T.C.
REPUBLIC OF TURKEY
HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF HEALTH SCIENCES

AUTOMATED DESIGN OF DRUG CANDIDATE MOLECULES WITH DEEP GRAPH LEARNING

Atabey ÜNLÜ

Program of Bioinformatics

MASTER'S THESIS

Ankara

2023

T.C.
REPUBLIC OF TURKEY
HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF HEALTH SCIENCES

AUTOMATED DESIGN OF DRUG CANDIDATE MOLECULES WITH DEEP GRAPH LEARNING

Atabey ÜNLÜ

Program of Bioinformatics

MASTER'S THESIS

ADVISOR OF THE THESIS

Assoc. Prof. Tunca DOĞAN

Ankara

2023

APPROVAL PAGE

HACETTEPE UNIVERSITY

GRADUATE SCHOOL OF HEALTH SCIENCES

AUTOMATED DESIGN OF DRUG CANDIDATE MOLECULES WITH DEEP GRAPH LEARNING

Atabey Ünlü

Supervisor: Assoc. Prof. Tunca Doğan

Co-supervisor: -

This thesis study has been approved and accepted as a Master dissertation in "Bioinformatics Program" by the assessment committee, whose members are

listed below, on 10/08/2023

Chairman of the Committee : Assist. Prof. Tunca Doğan

Hacettepe University

Advisor of the Dissertation : Assoc. Prof. Tunca Doğan

Hacettepe University

Member : Assoc. Prof. Aybar Acar
Middle East Technical University

This dissertation has been approved by the above committee in conformity to the related issues of Hacettepe University Graduate Education and Examination Regulation.

Prof. Müge YEMİŞCİ ÖZKAN, MD, PhD

Director

YAYIMLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kâğıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe Üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan ***"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"*** kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H.Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir. (1)
- Enstitü / Fakülte yönetim kurulunun gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren ... ay ertelenmiştir. (2)
- Tezimle ilgili gizlilik kararı verilmiştir. (3)

… / … /…
Atabey Ünlü

1"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"

*(1). Madde 6. 1. Lisansüstü tezle ilgili patent başvurusu yapılması veya patent alma sürecinin devam etmesi durumunda, tez **danışmanının** önerisi ve **enstitü anabilim dalının** uygun görüşü üzerine **enstitü** veya **fakülte yönetim kurulu** iki yıl süre ile tezin erişime açılmasının ertelenmesine karar verebilir.*

*(2). Madde 6. 2. Yeni teknik, materyal ve metotların kullanıldığı, henüz makaleye dönüşmemiş veya patent gibi yöntemlerle korunmamış ve internetten paylaşılması durumunda 3. şahıslara veya kurumlara haksız kazanç imkânı oluşturabilecek bilgi ve bulguları içeren tezler hakkında tez **danışmanının** önerisi ve **enstitü anabilim dalının** uygun görüşü üzerine **enstitü** veya **fakülte yönetim kurulunun** gerekçeli kararı ile altı ayı aşmamak üzere tezin erişime açılması engellenebilir.*

*(3). Madde 7. 1. Ulusal çıkarları veya güvenliği ilgilendiren, emniyet, istihbarat, savunma ve güvenlik, sağlık vb. konulara ilişkin lisansüstü tezlerle ilgili gizlilik kararı, **tezin yapıldığı kurum** tarafından verilir *. Kurum ve kuruluşlarla yapılan iş birliği protokolü çerçevesinde hazırlanan lisansüstü tezlere ilişkin gizlilik kararı ise, **ilgili kurum ve kuruluşun önerisi** ile **enstitü** veya **fakültenin** uygun görüşü üzerine **üniversite yönetim kurulu** tarafından verilir. Gizlilik kararı verilen tezler Yükseköğretim Kuruluna bildirilir.*

Madde 7.2. Gizlilik kararı verilen tezler gizlilik süresince enstitü veya fakülte tarafından gizlilik kuralları çerçevesinde muhafaza edilir, gizlilik kararının kaldırılması halinde Tez Otomasyon Sistemine yüklenir

****** Tez **danışmanının** önerisi ve **enstitü anabilim dalının** uygun görüşü üzerine **enstitü** veya **fakülte yönetim kurulu tarafından karar verilir.***

ETHICAL DECLARATION

In this thesis study, I declare that all the information and documents have been obtained in the base of the academic rules and all audio-visual and written information and results have been presented according to the rules of scientific ethics. I did not do any distortion in the data set. In the case of using other works, related studies have been fully cited in accordance with the scientific standards. I also declare that my thesis study is original except cited references. It was produced by myself in consultation with supervisor Assoc. Prof. Tunca DOĞAN and written according to the rules of thesis writing of Hacettepe University Institute of Health Sciences.

Atabey Ünlü

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to the following individuals and organizations who have supported and contributed to the completion of this thesis:

I am immensely grateful to my supervisor, Assoc. Prof. Tunca Doğan, for his guidance, expertise, and unwavering support throughout the entire research process. His invaluable insights and constructive feedback have played a pivotal role in shaping the direction of this study.

I am grateful that my colleagues Elif Çevrim, and Heval Ataş Güvenilir for their part in helping with the dataset preparation, Ahmet Sarıgün and Melih Gökay Yiğit for their help with the design process and downstream analysis, Erva Ulusoy and Hayriye Çelikbilek for their support while conducting and writing this thesis and the resources they provided, which helped my research endeavors.

Finally, I would like to express my deepest gratitude to my wife, Gülafra, and my family. Their unwavering love, support, and belief in my abilities have been my constant motivation. I am grateful for their sacrifices and the encouragement they have provided throughout my academic pursuit.

While every effort has been made to acknowledge all individuals who have contributed to this thesis, I apologize if any oversight has occurred. Please accept my sincere appreciation for your contributions, no matter how large or small.

That’s all folks!

This thesis is supported by TUBITAK 2210/A National MSc Scholarship Program.

ABSTRACT

Ünlü, A. Automated Design of Drug Candidate Molecules with Deep Graph Learning, Hacettepe University Graduate School of Health Sciences Bioinformatics Program Master's Thesis, Ankara, 2023. The discovery of new drug candidate molecules is an important step in the process of drug development. Deep generative learning, a frequently used approach in the field of artificial intelligence in recent years, has emerged as a promising method for generating realistic synthetic data within a defined theme. Additionally, the utility of these models in the drug development process depends on their ability to generate molecules specific to the biological target. In this study, a new generative system called "DrugGEN" has been developed specifically for the de novo design of drug candidate molecules that will interact with selected target proteins. The system represents compounds and protein structures as graphs and processes them using two sequentially connected generative adversarial networks (GANs) incorporating graph transformers. The training dataset of the system was created from a large collection of drug-like compound records and target-specific bioactive molecules obtained from the ChEMBL database. The developed model was trained with the aim of designing new molecules targeting the AKT1 protein, which plays a critical role in various cancer types. The performance of the DrugGEN model was evaluated comparatively with other methods in the literature using fundamental criteria. In addition, explanatory data analysis was performed on the generated results. The results demonstrated the novelty of molecules designed de novo by DrugGEN. Furthermore, it was shown that the outputs were comparable to the known ligands of the AKT1 protein both in terms of physicochemical properties and structure. Consequently, in this study, an artificial intelligence model was developed using deep learning algorithms and extensive chemical and biological data to automatically design completely novel molecules with the ability to target selected proteins.

Keywords: Machine Learning, Drug Discovery, AKT Protein

ÖZET

Ünlü, A. Derin Çizge Öğrenmesi ile İlaç Adayı Moleküllerin Otomatik Şekilde Tasarımı, Hacettepe Üniversitesi Sağlık Bilimleri Enstitüsü Biyoinformatik Programı Yüksek Lisans Tezi, Ankara, 2023. Yeni ilaç adayı moleküllerin keşfi, ilaç geliştirme sürecinde önemli bir adımdır. Yapay zekâ alanında son yıllarda sıkça kullanılmaya başlanan üretici derin öğrenme, belirlenen bir tema içinde gerçekçi sentetik veri üretme konusunda umut vaat eden bir yaklaşım olarak ön plana çıkmaktadır. Bunun yanında, bu modellerin ilaç geliştirme süreçlerinde kullanılabilirlikleri, biyolojik hedefe özgü moleküller üretme yeteneklerine bağlıdır. Bu çalışmada, seçilen hedef proteinlerle etkileşime girecek ilaç adayı moleküllerin de novo tasarımı için özel olarak oluşturulmuş yeni bir üretici sistem olan "DrugGEN" geliştirilmiştir. Sistem, bileşikleri ve protein yapılarını çizgeler olarak temsil eder ve bunları çizge dönüştürücü ("Transformer") içeren iki adet seri şekilde bağlı üretken rekabetçi ağ ("Generative Adversarial Network", GAN) kullanarak işlemektedir. Sistemin eğitim veri seti, ChEMBL veri tabanından elde edilen ilaç benzeri bileşik kayıtları ve hedefe özgü biyoaktif molekülleri içeren büyük bir veri kümesinden oluşturulmuştur. Geliştirilen model, farklı kanser tiplerinde kritik öneme sahip olan AKT1 proteinini hedefleyecek yeni moleküller tasarlaması amacıyla eğitime tabi tutulmuştur. DrugGEN modelinin performansı temel ölçütler kullanılarak, literatürdeki diğer yöntemlerle karşılaştırmalı biçimde değerlendirilirmiştir. Bunun yanında, üretilen sonuçlar üzerinde açıklayıcı veri analizi gerçekleştirilmiştir. Sonuçlar, DrugGEN tarafından de novo olarak tasarlanan moleküllerin orijinalliğini kanıtlamıştır. Ayrıca, çıktıların fizikokimyasal ve yapısal olarak AKT1 proteinin bilinen ligandlarıyla karşılaştırılabilir olduğu gösterilmiştir. Sonuç olarak, bu çalışmada derin öğrenme algoritmaları ve geniş çaplı kimyasal ve biyolojik veri kullanılarak seçili proteinleri hedefleme yeteneğine sahip tamamen yeni moleküllerin tasarımını otomatik biçimde gerçekleştiren bir yapay zekâ modeli geliştirilmiştir.

Anahtar Kelimeler: Makine Öğrenmesi, İlaç Keşfi, AKT Proteini

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATONS

GAN	Generative adversarial networks
MOSES	Molecular sets
SMILES	Simplified molecular-input line-entry system
AKT1	AKT serine/threonine kinase 1
DL	Deep learning
ML	Machine learning
CAMM	Computer assisted molecular modelling
CNN	Convolutional neural network
RNN	Recurrent neural network
LSTM	Long short term memory
VAE	Variational autoencoder
SAR	Structure activity relation
QSAR	Quantitative structure activity relation
GNN	Graph neural network
GNF	Graph normalizing flow
PDB	Protein data bank
RL	Reinforcement learning
MLP	Multilayer perceptron
GAN1	Generative adversarial network 1
GAN2	Generative adversarial network 2
G1	Generator 1
D1	Discriminator 1
G2	Generator 2
D2	Discriminator 2

C	Carbon
N	Nitrogen
O	Oxygen
F	Fluorine
Cl	Chlorine
Br	Bromine
S	Sulfur
P	Phosphorus
Å	Angstroms
QED	Quantitative estimate of drug-likeness
SA	Synthetic accessibility
LogP	Partition coefficient
3D	3 dimension
SELFIES	Self-referencing embedded strings
CVAE	Conditional variational autoencoder
MW	Molecular weight
ELBO	Evidence lower bound
KL	Kullback-Leibler
TUBITAK	The scientific and technological research council of Türkiye

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Drug development is a complex and time-consuming process that poses challenges to the rapid discovery of new drugs for complex diseases. The various steps involved, from initial screening to phase studies, demand significant resources and expertise. Traditional screening methods are labor-intensive and require extensive human effort to evaluate a diverse range of molecules against drug targets. Similarly, the process of designing de novo molecules using conventional methods can be time-consuming and necessitates specialized knowledge of the target. To overcome these limitations, emerging de novo design methods leverage advanced algorithms and models. These approaches harness the power of machine learning and deep learning algorithms to identify patterns in molecular data. By learning from available data, these models can generate novel and effective molecules, bypassing the need for prolonged timelines, extensive human involvement, and substantial funding. However, it is important to note that de novo molecule design is not a simple and immediate solution to drug design. Many generated de novo molecules may not be suitable for human use, and further refinement and optimization are often necessary. Target-based de novo drug design presents a promising approach to enhance the effectiveness of de novo design by integrating target-specific information with molecular features. By incorporating knowledge about the target, such as its structure and function, along with the physiochemical properties of molecules, it becomes possible to design structurally and physiochemically robust de novo molecules. This approach holds potential for improving the efficiency and success rates of de novo drug design efforts.

1.1. Problem Definition

Deep learning based de novo molecule design often leads to the generation of molecules that are not well-suited for becoming drug candidates. Models in this context primarily learn the statistical distribution and patterns of molecular features without considering the specific characteristics of the target. Consequently, such models tend to

replicate the physicochemical and structural features of existing molecules without necessarily optimizing them for the desired target. It becomes essential to consider the structural and functional properties of the target protein to guide the generation of molecules that possess desirable characteristics. The aspect of target-based molecule generative modeling discussed in this thesis is not extensively researched in the literature. The completion of this thesis will contribute to the development of a target-specific molecule generation model that has not yet been thoroughly studied in the molecule generative models.

1.2. Hypothesis

Deep learning-based de novo drug design has the capability to create molecules that are absent from existing databases. Within the literature, numerous studies have been conducted, primarily focusing on the generation of random molecules or the production of molecules possessing optimized traits. However, the efficacy of these designed molecules hinges on their ability to interact effectively with the designated target. Mere generation of random molecules or design according to specific characteristics proves inadequate for creating interacting partners tailored to precise targets. To address this, the integration of target information into the design system becomes pivotal. However, the exploration of target-based de novo molecule design remains limited within current research. Our hypothesis is that by incorporating target features into molecule generation, the employment of deep learning algorithms can yield superior design of drug-like molecules. This approach facilitates a more profound comprehension of the interaction requirements of the selected target, enabling the design of molecules based on this informed understanding.

1.3. Aim & Objective of the Study

The aim of this thesis is to implement an automated target-aware drug design model. The proposed model will integrate molecule features, validated drugs, and target characteristics into a novel deep learning-based framework. The goal is to develop a

model capable of designing potential drug candidate molecules specifically tailored to a given drug target.

Main objectives to implement this model are:

1. To obtain molecules and drugs in the form of SMILES text from ChEMBL and DrugBank databases.
2. To get the binding pocket structure of the AKT1 protein from PDB database.
3. To implement a custom SMILES-to-graph-structure function.
4. To design and implement a molecule generative deep learning model is developed using generative adversarial networks (GANs).
5. To train, optimize, and validate the designed model, utilizing the molecular sets (MOSES) benchmark, which is an established benchmark for generative models.
6. To test validated models through downstream analysis to further assess their generative success of trained the model.

These objectives are the main steps of this thesis to implement a target-based de novo molecules generative model.

2. GENERAL INFORMATION

The size of the chemical space encompassing potential drug candidate molecules ranges from 10^{23} to 10^{60}, rendering it practically impossible to thoroughly explore its boundaries. To initiate a search for lead molecules, one practical approach on the experimental front involves utilizing technologies like high-throughput screening (HTS). Nevertheless, it is important to note that such screenings are constrained to known chemical libraries, thus restricting the ability to search for entirely new molecules. Consequently, scientists often find themselves compelled to focus on identifying molecules that exhibit comparable physicochemical and pharmacological properties (1). Advancements in screening technologies have undeniably improved the rates of synthetic accessibility and the overall speed of identifying potential drug candidates. Nevertheless, it is worth noting that these advancements have not entirely resolved the efficiency issues and high failure rates associated with the drug development process (2).

