <https://www.ebi.ac.uk/chembl/>, accessed: 2020-09-30. 43
- [80] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014. 43, 44
- [81] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," *arXiv preprint arXiv:1511.06349*, 2015. 44
- [82] Z. Tang, Y. Shi, D. Wang, Y. Feng, and S. Zhang, "Memory visualization for gated recurrent neural networks in speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2736–2740. 44
- [83] Y. Santur, "Sentiment analysis based on gated recurrent unit," in *2019 International Artificial Intelligence and Data Processing Symposium (IDAP)*. IEEE, 2019, pp. 1–5. 44
- [84] M. Zulqarnain, S. Ishak, R. Ghazali, N. M. Nawi, M. Aamir, and Y. M. M. Hassim, "An improved deep learning approach based on variant two-state gated recurrent unit and word embeddings for sentiment classification," *International Journal of Advanced Computer Science and Applications*, vol. 11, pp. 594–603, 2020. 44
- [85] D. Reker, P. Schneider, G. Schneider, and J. Brown, "Active learning for computational chemogenomics," *Future medicinal chemistry*, vol. 9, no. 4, pp. 381–402, 2017. 45

- [86] E. B. Lenselink, N. Dijke, B. Bongers, G. Papadatos, H. W. Vlijmen, W. Kowalczyk, A. P. IJzerman, and G. J. Westen, "Beyond the hype: deep neural networks outperform established methods using a chembl bioactivity benchmark set," *Journal of Cheminformatics*, vol. 9, no. 1, p. 45, 2017. [45](#)
- [87] G. Maggiora, M. Vogt, D. Stumpfe, and J. Bajorath, "Molecular similarity in medicinal chemistry," *J. Med. Chem.*, vol. 57, pp. 3186–3204, 2013. [56](#)
- [88] D. Rogers and M. Hahn, "Extended-connectivity fingerprints," *J. Chem. Inf. Model*, vol. 50, pp. 742–754, 2010. [56](#)
- [89] G. R. Bickerton, G. V. Paolini, J. Besnard, S. Muresan, and A. L. Hopkins, "Quantifying the chemical beauty of drugs," *Nature chemistry*, vol. 4, no. 2, pp. 90–98, 2012. [62](#)
- [90] S. A. Wildman and G. M. Crippen, "Prediction of physicochemical parameters by atomic contributions," *Journal of chemical information and computer sciences*, vol. 39, no. 5, pp. 868–873, 1999. [63](#)
- [91] P. Ertl and A. Schuffenhauer, "Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions," *Journal of cheminformatics*, vol. 1, no. 1, p. 8, 2009. [63](#)
- [92] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, pp. 50–60, 1947. [64](#)
- [93] C. A. Lipinski, "Lead- and drug-like compounds: the rule-of-five revolution," *Drug Discov. Today Technol.*, vol. 1, pp. 337–341, 2004. [64](#)
- [94] Z.-J. Yao, J. Dong, Y.-J. Che, M.-F. Zhu, M. Wen, N.-N. Wang, S. Wang, A.-P. Lu, and D.-S. Cao, "TargetNet: a web service for predicting potential drug–target interaction profiling via multi-target SAR models," *J. Comput. Aided Mol. Des.*, vol. 30, pp. 413–424, 2016. [65](#)
- [95] M. M. Mysinger, M. Carchia, J. J. Irwin, and B. K. Shoichet, "Directory of useful decoys, enhanced (DUD-e): Better ligands and decoys for better benchmarking," *J. Med. Chem.*, vol. 55, pp. 6582–6594, 2012. [65](#)
- [96] S. Nowozin, "Optimal decisions from probabilistic models: the intersection-over-union case," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 548–555. [65](#)
- [97] G. W. Bemis and M. A. Murcko, "The properties of known drugs. 1. molecular frameworks," *J. Med. Chem.*, vol. 39, no. 15, pp. 2887–2893, 1996. [67](#)
- [98] Y. Hu, D. Stumpfe, and J. Bajorath, "Lessons learned from molecular scaffold analysis," *J. Chem. Inf. Model.*, vol. 51, no. 8, pp. 1742–1753, 2011. [67](#)

- [99] —, “Computational exploration of molecular scaffolds in medicinal chemistry,” *J. Med. Chem.*, vol. 59, no. 9, pp. 4062–4076, 2016. [67](#)
- [100] D. R. Koes, M. P. Baumgartner, and C. J. Camacho, “Lessons learned in empirical scoring with smina from the CSAR 2011 benchmarking exercise,” *J. Chem. Inf. Model.*, vol. 53, pp. 1893–1904, 2013. [69](#)
- [101] E. Denton, S. Chintala, A. Szlam, and R. Fergus, “Deep generative image models using a laplacian pyramid of adversarial networks,” *arXiv preprint arXiv:1506.05751*, 2015. [73](#)
- [102] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” *arXiv preprint arXiv:1511.06434*, 2015. [73](#), [88](#), [103](#)
- [103] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative adversarial text to image synthesis,” in *International Conference on Machine Learning*. PMLR, 2016, pp. 1060–1069. [73](#)
- [104] S. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, “Learning what and where to draw,” *arXiv preprint arXiv:1610.02454*, 2016. [73](#)
- [105] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, “Zero-shot text-to-image generation,” *arXiv preprint arXiv:2102.12092*, 2021. [73](#)
- [106] J. Klys, J. Snell, and R. Zemel, “Learning latent subspaces in variational autoencoders,” *arXiv preprint arXiv:1812.06190*, 2018. [75](#), [112](#)
- [107] D. Li, M. Zhang, W. Chen, and G. Feng, “Facial attribute editing by latent space adversarial variational autoencoders,” in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 1337–1342. [75](#)
- [108] X. Yan, J. Yang, K. Sohn, and H. Lee, “Attribute2image: Conditional image generation from visual attributes,” in *European Conference on Computer Vision*. Springer, 2016, pp. 776–791. [75](#)
- [109] L. Cai, H. Gao, and S. Ji, “Multi-stage variational auto-encoders for coarse-to-fine image generation,” in *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 2019, pp. 630–638. [75](#), [89](#)
- [110] A. Razavi, A. v. d. Oord, and O. Vinyals, “Generating diverse high-fidelity images with vq-vae-2,” *arXiv preprint arXiv:1906.00446*, 2019. [75](#)
- [111] G. K. Wallace, “The jpeg still picture compression standard,” *IEEE transactions on consumer electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992. [75](#)
- [112] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Springer Science & Business Media, 2012, vol. 159. [75](#)

- [113] I. Goodfellow, “Nips 2016 tutorial: Generative adversarial networks,” *arXiv preprint arXiv:1701.00160*, 2016. 76
- [114] R. D. Hjelm, A. P. Jacob, T. Che, A. Trischler, K. Cho, and Y. Bengio, “Boundary-seeking generative adversarial networks,” *arXiv preprint arXiv:1702.08431*, 2017. 76
- [115] X. Mao, Q. Li, H. Xie, R. Y. Lau, and Z. Wang, “Multi-class generative adversarial networks with the l2 loss function,” *arXiv preprint arXiv:1611.04076*, vol. 5, p. 00102, 2016. 76
- [116] J. Zhao, L. Xiong, J. Karlekar, J. Li, F. Zhao, Z. Wang, S. Pranata, S. Shen, S. Yan, and J. Feng, “Dual-agent gans for photorealistic and identity preserving profile face synthesis.” in *NIPS*, vol. 2, 2017, p. 3. 76
- [117] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” *arXiv preprint arXiv:1411.1784*, 2014. 76
- [118] D. Berthelot, T. Schumm, and L. Metz, “Began: Boundary equilibrium generative adversarial networks,” *arXiv preprint arXiv:1703.10717*, 2017. 76
- [119] L. A. Gatys, A. S. Ecker, and M. Bethge, “Image style transfer using convolutional neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423. 76
- [120] X. Huang and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1501–1510. 76
- [121] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410. 77
- [122] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, “Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5907–5915. 77
- [123] H. Wang, J. D. Williams, and S. Kang, “Learning to globally edit images with textual description,” *arXiv preprint arXiv:1810.05786*, 2018. 77
- [124] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, “Adversarial autoencoders,” *arXiv preprint arXiv:1511.05644*, 2015. 77
- [125] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov, “Generating images from captions with attention,” *arXiv preprint arXiv:1511.02793*, 2015. 77
- [126] J. Zhi, “Pixelbrush: Art generation from text with gans,” in *Cl. Proj. Stanford CS231N Convolutional Neural Networks Vis. Recognition, Sprint 2017*, 2017, p. 256. 77

- [127] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” *arXiv preprint arXiv:1603.07285*, 2016. 88
- [128] “Bilinear Upsampling,” [https://en.wikipedia.org/wiki/Bilinear\\_interpolation](https://en.wikipedia.org/wiki/Bilinear_interpolation), []. 88
- [129] T. Scherr, K. Streule, A. Bartschat, M. Böhlend, J. Stegmaier, M. Reischl, V. Orian-Rousseau, and R. Mikut, “Beadnet: deep learning-based bead detection and counting in low-resolution microscopy images,” *Bioinformatics*, vol. 36, no. 17, pp. 4668–4670, 2020. 88
- [130] C. W. Tseng, H.-R. Su, S.-H. Lai, and J. Liu, “Depth image super-resolution via multi-frame registration and deep learning,” in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–8. 88
- [131] L. Fang, F. Monroe, S. W. Novak, L. Kirk, C. R. Schiavon, B. Y. Seungyoon, T. Zhang, M. Wu, K. Kastner, Y. Kubota *et al.*, “Deep learning-based point-scanning super-resolution imaging,” *BioRxiv*, p. 740548, 2019. 88
- [132] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, “Understanding convolution for semantic segmentation,” in *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2018, pp. 1451–1460. 88
- [133] X. Hou, K. Sun, L. Shen, and G. Qiu, “Improving variational autoencoder with deep feature consistent and generative adversarial training,” *Neurocomputing*, vol. 341, pp. 183–194, 2019. 90
- [134] J. C. Leachtenauer, W. Malila, J. Irvine, L. Colburn, and N. Salvaggio, “General image-quality equation: Giqe,” *Applied optics*, vol. 36, no. 32, pp. 8322–8328, 1997. 94
- [135] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *arXiv preprint arXiv:1310.4546*, 2013. 103
- [136] X. Hou, L. Shen, K. Sun, and G. Qiu, “Deep feature consistent variational autoencoder,” in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 1133–1141. 103, 106, 112
- [137] R. E. Carhart, D. H. Smith, and R. Venkataraghavan, “Atom pairs as molecular features in structure-activity studies: definition and applications,” *Journal of Chemical Information and Computer Sciences*, vol. 25, no. 2, pp. 64–73, 1985.
- [138] R. Nilakantan, N. Bauman, J. S. Dixon, and R. Venkataraghavan, “Topological torsion: a new molecular descriptor for sar applications. comparison with other descriptors,” *Journal of Chemical Information and Computer Sciences*, vol. 27, no. 2, pp. 82–85, 1987. [Online]. Available: <http://pubs.acs.org/doi/abs/10.1021/ci00054a008>



- [139] M. Sastry, J. F. Lowrie, S. L. Dixon, and W. Sherman, "Large-scale systematic analysis of 2d fingerprint methods and parameters to improve virtual screening enrichments," *Journal of Chemical Information and Modeling*, vol. 50, no. 5, pp. 771–784, 2010, pMID: 20450209. [Online]. Available: <http://dx.doi.org/10.1021/ci100062n>
- [140] N. M. O’Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch, and G. R. Hutchison, "Open babel: An open chemical toolbox," *J. Cheminformatics*, vol. 3, 2011.
- [141] "PubChem Substructure Fingerprint Description," [ftp://ftp.ncbi.nlm.nih.gov/pubchem/specifications/pubchem\\_fingerprints.txt](ftp://ftp.ncbi.nlm.nih.gov/pubchem/specifications/pubchem_fingerprints.txt), [].
- [142] N. M. O’Boyle and R. A. Sayle, "Comparing structural fingerprints using a literature-based similarity benchmark," *Journal of Cheminformatics*, vol. 8, no. 1, p. 36, Jul 2016. [Online]. Available: <https://doi.org/10.1186/s13321-016-0148-0>
- [143] J. Hert, P. Willett, D. J. Wilton, P. Acklin, K. Azzaoui, E. Jacoby, and A. Schuffenhauer, "Comparison of topological descriptors for similarity-based virtual screening using multiple bioactive reference structures," *Org. Biomol. Chem.*, vol. 2, pp. 3256–3266, 2004. [Online]. Available: <http://dx.doi.org/10.1039/B409865J>
- [144] A. Bender, H. Y. Mussa, R. C. Glen, and S. Reiling, "Similarity searching of chemical databases using atom environment descriptors (molprint 2d): evaluation of performance," *Journal of chemical information and computer sciences*, vol. 44, no. 5, pp. 1708–1718, 2004.
- [145] G. J. van Westen, J. K. Wegner, A. P. IJzerman, H. W. van Vlijmen, and A. Bender, "Proteochemometric modeling as a tool to design selective compounds and for extrapolating to novel targets," *MedChemComm*, vol. 2, no. 1, pp. 16–30, 2011.
- [146] H. Yuan, I. Paskov, H. Paskov, A. J. González, and C. S. Leslie, "Multitask learning improves prediction of cancer drug sensitivity," *Scientific reports*, vol. 6, 2016.
- [147] G. E. Dahl, N. Jaitly, and R. Salakhutdinov, "Multi-task neural networks for qsar predictions," *arXiv preprint arXiv:1406.1231*, 2014.
- [148] B. R. Beno and J. S. Mason, "The design of combinatorial libraries using properties and 3d pharmacophore fingerprints," *Drug Discovery Today*, vol. 6, no. 5, pp. 251–258, 2001.
- [149] A. Bender, H. Y. Mussa, R. C. Glen, and S. Reiling, "Molecular similarity searching using atom environments, information-based feature selection, and a naive bayesian classifier," *Journal of chemical information and computer sciences*, vol. 44, no. 1, pp. 170–178, 2004.
- [150] E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant, "Pubchem: integrated platform of small molecules and biological activities," *Annual reports in computational chemistry*, vol. 4, pp. 217–241, 2008.