The availability of biomolecular data significantly grew, thereby facilitating the development and application of advanced statistical algorithms in the field of drug discovery, particularly in conjunction with experimental techniques such as HTS (3). Furthermore, the increase in computational power and its associated reduced costs have contributed to the processing of complex and voluminous biomolecular data. Machine learning and artificial intelligence algorithms leverage these advancements to create robust models that aid in the design of molecules (4). Machine learning models have found utility in a wide array of tasks spanning from molecular docking to molecular modeling. In initial studies, machine learning methods such as random forest and support vector machines were employed to classify molecules or make predictions about their features. However, for more intricate tasks like docking predictions, artificial neural networks were employed. These neural networks were trained using docking poses to enhance their predictive capabilities in docking scenarios (5).

Deep neural networks excel at complex transformations and abstract feature learning, surpassing shallow networks. They enhance molecular representation and compound classification without the need for complex descriptors. Additionally, deep architectures enable feature reuse and knowledge transfer between tasks, improving handling of missing data and multitask learning. These advancements enable efficient bioactivity testing against multiple targets and have potential applications in drug repurposing and identifying off-target activities (6). Deep neural networks are extensively used in de novo molecular design. Optimization algorithms guide molecule generation based on a given representation and objective function. Deep learning approaches, pretrained on large molecular datasets, efficiently explore property surfaces for optimal solutions. Various deep learning architectures, such as variational autoencoders, generative adversarial networks, recurrent neural networks, and transformers, have been proposed for generating molecule structures. Trained generative models enable sampling from the learned chemical space, coupled with Bayesian optimization or reinforcement learning, to efficiently identify desirable molecular profiles (7).

The application of artificial intelligence in generative modelling has led to the emergence of next generation de novo drug design methods. These methods, inspired by successful architectures in image and text generation, rapidly generate new lead compounds with desired biological and chemical properties. While generative modeling techniques show promise, further improvements, computational and experimental validations, and benchmarking tests are necessary. Nonetheless, generative models are expected to become a crucial component in de novo drug design, aiding medicinal chemists in generating novel ideas and expediting the drug discovery process (8).

2.1. Drug Development

Drug development is a complex and time-consuming process that involves the discovery, optimization, and evaluation of potential therapeutic compounds. Traditionally, this process heavily relied on empirical methods and high-throughput screening assays (9). Although high-throughput screening technology has enabled the simultaneous screening of thousands of compounds, the vast size of chemical and biomolecular spaces often hinders the discovery of optimal candidate molecules (128). With this rapid advancement of computational techniques, computational drug development has gained significant attention and recognition. As a result, many identified drug candidates ultimately fail in later stages due to high toxicity or low efficacy, leading to low success rates in drug development. By employing computational models and algorithms, researchers can bypass such obstructions by predicting and assessing the properties and behavior of drug candidates. This approach not only accelerates the drug discovery process but also aids in the design of safer and more effective treatments (10).

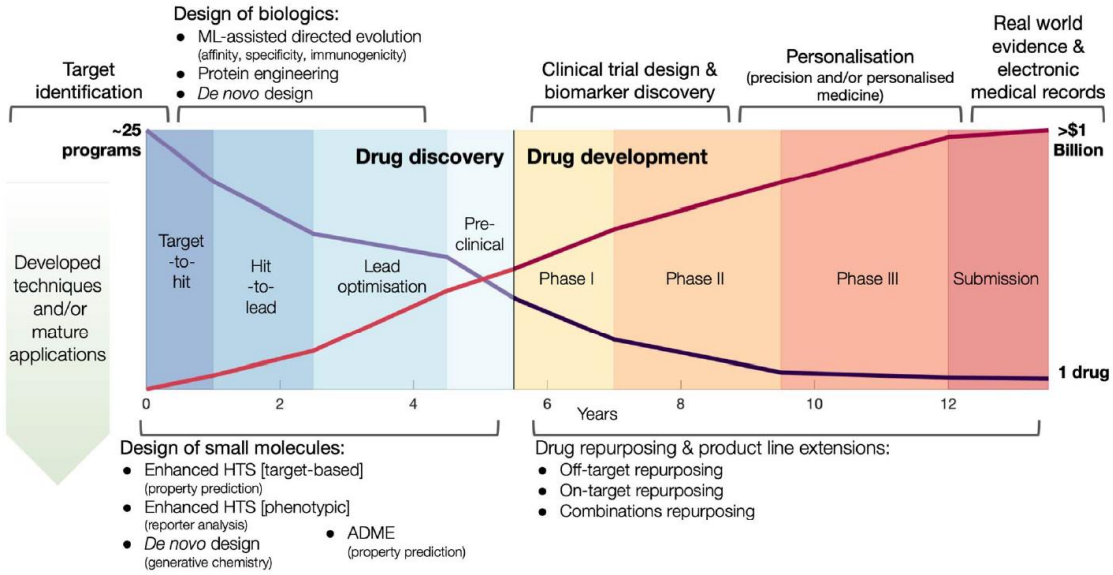

Figure 2.1. Computational and experimental phases of drug discovery. Adapted from (92).

This innovative discipline unites principles from computer science, bioinformatics, and medicinal chemistry to speed up discovering and designing novel therapeutic agents. By harnessing computational models, algorithms, and large-scale data analysis, researchers can efficiently screen extensive chemical libraries and predict the biological activity, pharmacokinetics, and toxicity of potential drug candidates (11). The integration of computational methods in drug development confers several advantages over traditional experimental approaches. It facilitates the rapid identification of lead compounds, streamlines the optimization of their properties, and diminishes the cost and time associated with experimental screening. Furthermore, computational drug design enables the exploration of diverse chemical space, presenting opportunities for the development of drugs with enhanced efficacy and reduced side effects (12).

2.2. Molecule Design

Molecule design, also referred to as molecular design, assumes a pivotal role in the exploration and development of novel drugs, materials, and chemical compounds (13). It encompasses the intentional and systematic creation of molecular structures possessing specific desired properties. By drawing upon principles from chemistry, physics, and computational science, researchers strive to design molecules that exhibit enhanced characteristics such as heightened efficacy, improved stability, and optimized biological activity (14). The process of molecule design encompasses diverse approaches, including rational design, de novo design, and fragment-based design, all of which heavily rely on computational methods and advanced modeling techniques (15).

Computational methods have become indispensable in molecule design, empowering scientists to traverse vast chemical spaces and predict the properties and behavior of molecules with remarkable precision. Quantum mechanical calculations, molecular dynamics simulations, and machine learning algorithms find extensive application in this domain. Quantum mechanics provides insights into the electronic structure, stability, and reactivity of molecules, guiding the design process based on

fundamental principles. Molecular dynamics simulations, on the other hand, enable researchers to investigate the dynamic behavior and interactions of molecules, aiding in the optimization of molecular structures and comprehension of their stability and conformational flexibility (16,17).

Machine learning algorithms, such as deep neural networks, have emerged as invaluable tools for expediting the molecule design process. These algorithms can be trained by extensive databases of existing chemical compounds, facilitating the generation of novel molecules with desired properties. Furthermore, they contribute to property prediction, encompassing solubility, toxicity, and bioactivity, thereby streamlining the molecule design pipeline. Through the integration of computational methods and advanced modeling techniques, molecule design holds tremendous promise in advancing various fields, including drug discovery, material science, and sustainable chemistry (17,18).

2.2.1. Traditional Approaches

Traditional approaches for molecule design encompass a combination of empirical and theoretical methods that have been widely employed in the field of chemistry for decades. These methods rely on established chemical principles and experimental observations to guide the design process (19). One such approach is structure-activity relationship (SAR) analysis, which entails investigating the correlation between a molecule's structure and its biological activity or desired property. By systematically modifying specific functional groups or regions of a molecule and assessing their impact on activity, researchers can gain valuable insights into the structure-activity relationship, thereby facilitating the design of molecules with enhanced properties (20). Another traditional method is scaffold hopping, wherein chemists identify a known molecular scaffold with desired properties and systematically modify it to generate novel compounds with analogous characteristics (21). Furthermore, medicinal chemists often employ retrosynthetic analysis, wherein the desired molecule is deconstructed into simpler building blocks, enabling the planning of a synthetic route

for the assembly of the final compound (22). Despite being time-consuming and resource-intensive, these traditional approaches have yielded significant successes in the development of numerous drugs and materials over the years, and they continue to serve as invaluable tools in molecule design and optimization (23).

2.2.2. Modern Approaches

Modern approaches to molecule design have experienced remarkable advancements, propelled by the integration of computational methods, high-throughput screening, and data-driven techniques. One prominent modern approach is computer-assisted molecular modeling (CAMM), which leverages computational modeling, virtual screening, and molecular docking to identify potential drug candidates. By utilizing three-dimensional structural information of target proteins and small molecule libraries, CAMM enables the prediction of binding affinities and the identification of novel lead compounds (24). Additionally, fragment-based drug design (FBDD) has gained prominence, where small fragments are screened and combined to construct larger molecules with optimized interactions. This approach facilitates a focused exploration of chemical space and efficient optimization of lead compounds (25). Furthermore, machine learning (ML) and deep learning (DL) techniques have found increasing application in molecule design. These methods can generate and evaluate extensive libraries of molecules, predict their properties, and guide the discovery of novel chemical space. The integration of experimental data with ML and DL models enables the development of more accurate predictive models for molecule design (26). Due to its success in processing large, complex datasets such as biological/biomedical data, which often contain errors and missing information, deep learning has recently started to be integrated into the fields of bioinformatics and chemoinformatics (76,77). In the field of drug discovery, computational approaches known as virtual screening are employed to tackle problems associated with traditional methods. These studies primarily aim to predict molecules that could be potential drug candidates or to reposition existing drugs

for different therapeutic purposes using data obtained from bioactivity measurement experiments (78).

2.3. Machine Learning & Deep Learning

Machine learning, a subset of artificial intelligence, encompasses the development of algorithms and models that can learn patterns and make predictions or decisions without explicit programming. It involves training models on extensive datasets to uncover underlying patterns and relationships, enabling them to generalize and provide accurate predictions on new, unseen data. Machine learning field contains various techniques, including statistical methods, regression models, decision trees, support vector machines, and more (27,28).

Deep learning, on the other hand, is a specialized branch of machine learning that harnesses artificial neural networks with multiple layers to acquire hierarchical representations of data. These deep neural networks excel at learning intricate patterns and have revolutionized diverse domains, such as computer vision, natural language processing, and speech recognition. Deep learning architectures have a diverse range of models that leverage artificial neural networks with multiple layers to acquire intricate representations of data. These architectures have propelled significant advancements in the field of machine learning, enabling breakthroughs in various domains (29,30).

Convolutional Neural Networks (CNNs) excel in image analysis tasks by exploiting the spatial relationships inherent in images. They employ convolutional layers to autonomously learn hierarchical representations of features, enabling tasks such as object recognition and image classification (31). The pioneering studies demonstrated the effectiveness of CNNs in achieving state-of-the-art results on the ImageNet dataset (32). Recurrent Neural Networks (RNNs) are tailored to handle sequential data, making them well-suited for natural language processing, speech recognition, and time series analysis (33). RNNs maintain internal states, enabling them to capture dependencies over long sequences. The Long Short-Term Memory (LSTM) network is a widely adopted

variant of RNNs that effectively addresses the vanishing gradient problem and facilitates modeling of long-range dependencies (34).

In the biological sciences, machine learning and deep learning models have found extensive applications in genomics, proteomics, and drug discovery. These models can analyze vast genomic and proteomic datasets to identify patterns, predict protein structure and function, and guide drug discovery endeavors (35). In the chemistry field, machine learning and deep learning demonstrate promise in various domains. They have been applied to quantitative structure-activity relationship (QSAR) modeling, enabling the prediction of chemical properties and biological activities of molecules. Furthermore, deep learning models can analyze chemical reaction data, predict reaction outcomes, and assist in retrosynthesis planning (36).

2.4. Generative Modelling

Generative modeling with deep learning has emerged as a powerful approach for creating new samples that resemble the training data. These models learn the underlying distribution of the data and can generate novel samples that exhibit similar characteristics (37). They have found applications in various domains, including computer vision, natural language processing, and biological and chemical sciences (38,39).

One prominent generative model is the Generative Adversarial Network (GAN), that consists of a generator network and a discriminator network that compete against each other. The generator learns to produce realistic samples, while the discriminator learns to distinguish between real and generated samples (40). GANs have demonstrated remarkable success in generating realistic samples, such as images, music, and text. They have also been applied to tasks like data augmentation and domain adaptation. GANs showcased their potential for generating high-quality synthetic images (41).

Variational Autoencoders (VAEs) represent a widely used generative model architecture. VAEs acquire knowledge of a low-dimensional latent space portrayal of the input data, enabling the generation of novel samples through sampling from this space. These models integrate an encoder network responsible for mapping the data into the

latent space, and a decoder network responsible for reconstructing the data based on the latent representation (42). VAEs, alongside subsequent research, have found applications across various domains, such as image generation and molecular design (43, 44).

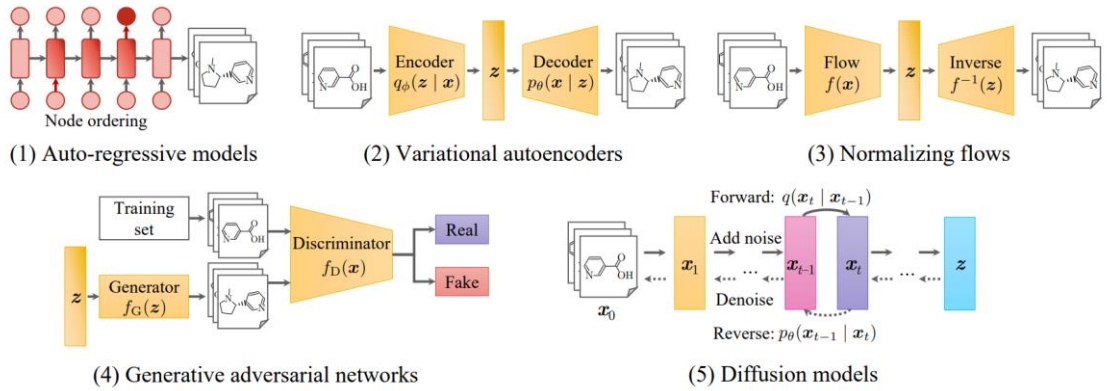

Figure 2.2. Different neural networks that are specifically designed to generate artificial data. Adapted from (93).

In the field of biology and chemistry, generative models have shown significant potential. For example, generative models have been employed in de novo drug design, where they generate new molecular structures with desired properties. Studies utilized generative models to design new molecules with desired properties, facilitating the exploration of chemical space and accelerating the drug discovery process (45, 46). Generative models have also been applied to protein structure prediction, which is crucial for understanding protein function and designing therapeutics. DeepMind's AlphaFold, based on deep learning and generative modeling principles, achieved remarkable success in predicting protein structures with high accuracy (47). These examples highlight the wide-ranging applications of generative modeling with deep learning in biological and chemical sciences. These models have the potential to revolutionize drug discovery, protein engineering, and molecular design by enabling the generation of novel molecules and predicting important biomolecular properties (48).