- [151] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse, "Reoptimization of mdl keys for use in drug discovery," *Journal of chemical information and computer sciences*, vol. 42, no. 6, pp. 1273–1280, 2002.
- [152] J. L. Melville, E. K. Burke, and J. D. Hirst, "Machine learning in virtual screening," *Combinatorial chemistry & high throughput screening*, vol. 12, no. 4, pp. 332–343, 2009.
- [153] D. Joseph-McCarthy, "Computational approaches to structure-based ligand design," *Pharmacology & therapeutics*, vol. 84, no. 2, pp. 179–191, 1999.
- [154] T. Liu, Y. Lin, X. Wen, R. N. Jorissen, and M. K. Gilson, "Bindingdb: a web-accessible database of experimentally determined protein–ligand binding affinities," *Nucleic acids research*, vol. 35, no. suppl\_1, pp. D198–D201, 2006.
- [155] P. D. Lyne, "Structure-based virtual screening: an overview," *Drug discovery today*, vol. 7, no. 20, pp. 1047–1055, 2002.
- [156] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1, pp. 37 – 52, 1987, proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0169743987800849>
- [157] B. Merget, S. Turk, S. Eid, F. Rippmann, and S. Fulle, "Profiling prediction of kinase inhibitors: Toward the virtual assay," *Journal of Medicinal Chemistry*, vol. 60, no. 1, pp. 474–485, 2017, pMID: 27966949. [Online]. Available: <http://dx.doi.org/10.1021/acs.jmedchem.6b01611>
- [158] A. M. Afzal, H. Y. Mussa, R. E. Turner, A. Bender, and R. C. Glen, "A multi-label approach to target prediction taking ligand promiscuity into account," *Journal of Cheminformatics*, vol. 7, no. 1, p. 24, May 2015. [Online]. Available: <https://doi.org/10.1186/s13321-015-0071-9>
- [159] H. Y. Mussa, D. Marcus, J. B. O. Mitchell, and R. C. Glen, "Verifying the fully "laplacianised" posterior naïve bayesian approach and more," *Journal of Cheminformatics*, vol. 7, no. 1, p. 27, Jun 2015. [Online]. Available: <https://doi.org/10.1186/s13321-015-0075-5>
- [160] J. Balfer and J. Bajorath, "Introduction of a methodology for visualization and graphical interpretation of bayesian classification models," *Journal of Chemical Information and Modeling*, vol. 54, no. 9, pp. 2451–2468, 2014, pMID: 25137527. [Online]. Available: <http://dx.doi.org/10.1021/ci500410g>
- [161] B. Chen, R. P. Sheridan, V. Hornak, and J. H. Voigt, "Comparison of random forest and pipeline pilot naïve bayes in prospective qsar predictions," *Journal of Chemical Information and Modeling*, vol. 52, no. 3, pp. 792–803, 2012, pMID: 22360769. [Online]. Available: <http://dx.doi.org/10.1021/ci200615h>



- [162] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, "Deep neural nets as a method for quantitative structure-activity relationships," *Journal of Chemical Information and Modeling*, vol. 55, no. 2, pp. 263–274, 2015, pMID: 25635324. [Online]. Available: <http://dx.doi.org/10.1021/ci500747n>
- [163] R. P. Sheridan, W. M. Wang, A. Liaw, J. Ma, and E. M. Gifford, "Extreme gradient boosting as a method for quantitative structure-activity relationships," *Journal of Chemical Information and Modeling*, vol. 56, no. 12, pp. 2353–2360, 2016, pMID: 27958738. [Online]. Available: <http://dx.doi.org/10.1021/acs.jcim.6b00591>
- [164] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random forest: A classification and regression tool for compound classification and qsar modeling," *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 6, pp. 1947–1958, 2003, pMID: 14632445. [Online]. Available: <http://dx.doi.org/10.1021/ci034160g>
- [165] R. L. Marchese Robinson, A. Palczewska, J. Palczewski, and N. Kidley, "Comparison of the predictive performance and interpretability of random forest and linear models on benchmark data sets," *Journal of Chemical Information and Modeling*, vol. 57, no. 8, pp. 1773–1792, 2017, pMID: 28715209. [Online]. Available: <http://dx.doi.org/10.1021/acs.jcim.6b00753>
- [166] K. Ullrich, M. Kamp, T. Gärtner, M. Vogt, and S. Wrobel, "Ligand-based virtual screening with co-regularised support vector regression," in *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, Dec 2016, pp. 261–268.
- [167] N. Sugaya, "Ligand efficiency-based support vector regression models for predicting bioactivities of ligands to drug target proteins," *Journal of Chemical Information and Modeling*, vol. 54, no. 10, pp. 2751–2763, 2014, pMID: 25220713. [Online]. Available: <http://dx.doi.org/10.1021/ci5003262>
- [168] S. E. Rensi and R. B. Altman, "Shallow representation learning via kernel pca improves qsar modelability," *Journal of Chemical Information and Modeling*, vol. 57, no. 8, pp. 1859–1867, 2017, pMID: 28727421. [Online]. Available: <http://dx.doi.org/10.1021/acs.jcim.6b00694>
- [169] K. Heikamp and J. Bajorath, "Support vector machines for drug discovery," *Expert Opinion on Drug Discovery*, vol. 9, no. 1, pp. 93–104, 2014, pMID: 24304044. [Online]. Available: <http://dx.doi.org/10.1517/17460441.2014.866943>
- [170] M. Duran-Frigola, D. Rossell, and P. Aloy, "A chemo-centric view of human health and disease," vol. 5, p. 5676, Dec 2014, article. [Online]. Available: <http://dx.doi.org/10.1038/ncomms6676>
- [171] M. Luo, X. S. Wang, and A. Tropsha, "Comparative analysis of qsar-based vs. chemical similarity based predictors of gpcrs binding affinity," *Molecular Informatics*,

- vol. 35, no. 1, pp. 36–41, 2016. [Online]. Available: <http://dx.doi.org/10.1002/minf.201500038>
- [172] Y. Wang, Y. Guo, Q. Kuang, X. Pu, Y. Ji, Z. Zhang, and M. Li, “A comparative study of family-specific protein–ligand complex affinity prediction based on random forest approach,” *Journal of Computer-Aided Molecular Design*, vol. 29, no. 4, pp. 349–360, Apr 2015. [Online]. Available: <https://doi.org/10.1007/s10822-014-9827-y>
- [173] C. Bendtsen, A. Degasperi, E. Ahlberg, and L. Carlsson, “Improving machine learning in early drug discovery,” *Annals of Mathematics and Artificial Intelligence*, vol. 81, no. 1, pp. 155–166, Oct 2017. [Online]. Available: <https://doi.org/10.1007/s10472-017-9541-2>
- [174] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [175] F. Chollet *et al.*, “Keras,” <https://github.com/fchollet/keras>, 2015.
- [176] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [177] A. Gaulton, A. Hersey, M. Nowotka, A. P. Bento, J. Chambers, D. Mendez, P. Mutowo, F. Atkinson, L. J. Bellis, E. Cibrián-Uhalte, M. Davies, N. Dedman, A. Karlsson, M. P. Magariños, J. P. Overington, G. Papadatos, I. Smit, and A. R. Leach, “The chembl database in 2017,” *Nucleic Acids Research*, vol. 45, no. 1, p. 10, Jan 2017. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5210557/>
- [178] B. Ramsundar, S. Kearnes, P. Riley, D. Webster, D. Konerding, and V. Pande, “Massively Multitask Networks for Drug Discovery,” *ArXiv e-prints*, Feb. 2015.
- [179] “Front matter,” in *Artificial Neural Network for Drug Design, Delivery and Disposition*, M. Puri, Y. Pathak, V. K. Sutariya, S. Tipparaju, and W. Moreno, Eds. Boston: Academic Press, 2016, pp. iii –. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/B9780128015599010012>
- [180] C. Empereur-Mot, J.-F. Zagury, and M. Montes, “Screening explorer—an interactive tool for the analysis of screening results,” *Journal of Chemical Information and*