2.4.1. VAEs

Variational Autoencoders (VAEs) are powerful generative models that learn a latent space representation of input data and enable the generation of new samples by sampling from this latent space. VAEs consist of two main components: an encoder network and a decoder network. The encoder network maps the input data to a latent space, while the decoder network reconstructs the data from the latent representation (49). In VAEs, the latent space is typically modeled as a multivariate Gaussian distribution with a mean and variance. During training, VAEs aim to maximize the evidence lower bound (ELBO), which comprises a reconstruction term and a regularization term. The reconstruction term encourages the decoder to accurately reconstruct the input data, while the regularization term, often based on the Kullback-Leibler (KL) divergence, ensures that the learned latent space adheres to the desired distribution (50, 51).

The effectiveness of VAEs in generating new digits from the MNIST dataset showcased the potential of the learned latent space for data interpolation and manipulation VAEs have found applications in various domains, including image generation, natural language processing, and molecular design (52, 53). In image generation, VAEs have been employed to generate realistic images across diverse datasets (54). In the domain of natural language processing, VAEs are utilized for text generation and demonstrated their ability to reconstruct and generate coherent sentences (55).

Furthermore, VAEs have been employed in the field of chemistry for tasks such as de novo drug design and molecular optimization. VAEs can be employed to generate novel molecular structures with desired properties, showcasing their potential for accelerating the drug discovery process. Overall, VAEs provide a powerful framework for learning latent representations of data and generating new samples. They have been successfully applied in various domains, demonstrating their versatility and potential for creative applications (56).

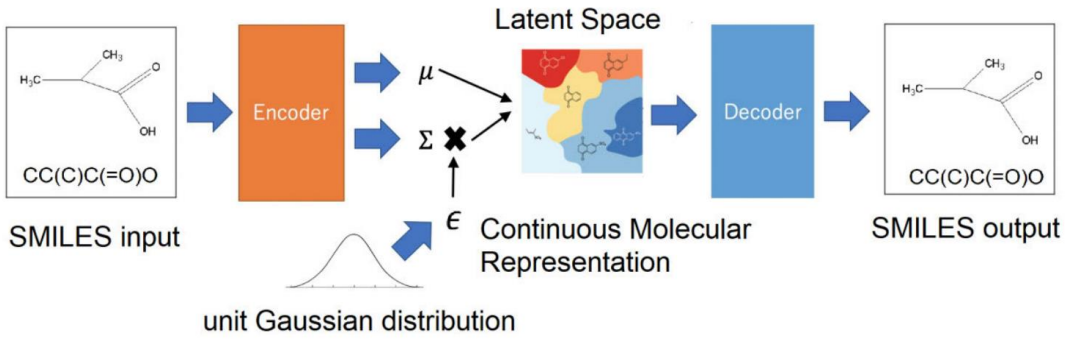

Figure 2.3. Overall generation schema of a variational autoencoder system. Adapted from (106).

2.4.2. GANs

Initially introduced by Goodfellow et al. (129), GANs have recently gained prominence in the field of generative models for their remarkable capability to generate accurate data. Comprising two essential components, namely the discriminator and the generator, GANs operate in a collaborative manner rather than a competitive one, despite the term "adversarial" (40). Together, these components learn features by leveraging each other's capabilities, without any pre-training involved. The generator takes random noise, known as a latent random variable, as input and generates synthetic data samples. The fundamental objective of GANs can be formulated as a minimax game, where the generator strives to maximize the objective by producing data that convincingly deceives the discriminator, while the discriminator aims to minimize the objective by accurately discerning real and fake data (57). Training GANs entails iteratively optimizing the generator and discriminator, refining their strategies to achieve a Nash equilibrium. At this equilibrium point, the generator generates realistic data that the discriminator cannot differentiate from genuine data (58). Nevertheless, during the initial stages of training, GANs may encounter difficulties due to insufficient gradients for the discriminator. This predicament arises when the discriminator is weak, making it easy to identify generated data and leading to gradient saturation. To overcome this challenge,

researchers have proposed modifications such as maximizing an alternative objective function that boosts the gradient (59, 60).

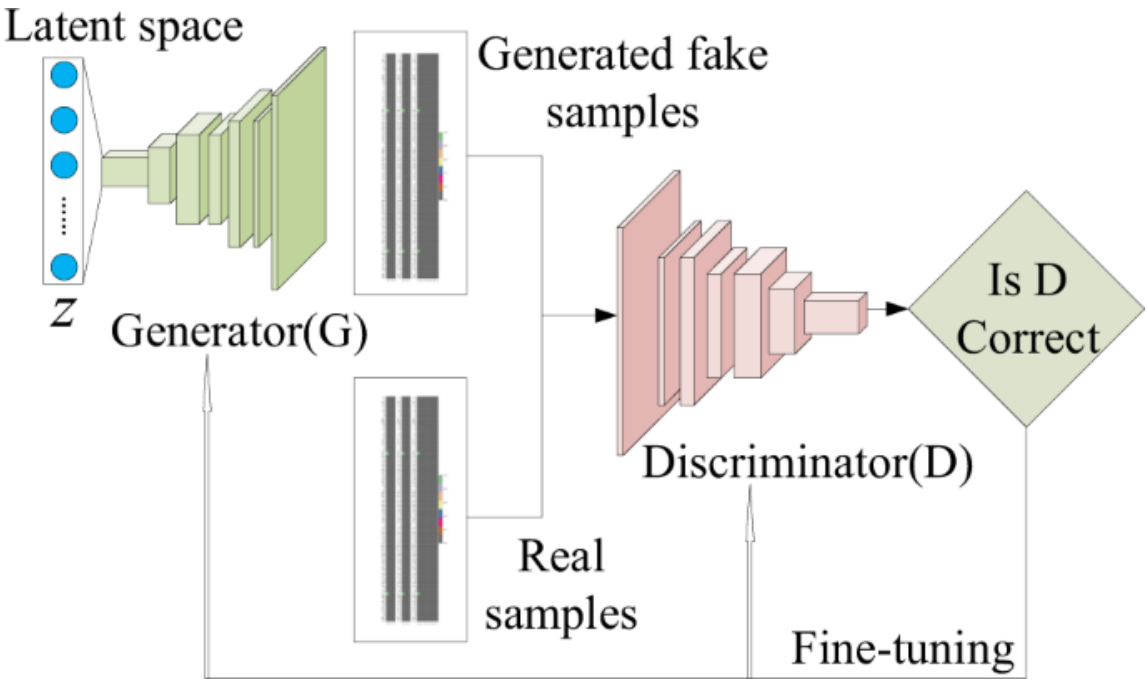

Figure 2.4. Generation process of a generative adversarial network using noise input. Adapted from (107).

2.4.3. Diffusion Models

Diffusion models represent a category of generative models in deep learning that aim to produce synthetic yet realistic data by utilizing input parameters. These models have gained significant traction due to their ability to smoothly learn complex distributions, handle high-dimensional data, and generate diverse samples (61). Traditionally, diffusion models have primarily been employed in continuous state spaces. However, recent advancements have broadened their applicability to include discrete state spaces as well. Discrete diffusion models operate with variables that are discrete, such as text or categorical data, which possess distinct characteristics and present unique challenges. Notably, a key distinction between continuous and discrete diffusion models lies in the treatment of noise. Continuous diffusion models utilize additive Gaussian noise

to perturb the data, whereas discrete diffusion models introduce discrete perturbations or transformations to modify the discrete states. This approach facilitates exploration of different states within discrete space and enhances the variety of generated samples (62). Moreover, transition probabilities in continuous and discrete diffusion models also differ. Continuous models rely on stochastic differential equations to define transition probabilities between states, while discrete models employ conditional distributions that capture dependencies between current and previous states. This enables information propagation and guides the diffusion process within the discrete state space (63, 64). By extending diffusion models to discrete state spaces, researchers can leverage these models to address generative tasks involving text or categorical data. The specific adaptations and techniques employed in discrete diffusion models enable effective modeling and generation of diverse samples within the discrete domain, opening new possibilities for applications in natural language processing and other domains involving discrete variables (61).

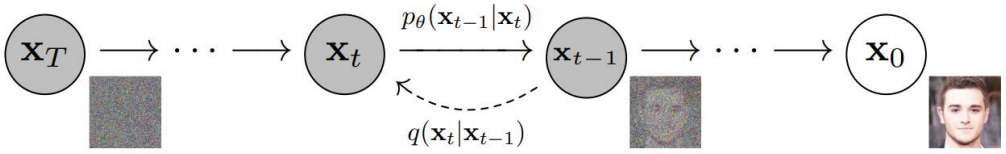

Figure 2.5. Diffusion process to noise and denoise given input. Adapted from (108).

2.4.4. Transformers

Transformers have emerged as a powerful tool in the field of deep learning, making significant contributions across various domains such as language understanding, image processing, and information retrieval. As a result, substantial research efforts have been devoted to enhancing the fundamental aspects of Transformers and developing more efficient variations. One key feature of Transformer models is the self-attention mechanism, which can be understood as a graph-like inductive bias that connects all

tokens in a sequence through relevance-based pooling (65). The Transformer architecture consists of multiple components within its blocks, including a multi-head self-attention mechanism, a position-wise feed-forward network, layer normalization modules, and residual connections. The multi-head self-attention mechanism plays a crucial role in the Transformer model by allowing each element in the sequence to learn how to gather information from other tokens within the same sequence. Essentially, the self-attention mechanism facilitates effective information exchange and aggregation among tokens, enabling the model to capture intricate dependencies and relationships within a sequence (65, 66).

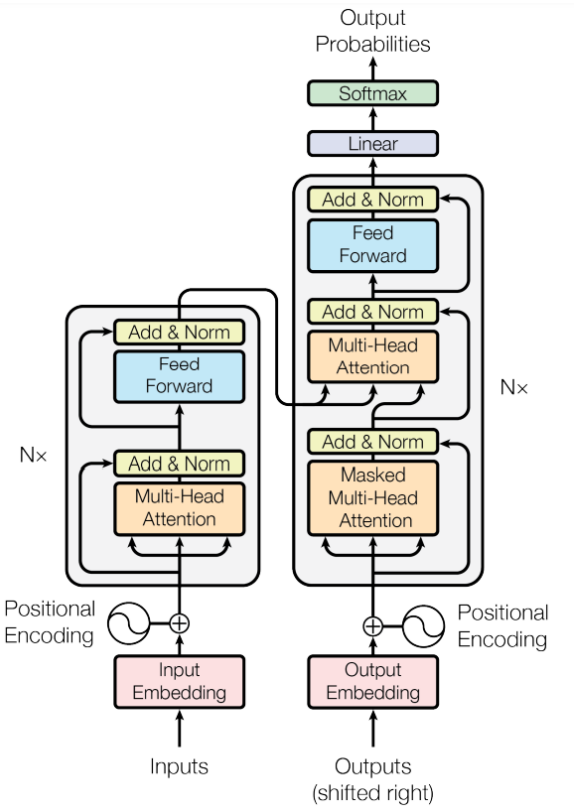

Figure 2.6. Transformer architecture that is proposed in the original study. Adapted from (94).

2.4.5. Normalizing Flow-based Models

Normalizing Flows involve transforming a simple probability distribution, such as a standard normal, into a more complex distribution using a sequence of invertible and differentiable mappings. By evaluating the density of a sample through inverse transformations and accounting for the associated change in volume, new families of distributions can be constructed. This approach allows for sampling from the transformed density and computing the likelihood of a sample (67). In order for normalizing flows to be practical and effective in various applications, they should be invertible, expressive, and computationally efficient. This approach results in a framework for constructing new families of distributions by employing a series of parameterized, invertible, and differentiable transformations. The process begins with an initial density, and then a sequence of transformations is applied to create a new density. (110). Normalizing flows can also be applied to graphs. A study proposes a novel approach to graph neural networks (GNNs) by expanding upon the concept of normalizing flows specifically tailored for graph-structured data called Graph normalizing flows (GNFs). GNFs possess a noteworthy characteristic: the message passing computation is entirely reversible, allowing for the precise reconstruction of input node features from the GNN representation (111). Graph representation for normalizing flows also extended to molecule generation process. A recent study proposes a model that is called SiamFlow where normalizing flows are leveraged for molecular generation. SiamFlow focuses on aligning the flow with the distribution of target sequence embeddings in latent space. This is achieved by employing an alignment loss and a uniform loss, which encourage agreement between target sequence embeddings and drug graph embeddings while preventing collapse (109).

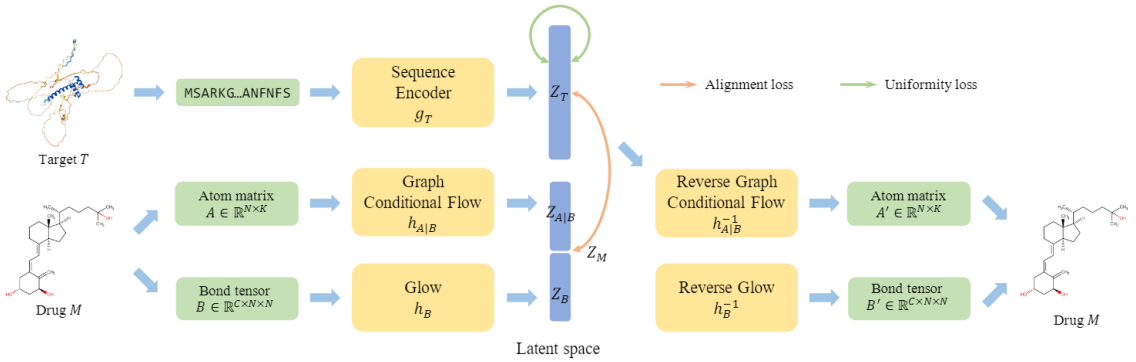

Figure 2.7. A double normalizing flow system that is used to generate target specific drug candidates. Adapted from (109).

2.5. De Novo Molecule Design

De novo molecular design, also known as generative chemistry, describes the process of automatically generating new chemical structures that meet specific criteria for desired biological responses and acceptable pharmacokinetic properties, primarily applied in drug discovery (7). The underlying goal of de novo drug design is to identify new drug candidate molecules that are structurally and biologically distinct from approved drugs on the market and drug candidate compounds with information available in chemical databases. This approach aims to overcome the diversity problem in existing small molecules. Classical de novo drug design is a manual procedure performed by medicinal chemists, involving modifications (additions and deletions) of atoms and bonds on existing structures to generate different molecules. Subsequently, the new molecule is synthesized and subjected to experimental processes. The directed approach used in classical de novo drug design produces reliable results, but experiments are time-consuming, the output is limited in scale, and structural diversity remains below the desired level due to reliance on known scaffold structures (75). Deep generative algorithms, such as RNNs, GANs, and GNNs, have been effectively employed in de novo drug design, demonstrating the ability to learn the probability distribution of chemical structures and generate molecules with desired properties. Techniques like transfer learning and reinforcement learning have further enhanced the fine-tuning of pre-

trained models to guide the generation of molecules with specific characteristics. Additionally, advancements in 3D generative models, including those incorporating molecular property constraints and utilizing VAE models, have aimed to generate high-quality and diverse drug-scale molecules (68). The MGM model is based on message-passing neural networks (MPNN) and employs a masked graph model to learn a distribution over graphs. It captures conditional distributions over unobserved nodes and edges based on observed ones (98).

2.5.1. Goal-Oriented Molecule Design

Goal-oriented molecular design aims to produce molecules that have specific characteristics such as desired synthetic accessibility (SA), drug-likeness, or water solubility. An approach called CVAE combines the benefits of latent space utilization with the incorporation and manipulation of molecular properties during the encoding and decoding processes. The CVAE model can generate drug-like molecules that meet specific target properties, allowing for control over individual properties while keeping others unchanged and even generating molecules with properties outside the database range (69). Another method uses transcriptomic data to train a generative adversarial network, that can automatically generate molecules with a high likelihood of inducing a desired transcriptomic profile, offering advantages such as the ability to design hit-like molecules without prior knowledge of active compounds, biological activity data, or target annotations. This multifunctional approach allows the same model to design molecules for multiple targets or biological states (70). The MolGPT model employs the Generative Pre-trained Transformer (GPT) framework to create novel molecules, utilizing the SMILES representation. Its primary focus lies in the generation of molecules aligned with specific objectives. This model employs conditional codes to enhance molecules according to predefined metrics. Furthermore, the model integrates molecular scaffold data, effectively steering the process of molecule generation to adhere to specific scaffold structures. Notably, the model demonstrates comparable performance to its previously published models (99). The REINVENT model is a reinforcement learning-based method

for fine-tuning a sequence-based generative model. It utilizes augmented episodic likelihood to generate structures with specific desired properties (101). MARS is a method that iteratively modifies fragments of molecular graphs using graph neural networks (GNNs) to generate chemical candidates (102). BIMODAL introduces generative Recurrent Neural Networks (RNNs) for molecule design based on SMILES representations. It combines two bidirectional methods and introduces a novel approach called bidirectional molecule design by alternate learning (103). Molecule Deep Q-Networks (MolDQN) is a model for molecule optimization that combines domain knowledge of chemistry with reinforcement learning techniques, specifically double Q-learning and randomized value functions. It directly defines modifications on molecules to ensure 100% chemical validity (104).

2.5.2. Target-Based Molecule Design

In recent years, computational methods have played a crucial role in accelerating drug discovery by employing deep generative models for de novo drug design. These models can be categorized based on their utilization of target information, with one group using known compounds for guidance and the other leveraging target structure. However, the limitations include the requirement of target-specific molecules or predictive models, scarcity of structural information for targets, and small training datasets, which hinder the models' generalizability (71). Generating molecules that specifically bind to protein binding sites using machine learning approaches poses several challenges. Firstly, there is the complexity of capturing both the 3D geometric structure and the chemical features of the binding site as important contextual information. Secondly, the enormous chemical space and the rarity of molecules with binding ability to specific targets make it challenging to explore and generate relevant molecules. Additionally, ensuring that the generative model is equivariant to rigid transformations of the binding site, meaning the generated molecules should behave consistently when the binding site is rotated or translated, is another important consideration (72). Several approaches have been developed to incorporate 3D molecular geometries into deep

learning-based generative architectures for drug design. However, while these models consider ligand and protein features, they often fail to address important aspects such as ligand binding patterns and pharmacophore features in the generated molecules (73,74). The RELATION model introduces a generative model based on 3D representations. It incorporates the bi-directional transfer learning (BiTL) algorithm to extract and transfer desired geometric features of protein-ligand complexes into a latent space for generation (100). QADD is a de novo drug design approach that integrates an iterative refinement framework with a graph-based molecular quality assessment model. It evaluates the drug potentials of generated molecules based on multiple objectives (105).

2.6. Protein Target Used in the Study

In this thesis, the AKT protein has been chosen as the designated target for the training of the system. The AKT protein is a conserved serine-threonine kinase that and an integral part of the PI3K/AKT signaling pathway, which regulates various cellular processes, including cell growth, proliferation, survival, and metabolism (78). The dysregulation of AKT signaling, characterized by its hyperactivation, is frequently observed in many cancer types. This aberrant activation leads to enhanced cell survival and uncontrolled proliferation. AKT comprises three isoforms: AKT1, AKT2, and AKT3, each with distinct functions and tissue-specific expression patterns. AKT1 is implicated in breast and ovarian cancers, where it promotes cell survival and resistance to apoptosis (79). AKT2 is frequently overexpressed in pancreatic, colorectal, and ovarian cancers, contributing to tumor growth and metastasis (80). AKT3 has been associated with melanoma and lung cancer, playing a role in cell proliferation and survival (81). Targeting the AKT signaling pathway has emerged as a potential therapeutic strategy in various cancers, including breast, ovarian, pancreatic, and hepatocellular carcinoma. Inhibiting this pathway could suppress tumor growth and enhance the effectiveness of other treatments. Several small molecule inhibitors targeting AKT have been advanced to clinical trials (82).

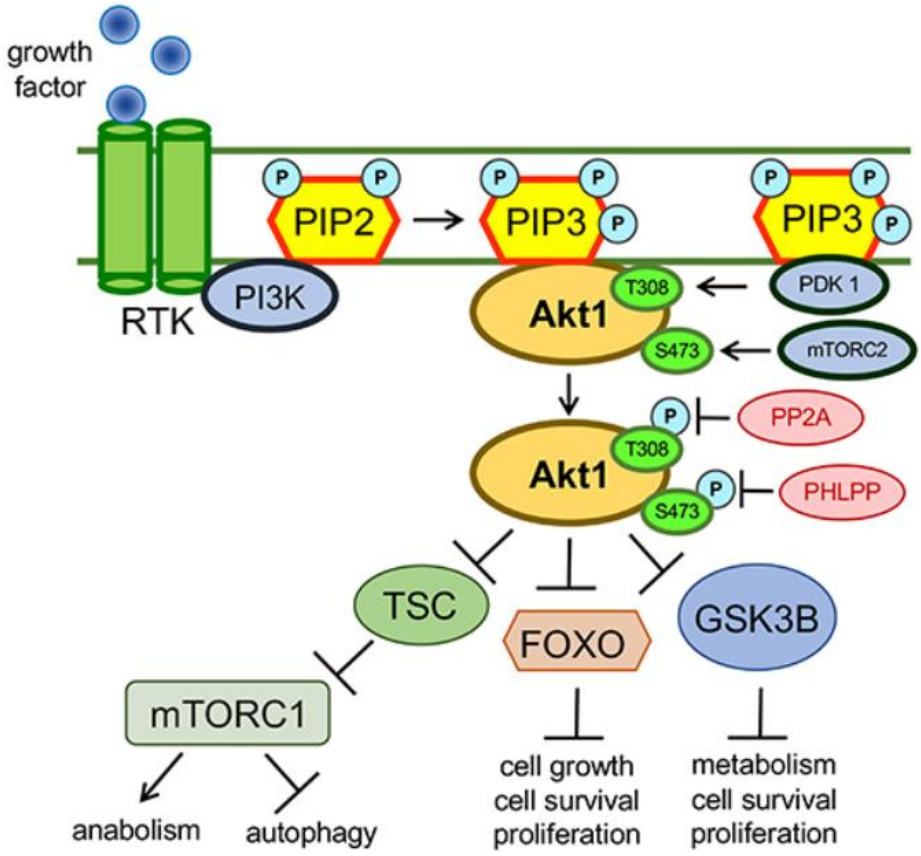

Figure 2.8: Pathway and molecular function of AKT1 protein. Adapted from (95).

3. MATERIALS AND METHODS

3.1. Data Preparation

In this thesis, a dataset of 83 biological assemblies extracted from the Protein Data Bank (PDB) was utilized. The primary emphasis was placed on the RAC-alpha serine/threonine-protein kinase (AKT1), which belongs to the non-specific serine/threonine protein kinase class (EC number: 2.7.11.1). Out of the extensive collection of 57,925 models of biological assemblies within the PDB, we carefully selected the ones that are associated with our target protein. The AKT1 protein is predominantly composed of two domains: the kinase domain and the pleckstrin homology (PH) domain. The utilization of experimental bioactivities was pivotal in the training process of our DrugGEN system. These bioactivities encompass the quantitative assessment of the physical interactions between compounds resembling drugs and their respective target proteins. To ensure standardization, we retrieved the ligand data from the ChEMBL database and implemented several filters. These filters were employed to select only specific criteria, including "single protein" targets, "binding assay" assay types, "standard" measurement types, and non-null pChEMBL values.

The compounds dataset contains SMILES representations of the molecules and served as the input for both the GAN1 and GAN2 modules, representing our "real" samples. For this study, we accessed the compound dataset from ChEMBL, specifically ChEMBL v29, which consists of a total of 1,914,648 small molecules.

During subsequent analysis, attention was redirected towards the ligand data linked to the AKT1 target protein. The ultimate dataset comprised interactions between ligands and the human AKT1 protein (CHEMBL4282), with a pChEMBL value of 6 or higher (equivalent to IC50 ≤ 1 μM). Moreover, we integrated the SMILES notations for these ligands. To enhance our activity dataset, we included 87 drug molecules from the DrugBank database that are recognized for their interaction with the human AKT1

protein. We further applied a filter to exclude molecules with more than 45 heavy atoms, resulting in approximately 2,582 molecules for training.

3.1.1. Data Statistics

The training of the model involves three distinct types of data: ChEMBL molecular data, ligand data, and protein data. The compound dataset utilized in the training process was carefully selected from the ChEMBL database. It consists of a total of 1,914,648 small molecules. To optimize the model's performance, the dataset was curated by setting a maximum limit of 45 heavy atoms for each molecule. After applying this filter, the remaining set contains 1,588,865 molecules. Detailed statistical summaries for both datasets, including various metrics, can be found below for examination.

Table 3.1. Statistical summary of the compound and ligand datasets.

Dataset	QED	logP	SA	MW	Heavy Atom
ChEMBL	0.541 ± 0.222	3.446 ± 2.029	2.993 ± 0.967	413.205 ± 187.211	29.350 ± 13.090
AKT	0.460 ± 0.180	4.071 ± 1.737	3.051 ± 0.491	466.019 ± 95.946	33.734 ± 7.051

The given table provides statistical information on two datasets: ChEMBL and AKT. These datasets contain several parameters for chemical compounds, including QED (Quantitative Estimation of Drug-likeness), logP (Octanol-water partition coefficient), SA (Surface Area), MW (Molecular Weight), and Heavy Atom count. In terms of QED, ChEMBL has a mean value of 0.541 with a standard deviation of 0.222, while AKT has a slightly lower mean of 0.460 with a standard deviation of 0.180. This suggests that, on average, the compounds in ChEMBL may exhibit a slightly higher drug-likeness estimation compared to AKT. Regarding logP, ChEMBL has a mean of 3.446 with a standard deviation of 2.029, whereas AKT has a higher mean of 4.071 with a standard deviation of 1.737. This indicates that AKT's compounds tend to have a higher octanol-water partition coefficient, suggesting a higher hydrophobicity compared to ChEMBL. For the SA

parameter, ChEMBL has a mean of 2.993 with a standard deviation of 0.967, while AKT has a slightly higher mean of 3.051 with a standard deviation of 0.491. The difference is relatively small, indicating that both datasets have similarity for ease of synthesis. In terms of MW, ChEMBL has a mean value of 413.205 with a standard deviation of 187.211, while AKT has a higher mean of 466.019 with a lower standard deviation of 95.946. This suggests that the compounds in AKT tend to have a higher molecular weight and a narrower range compared to ChEMBL. Finally, looking at the heavy atom count, ChEMBL has a mean of 29.350 with a standard deviation of 13.090, while AKT has a slightly higher mean of 33.734 with a standard deviation of 7.051. This indicates that the compounds in AKT generally have a higher number of heavy atoms compared to ChEMBL.

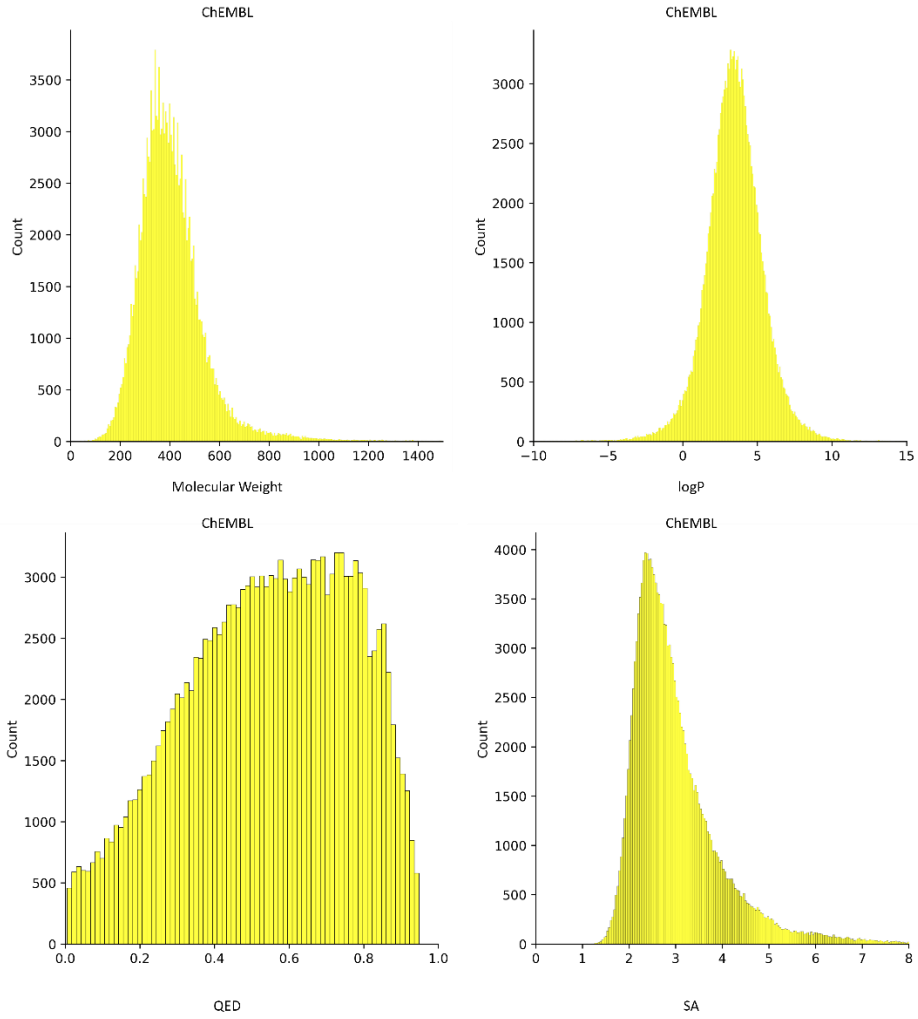

Figure 3.1. Histogram plots of the ChEMBL compound dataset. Left top plot shows the molecular weight distribution, the top right plot indicates the logP distribution, bottom left plot is the QED distribution, and bottom right is the SA distribution of the ChEMBL compound dataset.