- Modeling*, vol. 56, no. 12, pp. 2281–2286, 2016, pMID: 27808512. [Online]. Available: <http://dx.doi.org/10.1021/acs.jcim.6b00283>
- [181] H. B. Rao, F. Zhu, G. B. Yang, Z. R. Li, and Y. Z. Chen, “Update of profeat: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence,” *Nucleic Acids Research*, vol. 39, no. suppl2, pp. W385–W390, 2011. [Online]. Available: [+http://dx.doi.org/10.1093/nar/gkr284](http://dx.doi.org/10.1093/nar/gkr284)
- [182] Z. R. Li, H. H. Lin, L. Y. Han, L. Jiang, X. Chen, and Y. Z. Chen, “Profeat: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence,” *Nucleic Acids Research*, vol. 34, no. suppl2, pp. W32–W37, 2006. [Online]. Available: [+http://dx.doi.org/10.1093/nar/gkl305](http://dx.doi.org/10.1093/nar/gkl305)
- [183] T. Unterthiner, A. Mayr, G. unter Klambauer, M. Steijaert, J. Wenger, H. Ceulemans, and S. Hochreiter, “Deep learning as an opportunity in virtual screening,” in *Deep Learning and Representation Learning Workshop (NIPS 2014)*, 2014.
- [184] D. Rogers and M. Hahn, “Extended-connectivity fingerprints,” *Journal of Chemical Information and Modeling*, vol. 50, no. 5, pp. 742–754, 2010, pMID: 20426451. [Online]. Available: <http://dx.doi.org/10.1021/ci100050t>
- [185] S. J. Swamidass, C.-A. Azencott, T.-W. Lin, H. Gramajo, S.-C. Tsai, and P. Baldi, “Influence relevance voting: An accurate and interpretable virtual high throughput screening method,” *Journal of Chemical Information and Modeling*, vol. 49, no. 4, pp. 756–766, 2009, pMID: 19391629. [Online]. Available: <http://dx.doi.org/10.1021/ci8004379>
- [186] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “Grouplens: An open architecture for collaborative filtering of netnews,” in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW ’94. New York, NY, USA: ACM, 1994, pp. 175–186. [Online]. Available: <http://doi.acm.org/10.1145/192844.192905>
- [187] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, “Using collaborative filtering to weave an information tapestry,” *Commun. ACM*, vol. 35, no. 12, pp. 61–70, Dec. 1992. [Online]. Available: <http://doi.acm.org/10.1145/138859.138867>
- [188] S. P. Leelananda and S. Lindert, “Computational methods in drug discovery,” *Beilstein Journal of Organic Chemistry*, vol. 12, pp. 2694–2718, 2016.
- [189] D. Rognan, “The impact of in silico screening in the discovery of novel and safer drug candidates,” *Pharmacology & Therapeutics*, vol. 175, no. Supplement C, pp. 47 – 66, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0163725817300487>
- [190] J. B. O. Mitchell, “Machine learning methods in chemoinformatics,” *Wiley Interdisciplinary Reviews: Computational Molecular Science*, vol. 4, no. 5, pp. 468–481, 2014. [Online]. Available: <http://dx.doi.org/10.1002/wcms.1183>

- [191] H. van Vlijmen, R. L. Desjarlais, and T. Mirzadegan, "Computational chemistry at janssen," *Journal of Computer-Aided Molecular Design*, vol. 31, no. 3, pp. 267–273, Mar 2017. [Online]. Available: <https://doi.org/10.1007/s10822-016-9998-9>
- [192] V. Tsui, D. F. Ortwine, and J. M. Blaney, "Enabling drug discovery project decisions with integrated computational chemistry and informatics," *Journal of Computer-Aided Molecular Design*, vol. 31, no. 3, pp. 287–291, Mar 2017. [Online]. Available: <https://doi.org/10.1007/s10822-016-9988-y>
- [193] J. Hughes, S. Rees, S. Kalindjian, and K. Philpott, "Principles of early drug discovery," *British Journal of Pharmacology*, vol. 162, no. 6, pp. 1239–1249, 2011. [Online]. Available: <http://dx.doi.org/10.1111/j.1476-5381.2010.01127.x>
- [194] I. Muegge, A. Bergner, and J. M. Kriegl, "Computer-aided drug design at boehringer ingelheim," *Journal of Computer-Aided Molecular Design*, vol. 31, no. 3, pp. 275–285, Mar 2017. [Online]. Available: <https://doi.org/10.1007/s10822-016-9975-3>
- [195] D. B. Kitchen, "Computer-aided drug discovery research at a global contract research organization," *Journal of Computer-Aided Molecular Design*, vol. 31, no. 3, pp. 309–318, Mar 2017. [Online]. Available: <https://doi.org/10.1007/s10822-016-9991-3>
- [196] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "Moleculenet: a benchmark for molecular machine learning," *Chemical science*, vol. 9, no. 2, pp. 513–530, 2018.
- [197] W. Zheng and A. Tropsha, "Novel variable selection quantitative structure- property relationship approach based on the k-nearest-neighbor principle," *Journal of chemical information and computer sciences*, vol. 40, no. 1, pp. 185–194, 2000.
- [198] P. M. Petrone, B. Simms, F. Nigsch, E. Lounkine, P. Kutchukian, A. Cornett, Z. Deng, J. W. Davies, J. L. Jenkins, and M. Glick, "Rethinking molecular similarity: comparing compounds on the basis of biological activity," *ACS chemical biology*, vol. 7, no. 8, pp. 1399–1409, 2012.
- [199] D. Rognan, "The impact of in silico screening in the discovery of novel and safer drug candidates," *Pharmacology & therapeutics*, vol. 175, pp. 47–66, 2017.
- [200] V. Tsui, D. F. Ortwine, and J. M. Blaney, "Enabling drug discovery project decisions with integrated computational chemistry and informatics," *Journal of computer-aided molecular design*, vol. 31, no. 3, pp. 287–291, 2017.
- [201] D. B. Kitchen, "Computer-aided drug discovery research at a global contract research organization," *Journal of computer-aided molecular design*, vol. 31, no. 3, pp. 309–318, 2017.
- [202] I. Muegge, A. Bergner, and J. M. Kriegl, "Computer-aided drug design at boehringer ingelheim," *Journal of computer-aided molecular design*, vol. 31, no. 3, pp. 275–285, 2017.

- [203] H. van Vlijmen, R. L. Desjarlais, and T. Mirzadegan, "Computational chemistry at janssen," *Journal of computer-aided molecular design*, vol. 31, no. 3, pp. 267–273, 2017.
- [204] H. Song, R. Wang, S. Wang, and J. Lin, "A low-molecular-weight compound discovered through virtual database screening inhibits stat3 function in breast cancer cells," *Proceedings of the National Academy of Sciences*, vol. 102, no. 13, pp. 4700–4705, 2005.
- [205] B. Allen, S. Mehta, N. Ayad, and S. Schürer, "Ligand-and structure-based virtual screening to discover dual egfr and brd4 inhibitors," 2015.
- [206] A. Munir, S. Azam, A. Mehmood, Z. Khan, A. Mehmood, and S. Aqdas, "Structure-based pharmacophore modeling, virtual screening and molecular docking for the treatment of esr1 mutations in breast cancer," *Drug Des*, vol. 5, no. 3, p. 1000137, 2016.
- [207] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.
- [208] C. C. Aggarwal, "Neighborhood-based collaborative filtering," in *Recommender systems*. Springer, 2016, pp. 29–70.
- [209] D. Erhan, P.-J. L'Heureux, S. Y. Yue, and Y. Bengio, "Collaborative filtering on a family of biological targets," *Journal of chemical information and modeling*, vol. 46, no. 2, pp. 626–635, 2006.
- [210] G. E. Dahl, N. Jaitly, and R. Salakhutdinov, "Multi-task neural networks for qsar predictions," *arXiv preprint arXiv:1406.1231*, 2014.
- [211] C. Empereur-Mot, H. Guillemain, A. Latouche, J.-F. Zagury, V. Viallon, and M. Montes, "Predictiveness curves in virtual screening," *Journal of cheminformatics*, vol. 7, no. 1, p. 52, 2015.
- [212] J.-F. Truchon and C. I. Bayly, "Evaluating virtual screening methods: good and bad metrics for the "early recognition" problem," *Journal of chemical information and modeling*, vol. 47, no. 2, pp. 488–508, 2007.
- [213] N. Triballeau, F. Acher, I. Brabet, J.-P. Pin, and H.-O. Bertrand, "Virtual screening workflow development guided by the "receiver operating characteristic" curve approach. application to high-throughput docking on metabotropic glutamate receptor subtype 4," *Journal of medicinal chemistry*, vol. 48, no. 7, pp. 2534–2547, 2005.
- [214] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