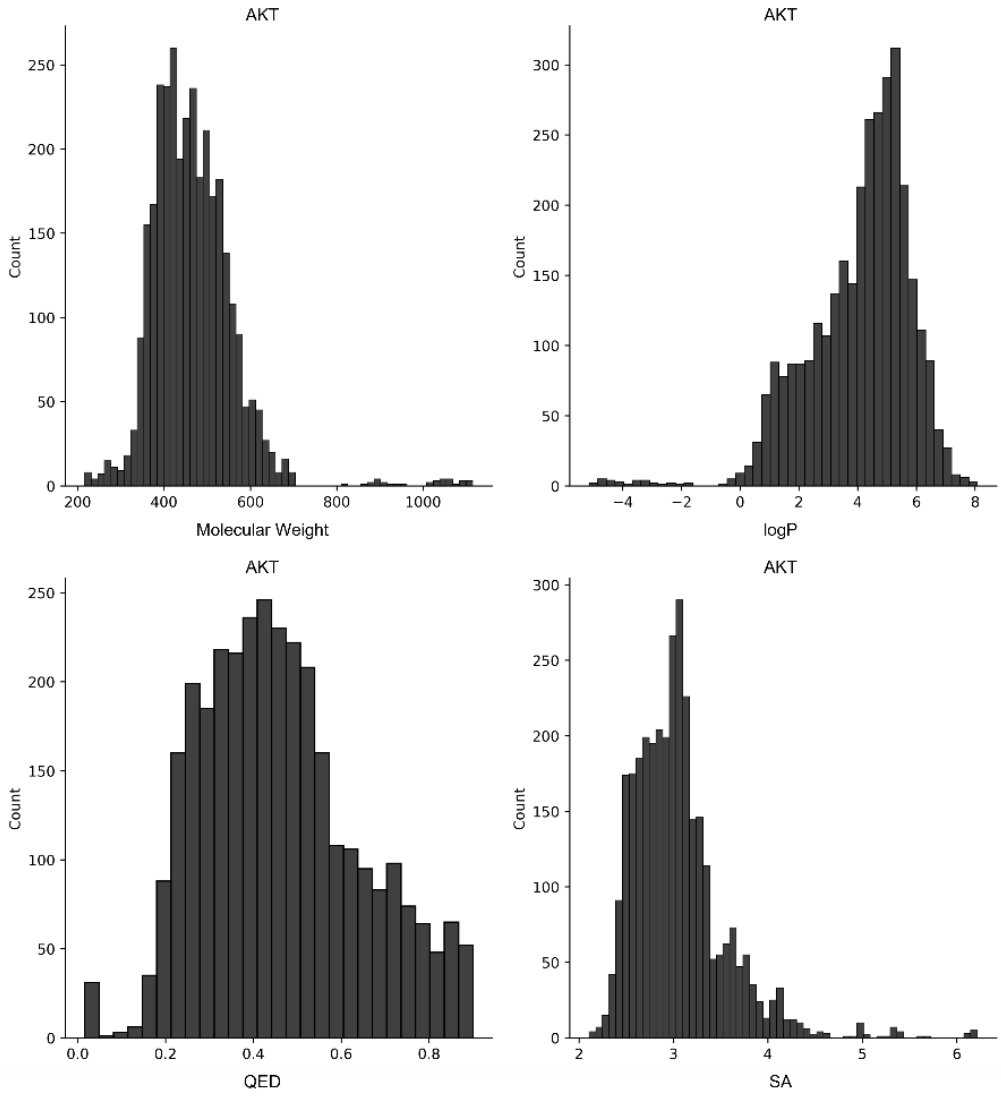

Figure 3.2. Histogram plots of the AKT ligand dataset. Left top plot shows the molecular weight distribution, the top right plot indicates the logP distribution, bottom left plot is the QED distribution, and bottom right is the SA distribution of the AKT compound dataset.

3.1.2. Compound Data

DrugGEN relies on graph representations of input molecules, which consist of two components: an annotation matrix and an adjacency matrix. These matrices capture crucial information about atom types and atomic bonds/interactions, respectively. To generate these matrices, we utilized the RDKit library along with the SMILES notations of the molecules. The annotation matrix represents 8 types of atoms based on PDBQT atom types (C, N, O, F, S, P, Br, Cl), including a category for the null case (i.e., no atoms). The number of rows in the matrix corresponds to the maximum length (number of heavy atoms) of the molecule to be generated, while the number of columns defines the atom types. The adjacency matrix is a two-dimensional matrix that indicates the presence and type of covalent bonds between atoms in the molecule. It has five dimensions representing bond types: 0th (no bond), 1st (single bond), 2nd (double bond), 3rd (triple bond), and 4th (aromatic bond).

3.1.3. Ligand Data

The ligand data shares similar characteristics with the compound dataset, with the difference being that it specifically includes small molecules related to AKT1, AKT2, and AKT3. The featurization process for this dataset follows the same rules as the compound dataset and incorporates the same atomic and bond features. For the summary of features, Table 3.2 can be examined.

3.1.4. Protein Data

Proteins are large biomolecules and directly presenting the entire protein structure to the model would introduce significant computational complexity, making it challenging to train a successful model. To prevent this, we focused on the binding sites/regions. To obtain the binding sites, we utilized the coordinates of protein-ligand complexes sourced from the Protein Data Bank (PDB).

In DrugGEN, target proteins are represented at the atomic resolution to align with compound features. The atom types are standardized to the PDBQT file format, which

utilizes reduced atom types. Additionally, hydrogen atoms were added to proteins to replicate their natural form. To perform these operations, we utilized protein and ligand processing scripts within AutoDockTools4 (88). The determination of protein atoms involved setting a cutoff distance between protein and ligand atoms using Euclidean distances, with a value of 9 Angstroms (Å) chosen based on literature (89). Thus, atoms within a maximum distance of 9 Å from any ligand atom were considered part of the binding site.

Protein adjacency matrices were constructed to accurately represent the protein structure, encompassing both covalent bonds and non-covalent interactions between atoms. The PDBeChem web service (https://www.ebi.ac.uk/pdbe-srv/pdbechem/) was employed to define existing bonds at the protein's binding site. The Python library Interfacea (https://github.com/JoaoRodrigues/interfacea/tree/master) was utilized to identify non-intrinsic covalent interactions, including intra-residue and inter-residue atoms. As a result, the annotation matrix for the AKT1 protein was constructed. It includes a total of 450 atoms belonging to seven types: C (aliphatic carbon), N (non-H-bonding nitrogen), OA (acceptor 2 H-bonds oxygen), A (aromatic carbon), SA (acceptor 2 H-bonds sulfur), NA (acceptor 1 H-bond nitrogen), HD (donor 1 H-bond hydrogen), along with an additional type to represent the absence of atoms. The adjacency matrix contains four types of covalent bonds and six types of non-covalent bonds: ionic, hydrogen bond, cation-π, hydrophobic, parallel π-π stacking, and t-shaped π-π stacking.

Table 3.2. Atom and bond types that were used in the study.

Data	Atom Types	Bond Types
ChEMBL	C, N, O, F, S, P, Br, Cl	No bond, single, double, triple, aromatic
Ligand	C, N, O, F, S, P, Br, Cl	No bond, single, double, triple, aromatic
AKT1	C, N, OA, A, SA, NA, HD	ionic, hydrogen, cation-π, hydrophobic, parallel π-π stacking, t-shaped π-π stacking

3.2. Architecture

DrugGEN is a stacked generative model specifically designed for target-based drug candidate molecule design. The primary objective of the DrugGEN system is to generate molecules that are both novel and tailored to interact with a selected protein. The model employs multiple Generative Adversarial Networks (GANs) to divide the molecule generation process into distinct tasks. The DrugGEN model consists of two stacked GAN modules, referred to as GAN1 and GAN2. Each GAN module comprises a generator submodule (G1, G2) and a discriminator submodule (D1, D2). The first generator, G1, takes a molecular input and generates novel molecules that are learned from the statistical distribution of a molecular dataset. The first discriminator, D1, compares the generated novel molecule candidates with existing molecules and guides G1 to explore valid molecular space.

The output of G1 is then passed to the second generator, G2, which transforms the novel molecule to serve as an interaction partner for the selected target protein. The transformation process incorporates protein data, enabling the redesign of the molecular data based on the target protein. Subsequently, the finalized molecule is compared to experimentally validated inhibitors of the chosen target in the second discriminator, D2. This step assists in guiding the generation process of G1 to better match the statistical distribution of the validated inhibitors. By utilizing this stacked GAN architecture and incorporating protein data, DrugGEN aims to generate novel molecules that exhibit desired interactions with specific protein targets. The iterative process involving the generators and discriminators enables the model to refine the molecule generation process and enhance the alignment with known inhibitors of the target protein.

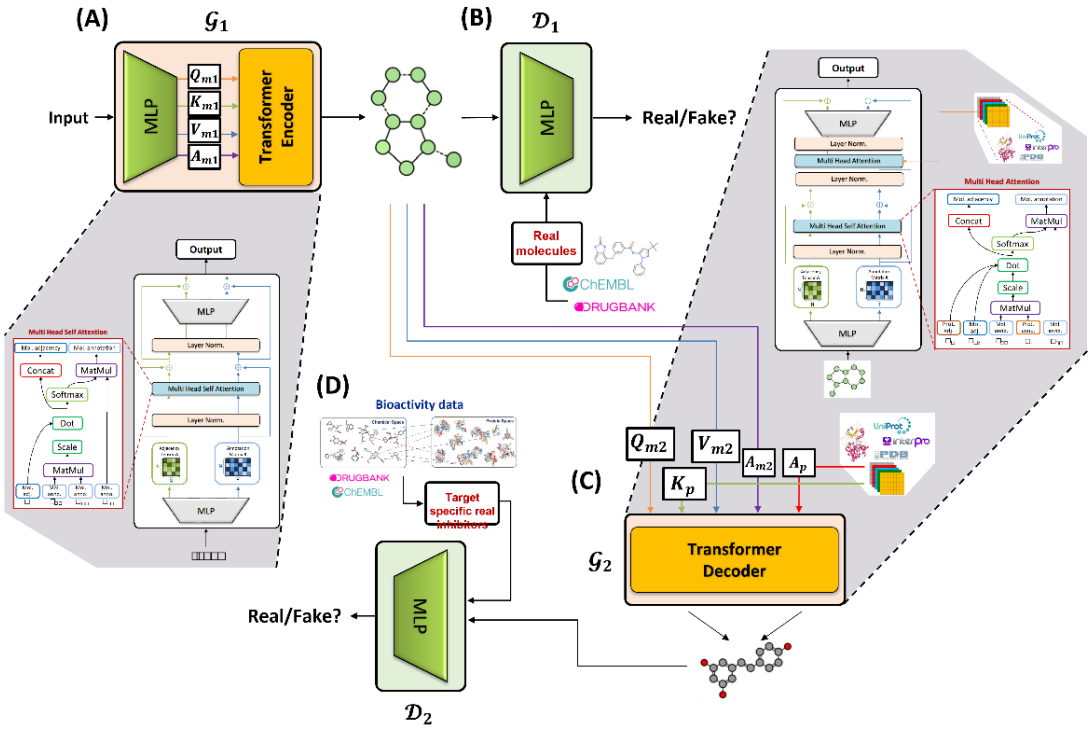

Figure 3.3. Target aware molecule generation schema of the DrugGEN, adapted from Unlu et al., (2023). Part A defines the graph transformer encoder generator while part B indicates the MLP discriminator of the GAN1 system. Part C is the graph transformer decoder of the GAN2 system where proteins and molecules processed together. Part C is the MLP discriminator where finalized generated molecules are compared with experimentally validated inhibitors.

3.2.1. GAN1

In the DrugGEN model, the first GAN (GAN1) is responsible for designing novel molecules based on the learned molecular space. GAN1 consists of two submodules: the generator submodule (G1) and the discriminator submodule (D1). These submodules engage in an adversarial training process, characteristic of GANs. During training, G1 aims to generate molecules that can deceive D1 into classifying them as real molecules, while D1 strives to accurately discriminate between real molecules and those generated by G1. This adversarial dynamic between the generator and discriminator leads to an iterative improvement of both submodules. G1 becomes more adept at generating novel

molecules, while D1 becomes more skilled at distinguishing real molecules from the generated ones.

The input for GAN1 is the set of real molecules obtained from the molecular dataset. G1 utilizes these real molecules to learn the underlying graph structure of the existing molecules, capturing the essential features and patterns. Meanwhile, D1 evaluates the generated molecules produced by G1 and discriminates between real and generated molecules. This discrimination process provides feedback to G1, encouraging it to generate novel molecules that do not exist in the training data, while still ensuring that the generated molecules are valid and realistic. Through this adversarial training process, GAN1 of the DrugGEN model enables the generation of novel molecules that extend beyond the known training data. The interplay between the generator (G1) and the discriminator (D1) facilitates the exploration of the molecular space and the production of valid, yet previously unseen molecules.

3.2.2. Graph Transformer Encoder Generator

The generator network of GAN1 in the DrugGEN model utilizes a graph transformer network to process the given input, which is graph-structured data representing molecules. The graph transformer network is derived from the classical transformer encoder architecture, which was originally designed for text-based inputs to handle words and sentences. In the case of the graph transformer encoder, the input consists of annotation and adjacency matrices, which contain atom and bond information of the molecules. The graph transformer encoder block follows a similar structure to the transformer encoder block used for text inputs. The graph transformer encoder block includes several components. Residual connection layers enable the information from the previous layer to be passed directly to the next layer, preserving important information in the network. Next, a graph multi-head attention layer is used, and unlike the attention mechanism used in the classical transformer encoder, the multi-head attention layer in the graph transformer encoder is specifically designed to handle

graph-structured data. It utilizes annotation and adjacency matrices to compute attention weights, allowing the model to capture relationships between atoms and bonds in the molecules. Feed forward layer, at the end, applies a non-linear transformation to the output of the multi-head attention layer, helping the model capture complex patterns and relationships within the graph-structured data.

The graph transformer encoder block repeats these layers multiple times to capture hierarchical representations and refine the learned features. It leverages the residual connections to enable efficient gradient flow during training.

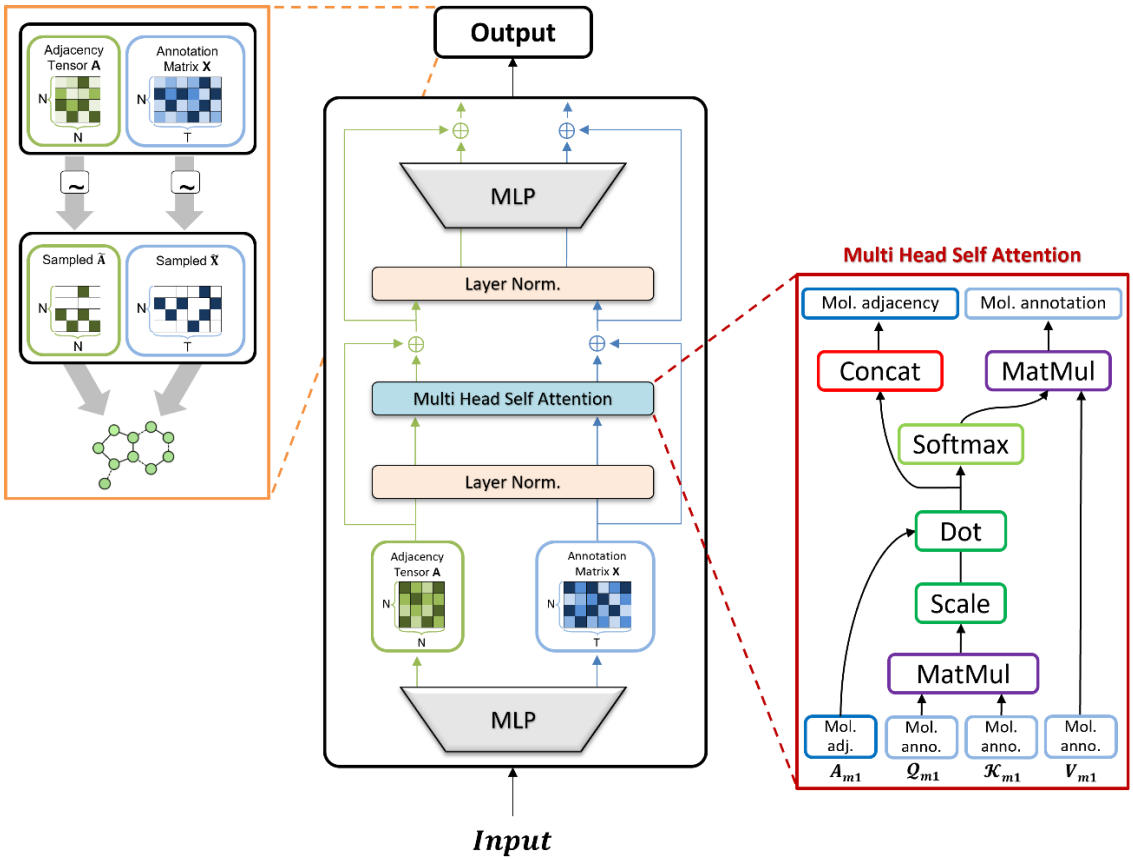

Figure 3.4. Working schema of the graph transformer encoder network.

By using the graph transformer network as the generator network in GAN1, the DrugGEN model can effectively process the graph-structured input data and generate

novel molecules that meet the desired criteria for interaction with the chosen protein target. In the DrugGEN model, the processing of the annotation and adjacency matrices occurs within the same module. The input consists of randomly selected real molecule matrices, which are then passed through individual Multi-Layer Perceptron (MLP) for the annotation and adjacency matrices. Each MLP consists of four layers and is responsible for creating embeddings of the respective matrices. These embeddings are designed to have dimensions compatible with the Transformer encoder. Following the embedding step, the input is fed into a Transformer encoder module, which contains one layer with eight multi-head attention heads. The Transformer encoder begins by applying layer normalization to the input. The self-attention mechanism is then utilized, where Qm1, Km1, and Vm1 represent the variables corresponding to the annotation matrix of the molecule. In the traditional Transformer architecture, Q, K, and V variables represent the same input sequence.

In the graph transformer setting of the DrugGEN model, attention weights are calculated differently. The attention weights are determined by multiplying the adjacency matrix Am1 of the molecules with the scaled dot product of Qm1 and Km1. This modified calculation accounts for the graph structure of the molecules. The resulting attention weights are then multiplied with Vm1 to create the final representation of the annotation matrix. For the adjacency matrix, the new representation is formed by concatenating the attention weights (90,91). In the default model configuration, the output dimension size of the Transformer is set to 128 for both the annotation and adjacency matrices. The calculation of the attention mechanism in the DrugGEN model is as follows and can be summed up as in equation below:

$$Attention_1(Q_{m1}, K_{m1}, V_{m1}) = softmax(\frac{Q_{m1}K_{m1}^T}{\sqrt{d_k}} A_{m1})V_{m1}$$

In the equation you provided, Q_{m1}, K_{m1}, and V_{m1} represent the annotation matrix of the molecules, while A_{m1} represents their adjacency matrix. The value dk corresponds

to the dimension of the transformer encoder module and is used to scale the attention weights. Multiplying the attention weights with the adjacency matrix A_{m1} ensures that the contribution of the adjacency information is incorporated into the attention mechanism. By doing so, the model can capture the relationships between atoms and bonds in the molecules and effectively use this information during the generation process.

3.2.3. GAN2

The generator network of the second GAN, GAN2-generator, takes the de novo molecules generated by GAN1 and processes them together with protein features. This incorporation of protein features allows GAN2-generator to consider the specific characteristics and requirements of the target protein while generating new molecules.

3.2.4. Graph Transformer Decoder Generator

The second generative network in DrugGEN, referred to as G2, is responsible for modifying the molecules generated by G1 to make them interact with the target protein. G2 utilizes the transformer decoder architecture, as introduced by Vaswani et al. (94), to perform this task. The transformer decoder module in G2 consists of 8 decoder layers and uses 8 multi-head attention heads. In the default model, both the input and output dimensions of the transformer decoder are set to 128. The input to G2 includes the data generated by G1, denoted as G1(z), and the protein features. The protein features are processed using self-attention mechanisms within the transformer decoder module, similar to how molecules are processed in the previous steps. The interactions between molecules and protein features are handled inside the multi-head attention module of the transformer decoder. The molecules and protein features are multiplied together using the scaled dot product operation, resulting in new molecular matrices. The attention calculation in this context can be represented by the following formula:

$$Attention_2(Q_{m2}, K_p, V_{m2}) = softmax(\frac{Q_{m2}K_p^T}{\sqrt{d_k}}(A_pA_{m2}))V_{m2}$$

In this equation, Q_{m2} represents the queries derived from the molecular matrices, K_p represents the keys derived from the protein features, and V_{m2} represents the values associated with the molecular features. The dot product of Q_{m2} and the transpose of K_p is scaled by the square root of the dimension d_k. The SoftMax operation is applied to normalize the attention weights, ensuring they sum up to 1. These attention weights are then multiplied elementwise with the values V_{m2}, resulting in the final representation of the molecular matrices, considering the interactions with the protein. By incorporating the protein features and calculating the attention between molecules and protein, G2 modifies the molecular matrices generated by G1 to enable them to interact specifically with the target protein, producing molecules that act as binders for the protein.

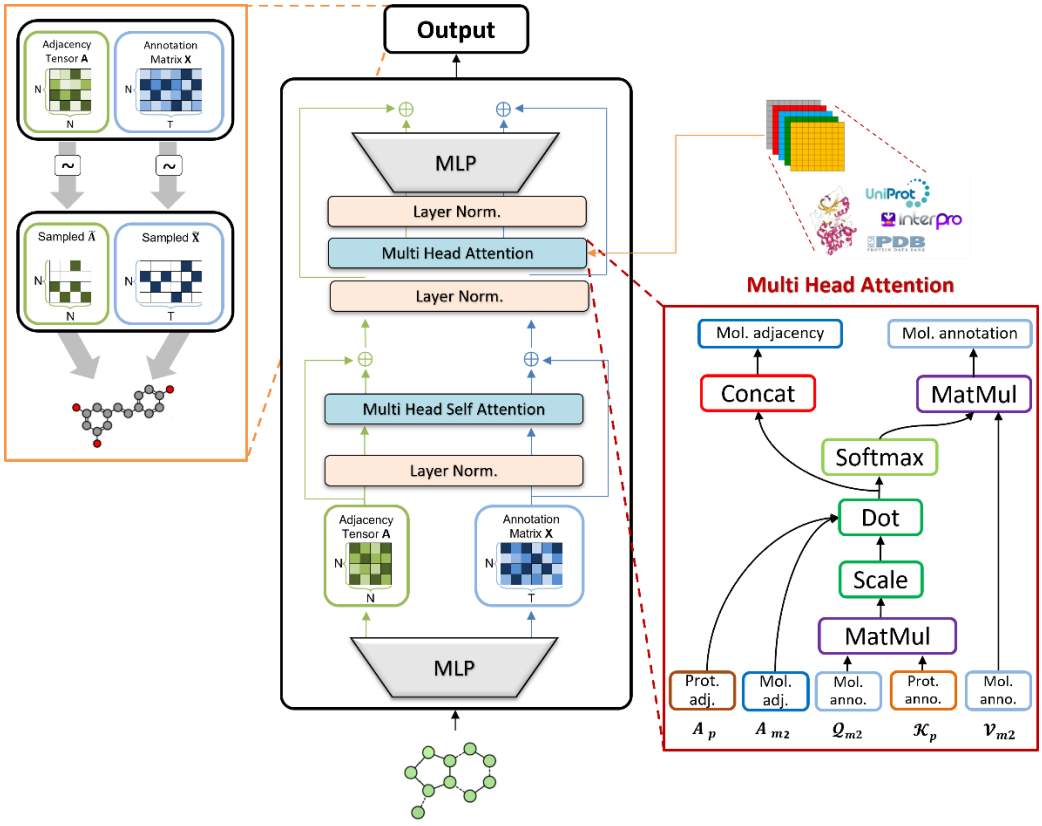

Figure 3.5. Working schema of graph transformer decoder network.

Overall, the transformer decoder architecture in G2 allows for the integration of protein information and facilitates the generation of molecules tailored to interact with the desired protein target.

3.2.5. MLP Discriminator

In DrugGEN, the discriminator plays a crucial role in distinguishing between real and synthetic (fake) data generated by the corresponding generators. The MLP-based discriminators in DrugGEN, namely D1 (used in GAN1) and D2 (used in GAN2), take flattened, one-dimensional vectors as input. These vectors are formed by concatenating the flattened versions of the annotation and adjacency matrices.

Both discriminators, D1 and D2, are independent and do not share parameters. However, they have the same modular structure and size. The layer sizes in the MLP discriminators are as follows: 256, 128, 64, 32, 16, 1, from input to output. The final layer consists of a single neuron with a hyperbolic tangent (tanh) activation function. This activation function maps the output of the discriminator to a value between -1 and 1. The objective of the discriminator is to discriminate between real and generated molecules. Ideally, a perfect discriminator would assign a value of 1 to real molecules and a value of -1 to generated molecules. By training the discriminators in an adversarial manner, they aim to become more effective at distinguishing real and synthetic molecules. The generator modules, G1 and G2, are trained to generate molecules that can successfully deceive the discriminators and receive a high score close to 1, indicating that they resemble real molecules.

By training the discriminators and generators iteratively, the GANs in DrugGEN aim to improve the quality and realism of the generated molecules and enhance the ability to generate molecules that interact specifically with the target protein.

3.3. Ablation Study

In the context of the DrugGEN system, alternative models were developed with variations in their architectural design and input data. These models focus on ligand-

based strategies for generating novel drug candidates, unlike the default model (DrugGEN-Prot) that combines both structure-based and ligand-based approaches.

DrugGEN-Prot (Default Model): This model, as depicted in Figure 1, incorporates protein features into the transformer decoder module of GAN2. It utilizes a combination of structure-based and ligand-based approaches to guide the generation of target-centric de novo molecules. The transformer decoder receives input from both GAN1 (generated molecules) and protein features, enabling the design of molecules specifically tailored for the target protein. The model undergoes end-to-end training, and a single overall loss is computed by combining the losses of both GAN modules.

DrugGEN-CrossLoss: This model consists of a single GAN, specifically GAN1 from the default model. It aims to shift the distribution of the input data towards the distribution of real inhibitors for the target protein within a simpler system. The graph transformer encoder-based generator network takes randomly selected real molecules as input and transforms their molecular structures to generate new molecules that resemble the real inhibitors. The discriminator network distinguishes between the de novo generated molecules and the real inhibitors of the target protein.

DrugGEN-Ligand: Similar to DrugGEN-Prot, this model comprises two GANs and follows the same training routine and hyperparameters. However, instead of using the features of the target protein, it incorporates the features of real inhibitor molecules of the target protein as input to the transformer decoder of GAN2. The objective of the transformer decoder in this model is to generate molecules that exhibit similar properties to the real inhibitors of the target protein. The generation process resembles machine translation, where the model transforms given molecules into inhibitor-like compounds.

DrugGEN-RL is another variant of the DrugGEN system that shares a similar overall architecture with DrugGEN-Ligand. The main objective of DrugGEN-RL is to design structurally diverse de novo molecules by avoiding the utilization of molecular scaffolds that are already present in the training set. To achieve this, DrugGEN-RL incorporates an

additional penalty term in the loss function. The purpose of this penalty term is to decrease the Tanimoto scaffold similarity, measured using the Bemis-Murcko framework, between the generated molecules and the molecules in the training set. In GAN1, the training set comprises molecules from the ChEMBL database, and in GAN2, the training set consists of real inhibitors of the given target protein. By incorporating the scaffold similarity penalty term, DrugGEN-RL encourages the generation of molecules that possess unique structural features and differ from the molecular scaffolds present in the training set. This modification helps promote the exploration of novel chemical space and enhances the diversity of generated molecules by discouraging the generation of molecules with similar molecular scaffolds to those already seen in the training data.

DrugGEN-NoTarget is a simplified version of the DrugGEN system that focuses solely on learning the chemical properties of real molecules from the ChEMBL training dataset. It does not involve target-specific generation or incorporate protein features into the model. The architecture of DrugGEN-NoTarget consists of only one GAN, specifically GAN1 from the default model. The purpose of this model is to capture and learn the statistical distribution and chemical properties of the molecules present in the ChEMBL dataset. By training on the ChEMBL dataset without considering any target-specific information, DrugGEN-NoTarget aims to generate novel molecules that possess desirable chemical properties and adhere to the learned distribution of the training data. DrugGEN-NoTarget employs the same hyperparameters as the default model, ensuring consistency in the training process and enabling direct comparisons with other models within the DrugGEN framework.

These alternative models provide different perspectives on ligand-based and structure-based drug candidate generation, exploring variations in input data and design strategies while leveraging the GAN framework and graph transformer architectures of the DrugGEN system.

3.4. Training

The training of DrugGEN employs the Wasserstein Generative Adversarial Network (WGAN) loss, which is specifically reformulated for the end-to-end training of a two-stage GAN system. The loss function combines the losses of the two discriminators in DrugGEN and incorporates a gradient penalty (GP) to further enhance performance.

The WGAN loss function, denoted as L, consists of four terms, as shown in Eq3. The first term represents the difference between the average discriminator output for real molecules, D1(x), and the average discriminator output for molecules generated by GAN1, D1(G1(z)). The second term represents the average discriminator output for real molecules that interact with the selected target protein, D2($\tilde{x}$), while the third term represents the average discriminator output for molecules generated by GAN2 with inputs G2(G1(z), (Kp, Ap)), where Kp and Ap are the annotation matrix and adjacency matrix of the protein, respectively.

$$\mathrm{L} = (\mathrm{E}_{\mathrm{x}\sim \mathrm{p}_r(\mathrm{x})}[\mathrm{D}_1(\mathrm{x})] - \mathrm{E}_{\mathrm{z}\sim \mathrm{p}_g(\mathrm{z})}[\mathrm{D}_1(\mathrm{G}_1(\mathrm{z}))] + \mathrm{E}_{\bar{\mathrm{x}}\sim \mathrm{p}_r(\bar{\mathrm{x}})}[\mathrm{D}_2(\bar{\mathrm{x}})] - \mathrm{E}_{\mathrm{K}\sim \mathrm{p}_g(\mathrm{K})}[\mathrm{D}_2(\mathrm{G}_2(\mathrm{G}_1(\mathrm{z}), \mathrm{K}))])$$

To improve the performance of the WGAN, a gradient penalty (GP) is introduced, as shown in Equation (4). The GP loss, denoted as L_{GP}, penalizes the gradients of the discriminator with respect to the interpolated samples, denoted as $\bar{\mathrm{x}}$, which are drawn from the real data distribution pr and the generator data distribution pg. The penalty coefficient λ is used to control the strength of the penalty.

$$\mathrm{L}_{GP} = \mathrm{E}_{\hat{\mathrm{x}}\sim \mathrm{p}_{\hat{X}}(\hat{\mathrm{x}})}[(||\nabla_{\hat{X}}\mathrm{D}(\hat{\mathrm{x}})||_2 - 1)^2]$$

By combining the WGAN loss (L) and the GP loss (L_{GP}), the final loss function, denoted as L_{Total}, is obtained as shown in Equation (5):

$$\mathrm{L}_{Total} = \mathrm{L} + \mathrm{L}_{GP}$$

The total loss function is used to optimize the parameters of the DrugGEN model during training, encouraging the generation of novel molecules and improving the discrimination between real and generated molecules.