- [215] C. D. Abernethy, G. M. Codd, M. D. Spicer, and M. K. Taylor, "A highly stable N-heterocyclic carbene complex of trichloro-oxo-vanadium(V) displaying novel Cl—C(carbene) bonding interactions," *J. Am. Chem. Soc.*, vol. 125, no. 5, pp. 1128–1129, 2003.
- [216] [Online]. Available: <http://pubs.acs.org/books/references.shtml>
- [217] A. J. Arduengo, III, H. V. R. Dias, R. L. Harlow, and M. Kline, "Electronic stabilization of nucleophilic carbenes," *J. Am. Chem. Soc.*, vol. 114, no. 14, pp. 5530–5534, 1992.
- [218] A. J. Arduengo, III, S. F. Gamper, J. C. Calabrese, and F. Davidson, "Low-coordinate carbene complexes of nickel(0) and platinum(0)," vol. 116, no. 10, pp. 4391–4394, 1994.
- [219] L. N. Appelhans, D. Zuccaccia, A. Kovacevic, A. R. Chianese, J. R. Miecznikowski, A. Macchioni, E. Clot, O. Eisenstein, and R. H. Crabtree, "An anion-dependent switch in selectivity results from a change of C—H activation mechanism in the reaction of an imidazolium salt with IrH5(PPh3)2," *J. Am. Chem. Soc.*, vol. 127, no. 46, pp. 16 299–16 311, 2005.
- [220] A. M. Coghill and L. R. Garson, Eds., *The ACS Style Guide*, 3rd ed. New York: Oxford University Press, Inc. and The American Chemical Society, 2006.
- [221] F. A. Cotton, G. Wilkinson, C. A. Murillio, and M. Bochmann, *Advanced Inorganic Chemistry*, 6th ed. Chichester, United Kingdom: Wiley, 1999.
- [222] "Communication from the european commission to the european council and the european parliament: 20 20 by 2020: Europe's climate change opportunity," Brussels, Belgium, Tech. Rep., 2008.
- [223] H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, and T. Blaschke, "The rise of deep learning in drug discovery," *Drug discovery today*, vol. 23, no. 6, pp. 1241–1250, 2018.
- [224] E. Friedman-Hill, *Writing Rules in Jess*, 1st ed. Greenwich, CT, USA: Manning Publications Co., 2003.
- [225] A. L. Johnson, "1-(alkylsubstituted phenyl)imidazoles useful in acth reverse assay," US Patent 3 637 731, 1972.
- [226] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, Montgomery, Jr., J. A., T. Vreven, K. N. Kudin, J. C. Burant, J. M. Millam, S. S. Iyengar, J. Tomasi, V. Barone, B. Mennucci, M. Cossi, G. Scalmani, N. Rega, G. A. Petersson, H. Nakatsuji, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, M. Klene, X. Li, J. E. Knox, H. P. Hratchian, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, P. Y. Ayala, K. Morokuma, G. A. Voth, P. Salvador, J. J.

- Dannenberg, V. G. Zakrzewski, S. Dapprich, A. D. Daniels, M. C. Strain, O. Farkas, D. K. Malick, A. D. Rabuck, K. Raghavachari, J. B. Foresman, J. V. Ortiz, Q. Cui, A. G. Baboul, S. Clifford, J. Cioslowski, B. B. Stefanov, G. Liu, A. Liashenko, P. Piskorz, I. Komaromi, R. L. Martin, D. J. Fox, T. Keith, M. A. Al-Laham, C. Y. Peng, A. Nanayakkara, M. Challacombe, P. M. W. Gill, B. Johnson, W. Chen, M. W. Wong, C. Gonzalez, and J. A. Pople, *Gaussian 03*, Gaussian, Inc., Wallingford, CT, USA, Gaussian, Inc., Wallingford, CT, 2004.
- [227] A. Abarca, P. Gómez-Sal, A. Martín, M. Mena, J. M. Poblet, and C. Yélamos, “Ammonolysis of mono(pentamethylcyclopentadienyl) titanium(IV) derivatives,” *Inorg. Chem.*, vol. 39, no. 4, pp. 642–651, 2000.
- [228] Z. Guo and D. Cremer, “Methods for a Rapid and Automated Description of Proteins,” in *Rev. Comp. Chem.*, K. Lipkowitz and D. Boyd, Eds. John Wiley & Sons, New York, 2016, pp. 369–438.
- [229] D. Cremer and E. Kraka, “From molecular vibrations to bonding, chemical reactions, and reaction mechanism,” *Curr. Org. Chem.*, vol. 14, pp. 1524–1560, 2010.
- [230] D. Chen, N. Oezguen, P. Urvil, C. Ferguson, S. M. Dann, and T. C. Savidge, “Regulation of protein-ligand binding affinity by hydrogen bond pairing,” *Science Advances*, vol. 2, p. e1501240, 2016.
- [231] Y. Itoh, Y. Nakashima, S. Tsukamoto, T. Kurohara, M. Suzuki, Y. Sakae, M. Oda, Y. Okamoto, and T. Suzuki, “N+-C-H...O hydrogen bonds in protein-ligand complexes,” *Scientific Reports*, vol. 9, 2019, doi: 10.1038/s41598-018-36987-9.
- [232] W. Zhou, H. Yan, and Q. Hao, “Analysis of surface structures of hydrogen bonding in protein–ligand interactions using the alpha shape model,” *Chem. Phys. Lett.*, vol. 545, pp. 125–131, 2012.
- [233] M. A. Williams and J. E. Ladbury, *Hydrogen Bonds in Protein-Ligand Complexes*. John Wiley & Sons, Ltd, 2005, ch. 6, pp. 137–161.
- [234] S. K. Panigrahi, “Strong and weak hydrogen bonds in protein-ligand complexes of kinases: a comparative study,” *Amino Acids*, vol. 34, pp. 617–633, 2008.
- [235] K. Kumar, S. M. Woo, T. Siu, W. A. Cortopassi, F. Duarte, and R. S. Paton, “Cation– $\pi$  interactions in protein–ligand binding: Theory and data-mining reveal different roles for lysine and arginine,” *Chem. Sci.*, vol. 9, pp. 2655–2665, 2018.
- [236] M. Brylinski, “Aromatic interactions at the ligand-protein interface: Implications for the development of docking scoring functions,” *Chemical Biology & Drug Design*, vol. 91, pp. 380–390, 2017.
- [237] J. P. Gallivan and D. A. Dougherty, “Cation- $\pi$  interactions in structural biology,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 96, pp. 9459–9464, 1999.