DrugGEN was trained using the ChEMBL compounds dataset, which served as the input of real molecules for the model. The dataset was randomly split into training and test partitions with a ratio of 90% for training and 10% for testing. The training procedure of DrugGEN was carried out as follows, the training began with discriminator D1 and D2 and then continued with generator G1 and G2. For the default model, a learning rate of 0.00001 was used for all components: G1, G2, D1, and D2. The batch size was set to 128, and the model was trained for a total of 50 epochs. It was observed that the loss values did not significantly change after 50 epochs. The Adam optimizer was utilized as the optimizer for the model, with beta1 set to 0.9 and beta2 set to 0.999. The training process for each model took approximately 2 days to complete, utilizing 10 Intel Xeon Gold 5215 CPU cores and a single NVIDIA A5000 GPU. The number of parameters in G1 of DrugGEN was around 37 million, while G2 had approximately 640 million parameters. Both discriminators, D1 and D2, had around 2.7 million parameters.

3.5. Performance Metrics

The performance of the models in generating molecules was evaluated using various molecular generation metrics from the MOSES benchmark platform. These metrics provide insights into the quality and diversity of the generated molecules. Here are the metrics used:

Validity measures the percentage of generated molecules that can be successfully parsed by the SMILES conversion function of the RDKit Python package. Higher validity indicates a higher percentage of syntactically valid molecules.

Uniqueness quantifies the dissimilarity of each generated molecule with respect to other molecules in the same batch. It ensures that the generated molecules are structurally diverse.

Internal Diversity (IntDiv) measures the mean dissimilarity between a generated molecule and other molecules in the same batch. It is typically computed using Tanimoto similarity based on molecular fingerprints (e.g., ECFP). Higher IntDiv indicates a greater structural diversity among the generated molecules.

Novelty calculates the ratio of generated molecules that are not present in the real (training) dataset to the total number of generated molecules. It assesses the ability of the model to generate novel molecules that differ from the molecules in the training set.

In addition to these metrics, other measures such as Quantitative Estimate of Drug-likeness (QED), partition coefficient (logP), synthetic accessibility (SA), and Frechet-ChemNet Distance (FCD) are used to evaluate the fitness of the generated molecules as potential drug candidates. These metrics assess various properties related to drug-likeness, chemical properties, and similarity to known molecules. he basis for the QED measure's empirical reasoning lies in the inherent distribution of molecular characteristics, encompassing factors like molecular weight, logP, topological polar surface area, count of hydrogen bond donors and acceptors, quantity of aromatic rings and flexible bonds, and the existence of undesired chemical functionalities. logP metric calculates the ratio of water and octanol solubility based on functional groups. SA is the heuristic evaluation indicating the level of difficulty (rated 10) or simplicity (rated 1) in synthesizing a specific molecule. The SA score is derived from a fusion of fragment-based contributions pertaining to the molecule. The calculation of FCD involves utilizing the activations from the second-to-last layer of a deep neural network called ChemNet. This network is trained to forecast the biological activities of pharmaceutical compounds (96, 97).

3.6. Secondary Design

The DrugGEN model was built with a different approach and there were several design choices that were considered for model.

Initially, the implementation of the DrugGEN model involved using noise as input. So that, the generator of the GAN1 would take a Gaussian noise input and tries to transform it to a valid molecule. The discriminator of the GAN1 was getting real molecules from ChEMBL compound set as input. System's training was the same as it was told in section 3.4. The generative network of the DrugGEN was implemented with graph transformer architecture. Before that, classic transformer network was used to create generative network. In this model annotation and adjacency matrices were multiplied in a linearized way resembling the sequential structure of a text representation. Attention between the annotation and adjacency matrices were calculated as described in the Vaswani et al. (94). The DrugGEN model initially attempted to use a graph neural network (GNN) as the discriminator to differentiate between real and generated molecules. In this model, a standard GNN was used from the Kipf et al. (45) study. This GNN would process the graph to create graph-level and atom-level predictions for the given input. In this model, the graph-level predictions were used to train the DrugGEN model. Predictions were between -1 and 1, which indicates whether and input was a real graph or a generated one.

4. RESULTS

Assessing the performance of DrugGEN in designing de novo molecules is crucial for evaluating its generative capabilities. In this context, several benchmarking metrics were employed to measure the quality and characteristics of the generated molecules. Additionally, a comparison was made between the de novo molecules and real molecules using physicochemical property value distribution plots and UMAP embeddings for visualization.

The benchmarking metrics, as previously mentioned, include validity, uniqueness, internal diversity (IntDiv), and novelty. These metrics provide quantitative assessments of the quality and diversity of the generated molecules. Validity measures the proportion of generated molecules that are syntactically valid, while uniqueness ensures that the generated molecules are structurally distinct from each other. Internal diversity (IntDiv) quantifies the dissimilarity between generated molecules within the same batch, indicating the diversity of the generated set. Novelty assesses the proportion of generated molecules that do not exist in the training dataset, indicating the model's ability to produce novel molecules. In addition to these metrics, physicochemical property value distribution plots were used to compare the properties of the de novo molecules with those of real molecules. These plots provide insights into the distribution and range of various physicochemical properties, such as molecular weight, logP, and other relevant descriptors. Comparing the distributions can help identify any differences or similarities between the generated and real molecules in terms of their properties.

Furthermore, UMAP (Uniform Manifold Approximation and Projection) embeddings were employed to visualize and compare the de novo and real molecules in a two-dimensional space. UMAP is a dimensionality reduction technique that preserves local neighborhood relationships, allowing for the visualization of high-dimensional data in a lower-dimensional space. By plotting the UMAP embeddings of the molecules, it is possible to observe patterns, clusters, or separations between the de novo and real

molecules, providing insights into their structural similarities or differences. These assessment methods collectively provide a comprehensive evaluation of DrugGEN's performance in generating de novo molecules, allowing for comparisons with real molecules in terms of both quantitative metrics and visualizations.

4.1. Performance

In this analysis, DrugGEN is compared to other generative methods previously reported in the literature using a range of benchmarking metrics. To conduct this evaluation, we generated approximately 10,000 novel molecules using each of the trained DrugGEN models, resulting in a total of 50,000 molecules. These generated molecules evaluated on MOSES benchmarking, as described in Polykovskiy et al. (96).

In the findings, generative performance of DrugGEN is presented alongside other models using widely adopted metrics, including validity, uniqueness, novelty, and internal diversity. However, it is important to note that these metrics provide only preliminary insights into the capabilities of a generative model and do not offer a comprehensive evaluation. While achieving high scores is considered a positive outcome, it is crucial to understand that being the top performer does not hold significant value on its own. This is because the objectives of different generative models can vary substantially. For instance, some models may focus on designing valid molecules, optimizing specific physicochemical properties, or generating molecules targeting specific biological targets. Therefore, a comprehensive assessment of a generative model should consider its specific objectives and applications beyond the basic benchmarking metrics.

Table 4.1 presents the performance of DrugGEN in comparison to a baseline model (ORGAN) and other recent methods, namely MolGPT, MGM, RELATION, REINVENT, MARS, BIMODAL, molDQN, and QADD. The selection of these methods was based on their algorithms and datasets, specifically utilizing ChEMBL for a fair comparison. DrugGEN demonstrates notably high performance across all metrics.

Unlike DrugGEN-Prot, the remaining DrugGEN models do not incorporate protein features. Instead, the transformer decoder input consists of either real AKT inhibitors or ChEMBL molecules. This simplification reduces overall complexity and facilitates the learning process. However, DrugGEN-Prot exhibits the highest uniqueness score among all DrugGEN models, comparable to the best-performing methods in this analysis.

Table 4.1. Performance comparison of default DrugGEN model against chosen molecule generative models.

Data type	Model name	Validity (↑)	Novelty (↑)	Uniq. (↑)	IntDiv (↑)	QED (↑)
Text	REINVENT	0.940	0.307	-	0.755	0.525
	BIMODAL	0.997	0.314	-	0.720	0.541
	RELATION	0.854	1.000	1.000	0.773	-
	MolGPT	0.994	0.797	1.000	0.857	-
	ORGAN	0.379	0.687	0.841	-	0.520
Graph	QADD	1.000	0.341	-	0.613	0.785
	MARS	0.997	0.333	-	0.641	0.746
	MGM	0.849	0.722	1.000	-	0.582
	molDQN	1.000	0.360	-	0.531	0.761
	DrugGEN	1.000	1.000	1.000	0.871	0.528

Unlike ORGAN, MolGPT, and MGM, which suffer from low novelty or uniqueness, DrugGEN leverages graph transformers, a novel architecture within its GAN generators. This approach contributes to higher novelty and validity scores. MolGPT and MGM also

employ transformer architectures, but their usage in generative modeling may result in lower novelty scores due to potential overfitting to training data. DrugGEN likely mitigates overfitting issues by utilizing probabilistic discrimination instead of cross-entropy loss. This hypothesis is an open-ended question about the generation of loss which should be further pursued, however it is not included in the scope of this thesis. This distinction is crucial, as the IntDiv metric reveals the diversity of structures among generated samples. DrugGEN achieves high IntDiv scores, indicating its ability to learn different molecular structures from the training dataset and generate diverse structural distributions. On the other hand, models like QADD, molDQN, BIMODAL, REINVENT, and MARS exhibit higher validity rates than DrugGEN. However, they suffer from low novelty scores, likely attributed to overfitting. These models generate a significant number of samples already present in their training sets.

4.2. Ablation Results

In this analysis, we conducted a comparison among various DrugGEN models, as described in section 3.3. The comparison involved evaluating the outputs of these models along with the molecules in the training datasets, using a set of established benchmarking metrics, including QED, SA, and logP metrics (96). The results presented in Table 2 indicate that all DrugGEN models exhibit high validity and uniqueness values, although there are variations in their novelty scores. Notably, both the DrugGEN-Prot and DrugGEN-CrossLoss models achieved a perfect novelty score of 1.000, indicating that all generated molecules differ from those in the training dataset. Regarding the internal diversity (IntDiv) of the generated molecules, all models demonstrated similar behavior and achieved significantly high values, comparable to the internal diversity observed in the entire ChEMBL dataset. While the actual AKT1 inhibitors displayed slightly lower internal diversity, the targeted models still managed to match the diversity observed in the larger ChEMBL training dataset. Furthermore, the DrugGEN-Prot model attained the best (lowest) FCD score of 15.581, which measures the proximity between the

distribution of physicochemical characteristics of the generated molecules and the distribution of the training dataset (98).

By comparing the FCD score, we observed that DrugGEN-Prot and other targeted models outperformed the baseline DrugGEN-NoTarget model. This improvement can be attributed to the utilization of target features within the generator network, instead of directly employing the features of real inhibitors. It was found that incorporating target features enhanced the learning process of the physicochemical properties specific to the selected target's real inhibitors. The DrugGEN-NoTarget model was excluded from this comparison as its FCD score was evaluated against ChEMBL molecules in its training dataset, rather than specifically focusing on AKT1 inhibitors like the other models.

Table 4.2. Ablation study results and models' comparison against dataset.

Models / datasets	Validity (↑)	Uniq. (↑)	Novelty (↑)	IntDiv (↑)	FCD (↓)	QED (↑)	logP	SA (↓)
ChEMBL Data	1.000	0.999	-	0.877	-	0.543	3.442	3.002
AKT1 inhibitors	1.000	0.750	-	0.827	-	0.460	4.071	3.051
DrugGEN-Prot	1.000	1.000	1.000	0.878	15.581	0.528	3.861	3.674
DrugGEN-CrossLoss	1.000	1.000	1.000	0.877	20.440	0.543	4.511	3.281
DrugGEN-Ligand	1.000	1.000	0.981	0.881	25.123	0.506	5.546	3.281
DrugGEN-RL	0.992	1.000	0.902	0.881	18.573	0.531	4.579	3.051

DrugGEN-NoTarget	1.000	1.000	0.990	0.883	10.449	0.572	3.761	3.302

The variations in physicochemical property-related metrics are evident from the QED, logP, and SA values presented in Table 2. The QED values of the DrugGEN models differ between the ChEMBL dataset and the AKT1 dataset, which was expected since all models, except DrugGEN-NoTarget, utilize both datasets during the learning process. Higher QED values indicate a positive outcome, suggesting that the de novo-generated compounds align well with the typical requirements of drug development. Therefore, from this perspective, all models can be considered successful as they enhance the QED value of the real AKT1 inhibitors dataset. On the other hand, when it comes to logP, there is no universally correct value since the optimal range varies depending on the specific ADME-related requirements of the drug under development.

4.3. Physicochemical Comparison with AKT1

Density plots are visualized for the examination of distributions of physicochemical properties in the molecules compared to AKT1 and ChEMBL molecules. It can be observed that the property distributions of non-targeted de novo molecules (DrugGEN-NoTarget) are similar to the ChEMBL molecules, which represent the training dataset of this model. On the other hand, the distributions of targeted de novo molecules (the other DrugGEN models) resemble those of real AKT1 inhibitors. In some of the plots, there might be a slight mean shift, where the real molecules have higher property values, or a slight difference in the distributions in terms of a right tail. The de novo molecules generated by these two DrugGEN models and the AKT1 inhibitors occupy a similar region in the physicochemical property space.

Hence, LogP values can be evaluated based on their similarity to the training datasets. In this context, DrugGEN-Prot demonstrated better scores, indicating a closer

match between the LogP values of its generated molecules and those of both the ChEMBL dataset and the real AKT inhibitors dataset. The synthetic accessibility (SA) score measures the ease of synthesis, where lower values are preferable. In this regard, all models produced comparable results except for DrugGEN-Prot, which had a slightly higher (worse) SA score compared to both the other models and the training dataset. The design process of DrugGEN-Prot involves modifying de novo molecules with respect to the selected target protein by incorporating protein features into the generator network. While the generator network does not directly consider the properties of the actual inhibitors of the target protein, there is an indirect influence due to the incorporation of those real inhibitors into the discriminator network. Consequently, this indirect influence may lead to de novo designs that differ from the real inhibitors of the selected target in terms of synthetic accessibility.

Target-specific models can be compared to the DrugGEN-NoTarget model and the utilized datasets in terms of physicochemical distribution to assess the models' ability to generate molecules specific to validated AKT1 inhibitors. For instance, the datasets indicate that AKT1 inhibitors have a slightly lower average QED score compared to the ChEMBL dataset. Therefore, it is expected that target-specific models would have an overall average QED score closer to 0.460. However, since the generator was trained on ChEMBL molecules, the ChEMBL dataset inevitably influences the generated molecules as a foundational structure. Figure 4.1 provides a visual comparison of the models.

DrugGEN-RL and DrugGEN-Ligand share the overall generation process, with the only difference being that DrugGEN-RL incorporates an additional loss value discouraging the use of existing AKT1 scaffolds. Regarding QED, the DrugGEN-RL model exhibits a more pronounced peak near the 0.500 value compared to DrugGEN-Ligand. This indicates that the DrugGEN-RL model focuses its generation process on specific molecular types more than DrugGEN-Ligand, which displays a broader distribution, suggesting the utilization of diverse molecular structures. Both DrugGEN-RL and DrugGEN-Ligand exhibit peaks near

the AKT1 inhibitor set, indicating that the generation process has shifted the distribution of the ChEMBL molecules to be more aligned with the AKT1 distribution. This distinction is evident in Figure 4.1A, as DrugGEN-NoTarget's distribution matches the ChEMBL molecular set. On the other hand, the DrugGEN-CrossLoss model exhibits a distribution that aligns with the ChEMBL molecules rather than AKT1 inhibitors on QED. DrugGEN-CrossLoss employs a single GAN system to shift the distribution of the ChEMBL molecules towards AKT1 inhibitors. However, in this case, the average QED score is nearly the same as that of the ChEMBL molecules. Surprisingly, the DrugGEN model displays two peaks in its distribution: one near the peak of AKT1 inhibitors and the other near the peak of the ChEMBL molecule set.