- [238] R. Patil, S. Das, A. Stanley, L. Yadav, A. Sudhakar, and A. K. Varma, "Optimized hydrophobic interactions and hydrogen bonding at the target-ligand interface leads the pathways of drug-designing," *PLoS ONE*, vol. 5, p. e12029, 2010.
- [239] C. E. Eyers, M. Vonderach, S. Ferries, K. Jeacock, and P. A. Eyers, "Understanding protein–drug interactions using ion mobility–mass spectrometry," *Curr. Opin. Chem. Biol.*, vol. 42, pp. 167–176, 2018.
- [240] H. Zhou and A. Sharma, "Therapeutic protein-drug interactions: plausible mechanisms and assessment strategies," *Expert Opin. Drug Metab. Toxicol*, vol. 12, pp. 1323–1331, 2016.
- [241] G. M. West, C. L. Tucker, T. Xu, S. K. Park, X. Han, J. R. Yates, and M. C. Fitzgerald, "Quantitative proteomics approach for identifying protein-drug interactions in complex mixtures using protein stability measurements," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 107, pp. 9078–9082, 2010.
- [242] D. B. Kitchen, H. Decornez, J. R. Furr, and J. Bajorath, "Docking and scoring in virtual screening for drug discovery: Methods and applications," *Nat. Rev. Drug Discovery*, vol. 3, pp. 935–949, 2004.
- [243] J. Hughes, S. Rees, S. Kalindjian, and K. Philpott, "Principles of early drug discovery," *Br. J. Pharmacol.*, vol. 162, pp. 1239–1249, 2011.
- [244] G. Sliwoski, S. Kothiwale, J. Meiler, and E. W. Lowe, "Computational methods in drug discovery," *Pharmacol. Rev.*, vol. 66, pp. 334–395, 2013.
- [245] P. D. Lyne, "Structure-based virtual screening: An overview," *Drug Discovery Today*, vol. 7, pp. 1047–1055, 2002.
- [246] Y.-C. Chen, "Beware of docking!" *Trends Pharmacol. Sci.*, vol. 36, pp. 78–95, 2015.
- [247] N. S. Pagadala, K. Syed, and J. Tuszynski, "Software for molecular docking: A review," *Biophys. Rev.*, vol. 9, pp. 91–102, 2017.
- [248] A. Lavecchia, "Machine-learning approaches in drug discovery: methods and applications," *Drug Discov. Today*, vol. 20, pp. 318–331, 2015.
- [249] Y.-C. Lo, S. E. Rensi, W. Torng, and R. B. Altman, "Machine learning in chemoinformatics and drug discovery," *Drug Discov. Today*, vol. 23, pp. 1538–1546, 2018.
- [250] A. N. Lima, E. A. Philot, G. H. G. Trossini, L. P. B. Scott, V. G. Maltarollo, and K. M. Honorio, "Use of machine learning approaches for novel drug discovery," *Expert Opin. Drug Discovery*, vol. 11, pp. 225–239, 2016.
- [251] L. Zhang, J. Tan, D. Han, and H. Zhu, "From machine learning to deep learning: progress in machine intelligence for rational drug discovery," *Drug Discov. Today*, vol. 22, pp. 1680–1685, 2017.



- [252] H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, and T. Blaschke, “The rise of deep learning in drug discovery,” *Drug Discov. Today*, vol. 23, pp. 1241–1250, 2018.
- [253] F. Ghasemi, A. Mehridehnavi, A. Pérez-Garrido, and H. Pérez-Sánchez, “Neural network and deep-learning algorithms used in QSAR studies: merits and drawbacks,” *Drug Discov. Today*, vol. 23, pp. 1784–1790, 2018.
- [254] I. I. Baskin, D. Winkler, and I. V. Tetko, “A renaissance of neural networks in drug discovery,” *Expert Opin. Drug Discovery*, vol. 11, pp. 785–795, 2016.
- [255] A. S. Rifaioğlu, H. Atas, M. J. Martin, R. Cetin-Atalay, V. Atalay, and T. Doğan, “Recent applications of deep learning and machine intelligence on in silico drug discovery: methods, tools and databases,” *Brief. Bioinform.*, pp. 1–36, 2018.
- [256] J. Panteleev, H. Gao, and L. Jia, “Recent applications of machine learning in medicinal chemistry,” *Bioorg. Med. Chem. Lett.*, vol. 28, pp. 2807–2815, 2018.
- [257] Y. Jing, Y. Bian, Z. Hu, L. Wang, and X.-Q. S. Xie, “Deep learning for drug design: an artificial intelligence paradigm for drug discovery in the big data era,” *The AAPS Journal*, vol. 20, p. 58, 2018.
- [258] G. Hessler and K.-H. Baringhaus, “Artificial intelligence in drug design,” *Molecules*, vol. 23, p. 2520, 2018.
- [259] D. Dana, S. Gadhiya, L. S. Surin, D. Li, F. Naaz, Q. Ali, L. Paka, M. Yamin, M. Narayan, I. Goldberg, and P. Narayan, “Deep learning in drug discovery and medicine; scratching the surface,” *Molecules*, vol. 23, p. 2384, 2018.
- [260] C. Shen, J. Ding, Z. Wang, D. Cao, X. Ding, and T. Hou, “From machine learning to deep learning: Advances in scoring functions for protein–ligand docking,” *WIREs Comput Mol Sci*, p. e1429, 2019.
- [261] M. Bredel and E. Jacoby, “Chemogenomics: An emerging strategy for rapid target and drug discovery,” *Nat. Rev. Genet.*, vol. 5, pp. 262–275, 2004.
- [262] S. Zheng, Y. Li, S. Chen, J. Xu, and Y. Yang, “Predicting drug protein interaction using quasi-visual question answering system,” 2019, doi: 10.1101/588178.
- [263] I. Lee, J. Keum, and H. Nam, “DeepConv-DTI: Prediction of drug-target interactions via deep learning with convolution on protein sequences,” *PLoS Comput. Biol.*, vol. 15, p. e1007129, 2019.
- [264] S. Ranganathan, D. Izotov, E. Kraka, and D. Cremer, “Description and recognition of regular and distorted secondary structures in proteins using the automated protein structure analysis method,” *Proteins*, vol. 76, pp. 418–438, 2009.
- [265] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” 2013, arXiv:1301.3781.

- [266] Y. Goldberg and O. Levy, “word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method,” 2014, arXiv:1402.3722.
- [267] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” 2013, arXiv:1310.4546.
- [268] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” *IEEE Trans. Neural Netw.*, vol. 20, pp. 61–80, 2009.
- [269] S. Ekins, A. C. Puhl, K. M. Zorn, T. R. Lane, D. P. Russo, J. J. Klein, A. J. Hickey, and A. M. Clark, “Exploiting machine learning for end-to-end drug discovery and development,” *Nat. Mater.*, vol. 18, pp. 435–441, 2019.
- [270] A. Murzin, “Scop: A structural classification of proteins database for the investigation of sequences and structures,” *J. Mol. Biol.*, vol. 247, pp. 536–540, 1995.
- [271] S. Dietmann and L. Holm, “Identification of homology in protein structure classification,” *Nat. Struct. Biol.*, vol. 8, pp. 953–957, 2001.
- [272] C. Orengo, A. Michie, S. Jones, D. Jones, M. Swindells, and J. Thornton, “CATH – a hierarchic classification of protein domain structures,” *Structure*, vol. 5, pp. 1093–1109, 1997.
- [273] J. Martin, G. Letellier, A. Marin, J.-F. Taly, A. G. de Brevern, and J.-F. Gibrat, “Protein secondary structure assignment revisited: A detailed analysis of different assignment methods,” *BMC Struct. Biol.*, vol. 5, p. 17, 2005.
- [274] W. Kabsch and C. Sander, “Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features,” *Biopolymers*, vol. 22, pp. 2577–2637, 1983.
- [275] D. Frishman and P. Argos, “Knowledge-based protein secondary structure assignment,” *Proteins*, vol. 23, pp. 566–579, 1995.
- [276] F. M. Richards and C. E. Kundrot, “Identification of structural motifs from protein coordinate data: Secondary structure and first-level supersecondary structure,” *Proteins*, vol. 3, pp. 71–84, 1988.
- [277] R. Day, D. A. Beck, R. S. Armen, and V. Daggett, “A consensus view of fold space: Combining SCOP, CATH, and the dali domain dictionary,” *Protein Sci.*, vol. 12, pp. 2150–2160, 2003.
- [278] C. M. Venkatachalam, “Stereochemical criteria for polypeptides and proteins. v. conformation of a system of three linked peptide units,” *Biopolymers*, vol. 6, pp. 1425–1436, 1968.

- [279] N. Triballeau, F. Acher, I. Brabet, J.-P. Pin, and H.-O. Bertrand, "Virtual screening workflow development guided by the "receiver operating characteristic" curve approach. application to high-throughput docking on metabotropic glutamate receptor subtype 4," *J. Med. Chem.*, vol. 48, pp. 2534–2547, 2005.
- [280] D. A. Pearlman and P. S. Charifson, "Improved scoring of ligand-protein interactions using OWFEG free energy grids," *J. Med. Chem.*, vol. 44, pp. 502–511, 2001.
- [281] C. Empereur-mot, H. Guillemain, A. Latouche, J.-F. Zagury, V. Viallon, and M. Montes, "Predictiveness curves in virtual screening," *J. Cheminf.*, vol. 7, p. 52, 2015.
- [282] K. Bleakley and Y. Yamanishi, "Supervised prediction of drug–target interactions using bipartite local models," *Bioinformatics*, vol. 25, pp. 2397–2403, 2009.
- [283] T. van Laarhoven, S. B. Nabuurs, and E. Marchiori, "Gaussian interaction profile kernels for predicting drug–target interaction," *Bioinformatics*, vol. 27, pp. 3036–3043, 2011.
- [284] M. Gonen, "Predicting drug-target interactions from chemical and genomic kernels using bayesian matrix factorization," *Bioinformatics*, vol. 28, pp. 2304–2310, 2012.
- [285] M. Hamanaka, K. Taneishi, H. Iwata, J. Ye, J. Pei, J. Hou, and Y. Okuno, "CGBVS-DNN: Prediction of compound-protein interactions based on deep learning," *Mol. Inf.*, vol. 36, p. 1600045, 2016.
- [286] K. Tian, M. Shao, Y. Wang, J. Guan, and S. Zhou, "Boosting compound-protein interaction prediction by deep learning," *Methods*, vol. 110, pp. 64–72, 2016.
- [287] H. Liu, J. Sun, J. Guan, J. Zheng, and S. Zhou, "Improving compound–protein interaction prediction by building up highly credible negative samples," *Bioinformatics*, vol. 31, pp. i221–i229, 2015.
- [288] S. Gunther, M. Kuhn, M. Dunkel, M. Campillos, C. Senger, E. Petsalaki, J. Ahmed, E. G. Urdiales, A. Gewiess, L. J. Jensen, R. Schneider, R. Skoblo, R. B. Russell, P. E. Bourne, P. Bork, and R. Preissner, "SuperTarget and matador: Resources for exploring drug-target relationships," *Nucleic Acids Res.*, vol. 36, pp. D919–D922, 2007.
- [289] D. S. Wishart, C. Knox, A. C. Guo, D. Cheng, S. Shrivastava, D. Tzur, B. Gautam, and M. Hassanali, "DrugBank: A knowledgebase for drugs, drug actions and drug targets," *Nucleic Acids Res.*, vol. 36, pp. D901–D906, 2007.
- [290] Y. Tabei and Y. Yamanishi, "Scalable prediction of compound-protein interactions using minwise hashing," *BMC Systems Biology*, vol. 7, p. S3, 2013.
- [291] M. K. Gilson, T. Liu, M. Baitaluk, G. Nicola, L. Hwang, and J. Chong, "BindingDB in 2015: A public database for medicinal chemistry, computational chemistry and systems pharmacology," *Nucleic Acids Research*, vol. 44, pp. D1045–D1053, 2015.

- [292] L. Wang, Z.-H. You, X. Chen, S.-X. Xia, F. Liu, X. Yan, Y. Zhou, and K.-J. Song, “A computational-based method for predicting drug–target interactions by using stacked autoencoder deep neural network,” *J. Comput. Biol.*, vol. 25, pp. 361–373, 2018.
- [293] O. Trott and A. J. Olson, “AutoDock vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading,” *J. Comput. Chem.*, vol. 31, pp. 455–461, 2009.
- [294] M. Ragoza, J. Hochuli, E. Idrobo, J. Sunseri, and D. R. Koes, “Protein–ligand scoring with convolutional neural networks,” *J. Chem. Inf. Model.*, vol. 57, pp. 942–957, 2017.
- [295] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-cam: Visual explanations from deep networks via gradient-based localization,” 2016, arXiv:1610.02391.
- [296] Y. Wang, N. Y. Chirgadze, S. L. Briggs, S. Khan, E. V. Jensen, and T. P. Burris, “A second binding site for hydroxytamoxifen within the coactivator-binding groove of estrogen receptor beta,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 103, pp. 9908–9911, 2006.
- [297] J. M. Hsieh, K. Tsiurlov, M. R. Sawaya, N. Magilnick, N. Abuladze, I. Kurtz, J. Abramson, and A. Pushkin, “Structures of aminoacylase 3 in complex with acetylated substrates,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 107, pp. 17 962–17 967, 2010.
- [298] T. Warne, M. J. Serrano-Vega, J. G. Baker, R. Moukhametzianov, P. C. Edwards, R. Henderson, A. G. W. Leslie, C. G. Tate, and G. F. X. Schertler, “Structure of a  $\beta$ 1-adrenergic g-protein-coupled receptor,” *Nature*, vol. 454, pp. 486–491, 2008.
- [299] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, “Deep neural networks for small footprint text-dependent speaker verification,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.
- [300] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, “X-vectors: Robust dnn embeddings for speaker recognition,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.
- [301] S. Ramoji, A. Mohan, B. Mysore, A. Bhatia, P. Singh, H. Vardhan, and S. Ganapathy, “The leap speaker recognition system for nist sre 2018 challenge,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5771–5775.
- [302] S. N. Hewitt, D. M. Dranow, B. G. Horst, J. A. Abendroth, B. Forte, I. Hallyburton, C. Jansen, B. Baragaña, R. Choi, K. L. Rivas, M. A. Hulverson, M. Dumais, T. E. Edwards, D. D. Lorimer, A. H. Fairlamb, D. W. Gray, K. D. Read, A. M. Lehane, K. Kirk, P. J. Myler, A. Wernimont, C. Walpole, R. Stacy, L. K. Barrett, I. H.

- Gilbert, and W. C. V. Voorhis, “Biochemical and structural characterization of selective allosteric inhibitors of the plasmodium falciparum drug target, prolyl-tRNA-synthetase,” *ACS Infect. Dis.*, vol. 3, pp. 34–44, 2016.
- [303] M. Duran-Frigola, A. Fernández-Torras, M. Bertoni, and P. Aloy, “Formatting biological big data for modern machine learning in drug discovery,” *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, vol. 9, p. e1408, 2018.
- [304] Y. Hong, B. Hou, H. Jiang, and J. Zhang, “Machine learning and artificial neural network accelerated computational discoveries in materials science,” *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, p. e1450, 2019.
- [305] H. J. Kulik, “Making machine learning a useful tool in the accelerated discovery of transition metal complexes,” *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, p. e1439, 2019.
- [306] P. Cimermancic, P. Weinkam, T. J. Rettenmaier, L. Bichmann, D. A. Keedy, R. A. Woldeyes, D. Schneidman-Duhovny, O. N. Demerdash, J. C. Mitchell, J. A. Wells, J. S. Fraser, and A. Sali, “CryptoSite: Expanding the druggable proteome by characterization and prediction of cryptic binding sites,” *J. Mol. Bio.*, vol. 428, pp. 709–719, 2016.
- [307] S. Vajda, D. Beglov, A. E. Wakefield, M. Egbert, and A. Whitty, “Cryptic binding sites on proteins: definition, detection, and druggability,” *Curr. Opin. Chem. Biol.*, vol. 44, pp. 1–8, 2018.
- [308] R. A. Laskowski, F. Gerick, and J. M. Thornton, “The structural basis of allosteric regulation in proteins,” *FEBS Letters*, vol. 583, pp. 1692–1698, 2009.
- [309] D. Beglov, D. R. Hall, A. E. Wakefield, L. Luo, K. N. Allen, D. Kozakov, A. Whitty, and S. Vajda, “Exploring the structural origins of cryptic sites on proteins,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 115, pp. E3416–E3425, 2018.
- [310] L. Zou, F. Ruan, M. Huang, L. Liang, H. Huang, Z. Hong, J. Yu, M. Kang, Y. Song, J. Xia, Q. Guo, T. Song, J. He, H.-L. Yen, M. Peiris, and J. Wu, “SARS-CoV-2 viral load in upper respiratory specimens of infected patients,” *New Eng J Med.*, vol. 382, pp. 1177–1179, 2020.
- [311] B. R. Amman, S. A. Carroll, Z. D. Reed, T. K. Sealy, S. Balinandi, R. Swanepoel, A. Kemp, B. R. Erickson, J. A. Comer, S. Campbell, D. L. Cannon, M. L. Khristova, P. Atimnedi, C. D. Paddock, R. J. K. Crockett, T. D. Flietstra, K. L. Warfield, R. Unfer, E. Katongole-Mbidde, R. Downing, J. W. Tappero, S. R. Zaki, P. E. Rollin, T. G. Ksiazek, S. T. Nichol, and J. S. Towner, “Seasonal pulses of marburg virus circulation in juvenile rousettus aegyptiacus bats coincide with periods of increased risk of human infection,” *PLoS Pathogens*, vol. 8, p. e1002877, 2012.
- [312] G. R. Bowman, E. R. Bolin, K. M. Hart, B. C. Maguire, and S. Marqusee, “Discovery of multiple hidden allosteric sites by combining markov state models and experiments,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 112, pp. 2734–2739, 2015.

- [313] O. Schueler-Furman and S. J. Wodak, "Computational approaches to investigating allostery," *Curr. Opin. Chem. Biol.*, vol. 41, pp. 159–171, 2016.
- [314] S. G. Rohrer and K. Baumann, "Maximum unbiased validation (MUV) data sets for virtual screening based on PubChem bioactivity data," *J Chem Inf Model*, vol. 49, no. 2, pp. 169–184, 2009.
- [315] J. Yang, C. Shen, and N. Huang, "Predicting or pretending: Artificial intelligence for protein-ligand interactions lack of sufficiently large and unbiased datasets," *Front. Pharmacol.*, vol. 11, 2020.
- [316] J. Sieg, F. Flachsenberg, and M. Rarey, "In need of bias control: Evaluating chemical data for machine learning in structure-based virtual screening," *J. Chem. Inf. Model.*, vol. 59, no. 3, pp. 947–961, 2019.
- [317] M. von Korff, J. Freyss, and T. Sander, "Comparison of ligand- and structure-based virtual screening on the DUD data set," *J. Chem. Inf. Model.*, vol. 49, pp. 209–231, 2009.
- [318] V. Venkatraman, V. I. Pérez-Nueno, L. Mavridis, and D. W. Ritchie, "Comprehensive comparison of ligand-based virtual screening tools against the DUD data set reveals limitations of current 3d methods," *J. Chem. Inf. Model.*, vol. 50, pp. 2079–2093, 2010.
- [319] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: moving beyond fingerprints," *J. Comput. Aided Mol. Des.*, vol. 30, pp. 595–608, 2016.
- [320] L. Masters, S. Eagon, and M. Heying, "Evaluation of consensus scoring methods for AutoDock vina, smina and idock," *J. Mol. Graph. Model.*, vol. 96, p. 107532, 2020.
- [321] S. S. Ericksen, H. Wu, H. Zhang, L. A. Michael, M. A. Newton, F. M. Hoffmann, and S. A. Wildman, "Machine learning consensus scoring improves performance across targets in structure-based virtual screening," *J. Chem. Inf. Model.*, vol. 57, pp. 1579–1590, 2017.
- [322] J. B. Cross, D. C. Thompson, B. K. Rai, J. C. Baber, K. Y. Fan, Y. Hu, and C. Humblet, "Comparison of several molecular docking programs: Pose prediction and virtual screening accuracy," *J. Chem. Inf. Model.*, vol. 49, pp. 1455–1474, 2009.
- [323] P. Kolb and J. Irwin, "Docking screens: Right for the right reasons?" *Curr Top Med Chem.*, vol. 9, pp. 755–770, 2009.
- [324] T. Scior, A. Bender, G. Tresadern, J. L. Medina-Franco, K. Martínez-Mayorga, T. Langer, K. Cuanalo-Contreras, and D. K. Agrafiotis, "Recognizing pitfalls in virtual screening: A critical review," *J. Chem. Inf. Model.*, vol. 52, pp. 867–881, 2012.

- [325] A. N. Jain and A. Nicholls, “Recommendations for evaluation of computational methods,” *J. Comput. Aided Mol. Des.*, vol. 22, pp. 133–139, 2008.
- [326] A. Nicholls, “What do we know and when do we know it?” *J. Comput. Aided Mol. Des.*, vol. 22, pp. 239–255, 2008.
- [327] M. McGann, A. Nicholls, and I. Enyedy, “The statistics of virtual screening and lead optimization,” *J. Comput. Aided Mol. Des.*, vol. 29, pp. 923–936, 2015.
- [328] N. Furnham, G. L. Holliday, T. A. P. de Beer, J. O. B. Jacobsen, W. R. Pearson, and J. M. Thornton, “The catalytic site atlas 2.0: cataloging catalytic sites and residues identified in enzymes,” *Nucleic Acids Res.*, vol. 42, pp. D485–D489, 2013.
- [329] M. McGann, “FRED and HYBRID docking performance on standardized datasets,” *J. Comput. Aided Mol. Des.*, vol. 26, no. 8, pp. 897–906, 2012.
- [330] J.-F. Truchon and C. I. Bayly, “Evaluating virtual screening methods: good and bad metrics for the “early recognition” problem,” *J. Chem. Inf. Model.*, vol. 47, pp. 488–508, 2007.
- [331] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training gans,” *arXiv preprint arXiv:1606.03498*, 2016.
- [332] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” *arXiv preprint arXiv:1706.08500*, 2017.
- [333] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.
- [334] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.
- [335] W. Brendel and M. Bethge, “Approximating cnns with bag-of-local-features models works surprisingly well on imagenet,” *arXiv preprint arXiv:1904.00760*, 2019.
- [336] “Inception Score,” <https://medium.com/octavian-ai/a-simple-explanation-of-the-inception-score-372dff6a8c7a>, [].
- [337] J. Y. Koh, J. Baldridge, H. Lee, and Y. Yang, “Text-to-image generation grounded by fine-grained user attention,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 237–246.
- [338] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *European conference on computer vision*. Springer, 2014, pp. 740–755.

- [339] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, “The caltech-ucsd birds-200-2011 dataset,” 2011.
- [340] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.
- [341] E. Collins, R. Bala, B. Price, and S. Susstrunk, “Editing in style: Uncovering the local semantics of gans,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5771–5780.
- [342] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, “Ganspace: Discovering interpretable gan controls,” *arXiv preprint arXiv:2004.02546*, 2020.
- [343] Y. Shen, J. Gu, X. Tang, and B. Zhou, “Interpreting the latent space of gans for semantic face editing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9243–9252.
- [344] A. Tewari, M. Elgharib, G. Bharaj, F. Bernard, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt, “Stylerig: Rigging stylegan for 3d control over portrait images,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6142–6151.