In contrast, DrugGEN-Prot does not rely on validated molecules during the generation process but rather focuses on protein features. In this scenario, DrugGEN-Prot successfully replicates the distribution of both datasets. Although the molecular information of the ChEMBL molecules is implicitly available to DrugGEN-Prot as input, the generator network does not have direct exposure to AKT1 inhibitors. This result indicates that DrugGEN-Prot was able to comprehend the overall drug likeness of the AKT1 inhibitor candidates, despite not having direct access to them during the generation process.

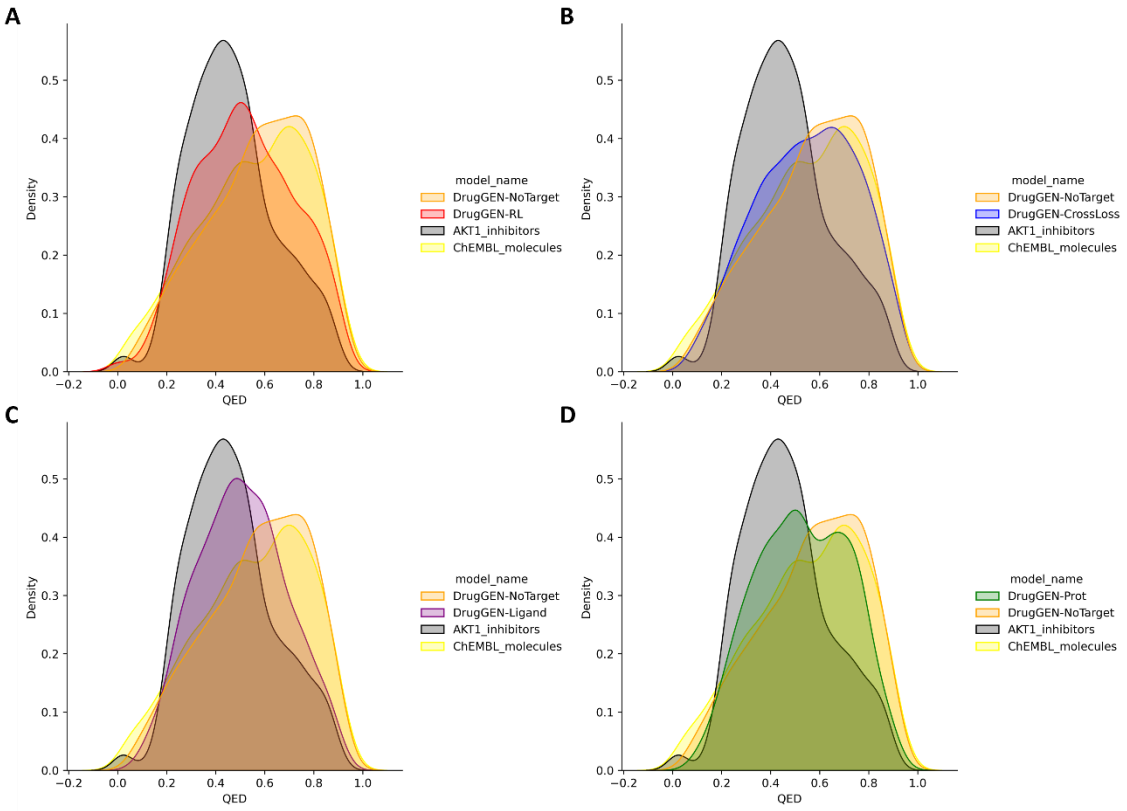

Figure 4.1. A comparative analysis of the target specific DrugGEN models is performed, evaluating their distribution against both datasets and a non-specific model, with a focus on the QED metric.

The logP metric analysis offers valuable insights into the performance of different models in generating compounds that align with the distribution of ChEMBL molecules. Notably, DrugGEN-CrossLoss, DrugGEN-Prot, and DrugGEN-RL demonstrate a connection to the distribution of ChEMBL molecules, indicating their ability to generate compounds that align with the logP characteristics of the dataset. This suggests that these models are effective in capturing the desired properties associated with the ChEMBL molecules. DrugGEN-Ligand displays a broader distribution with a higher maximum value which indicates the potential for generating compounds with higher lipophilicity.

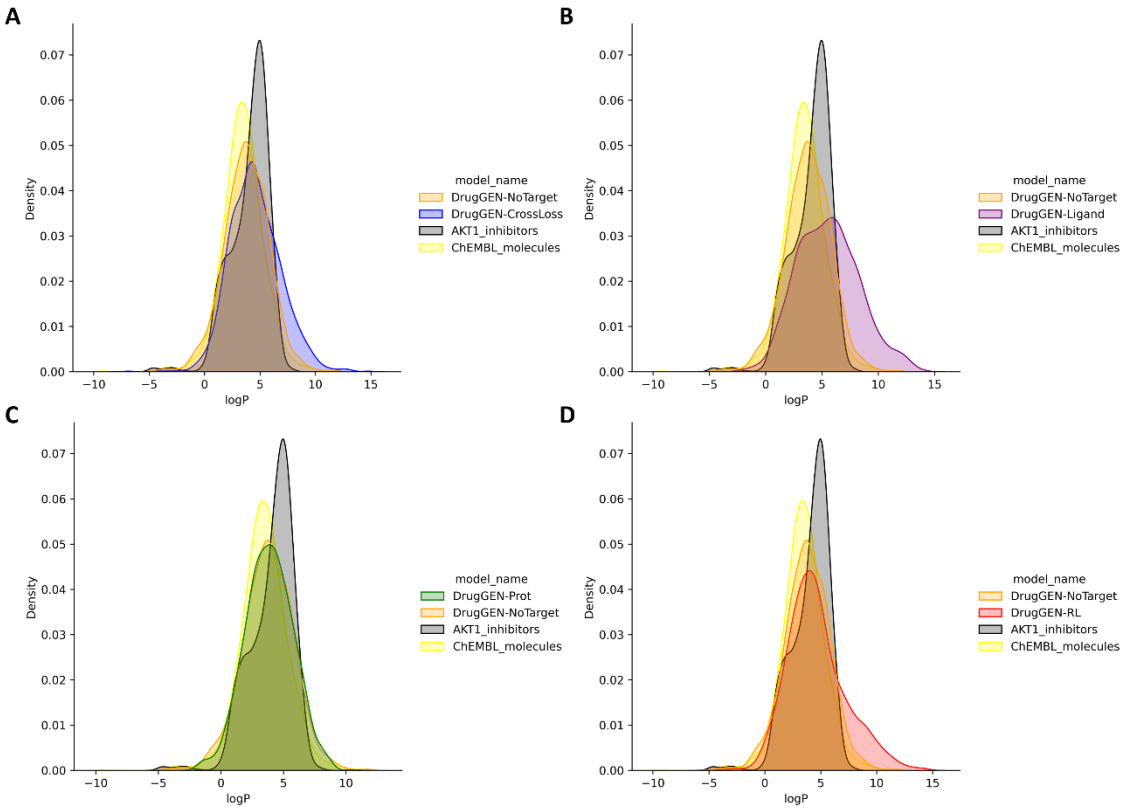

Figure 4.2. A comparative assessment is conducted to evaluate the distribution of the target specific DrugGEN models in comparison to both datasets and a non-specific model, with a specific emphasis on the logP metric.

The SA metric provides valuable insights into the performance of different models in generating compounds that are easy or hard to synthesize. Among the models evaluated, DrugGEN-CrossLoss, DrugGEN-RL, and DrugGEN-Ligand demonstrate a clear connection to the AKT1 distribution, while DrugGEN-Prot exhibits higher SA values, indicating a comparatively worse performance. This performance decline for DrugGEN-Prot model can be associated with the usage of the protein features. This model tries to modify the novel molecules based on the protein's features matrices. The process itself is computationally complex than other models and molecule generation based on the protein features might further drive the process to generate synthetically worse molecules in order to match the given protein features.

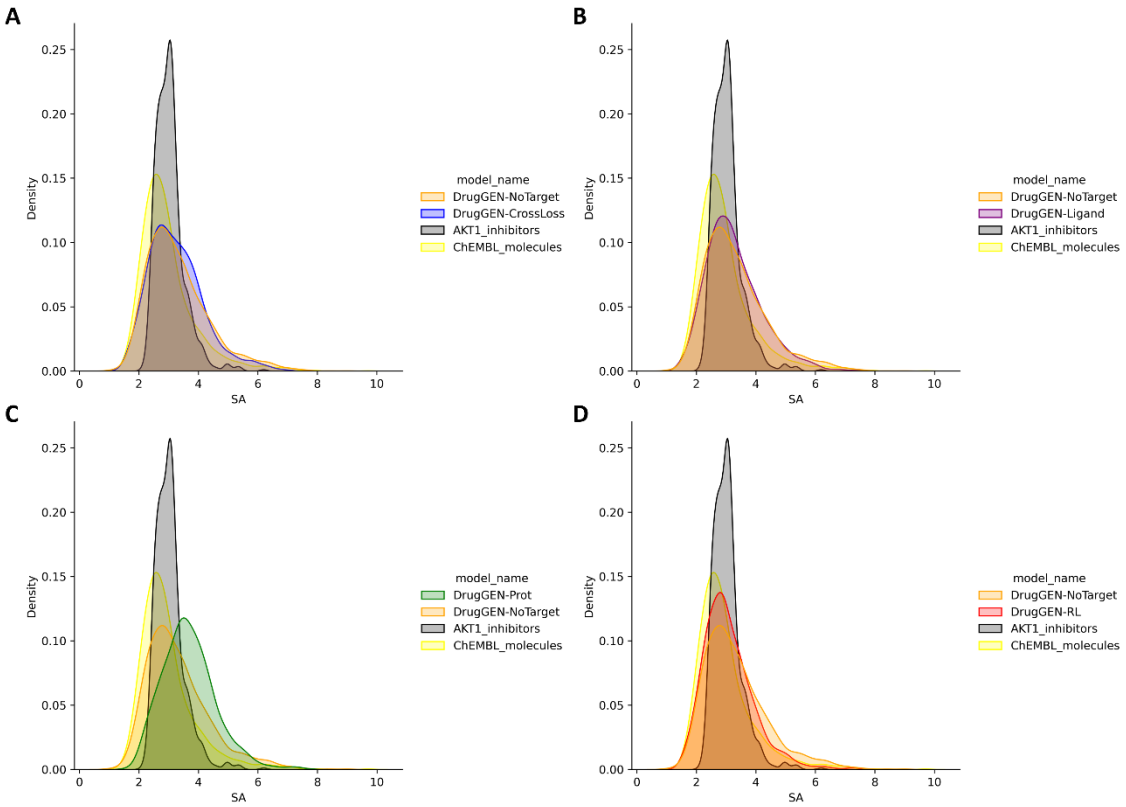

Figure 4.3. The target specific DrugGEN models are subjected to a distribution comparison against both datasets and a non-specific model, specifically analyzing their performance in terms of the SA metric.

4.4. Exploration of the Generated Data with Dimensionality Reduction

Data reduction analysis was conducted to further analyze the molecular embeddings. We randomly selected 1,000 real AKT1 inhibitors and 1,000 de novo molecules from DrugGEN-Prot, DrugGEN-CrossLoss, and DrugGEN-NoTarget models. The resulting UMAP visualization in 2D is depicted in Figure X, using UMAP parameters of n_neighbors=50, min_dist=0.8, and metric="jaccard". Each dot in the figures represents a molecule, and the colors indicate their respective sources.

It is observed that the models have learned the approximate structural distribution of ChEMBL molecules, with distinct clusters of molecules corresponding to different model variations. The proximity of the de novo molecules to the ChEMBL

molecules suggests their similarity in structural characteristics. Notably, the UMAP visualization using MACCS fingerprints enables differentiation between molecules based on the model that generated them, emphasizing the diverse molecule generation capability of the DrugGEN system.

In these plots, every data point represents a molecule, with colors displaying their source: either de novo molecules generated by the DrugGEN models or molecules from a real dataset. The Euclidean distances between the dots reflect the structural similarities based on Tanimoto similarity, calculated using descriptor-based molecular fingerprints. For representation, both the de novo generated, and training molecules are depicted using MACCS (Molecular ACCess System) descriptors. These descriptors consist of a 166-dimensional set of binary fingerprints, where each dimension represents the presence or absence of specific predefined structural features, such as structural patterns or functional groups (112).

UMAP plot specifically focuses on the de novo molecules generated by the DrugGEN-Prot model, revealing significant overlap with AKT1 molecules. This indicates that the DrugGEN-Prot model possesses a higher capacity to generate molecules resembling AKT1. Conversely, the DrugGEN-NoTarget model exhibits minimal overlap with AKT1 molecules. The UMAP plot provides valuable insights into the relationship between the de novo molecules generated by DrugGEN-Prot and the AKT1 inhibitors, highlighting potential structural resemblances and supporting the effectiveness of the generative model in producing molecules with similar properties.

In Figure 4.4A, we observe distinct clusters representing molecules generated from DrugGEN-NoTarget, DrugGEN-CrossLoss, and DrugGEN-Prot, which are located on different sides of the plane. In contrast, ChEMBL molecules appear in the center of the plane, with some instances positioned at the outer shell of the DrugGEN-NoTarget molecules. The presence of separate clusters indicates structural differences among the generated molecules. These differences arise because the embeddings utilized in this analysis are based on MACCS fingerprints, which derive from the structural backbones

and generate unique fingerprints for each molecule. In Figure 4.4B, a more complex distribution of molecules is observed. There are still distinct clusters representing DrugGEN-NoTarget and DrugGEN-CrossLoss. However, DrugGEN-Prot and the AKT1 inhibitor dataset appear to overlap on the plane, indicating that the molecules generated by DrugGEN-Prot resemble AKT1 inhibitors more closely compared to DrugGEN-CrossLoss and DrugGEN-NoTarget, while ensuring novelty in all generated molecules. The separation of molecules generated by DrugGEN-CrossLoss from the AKT1 cluster suggests that DrugGEN-Prot exhibits superior target-specific generation capabilities. This outcome was expected for DrugGEN-NoTarget, as it lacks internal mechanisms to generate molecules similar to AKT1.

When DrugGEN-RL and DrugGEN-Ligand are embedded with DrugGEN-NoTarget and ChEMBL molecules, a spherical overall localization pattern becomes apparent. The generated molecules are concentrated in the center of the visualization, while the ChEMBL molecules form a clustered shell-like structure around them. Comparing the generated molecules with AKT1 inhibitors, both DrugGEN-Ligand and DrugGEN-RL exhibit a mixture of generated molecules and AKT1 inhibitors on the plane. This observation suggests a structural similarity between the generated molecules and AKT1 inhibitors. On the other hand, the DrugGEN-NoTarget molecules form a separate cluster, distinct from both the target-specific models and the AKT1 dataset. This finding highlights the distinct structural differences between the target-specific molecules and the molecules generated by DrugGEN-NoTarget.

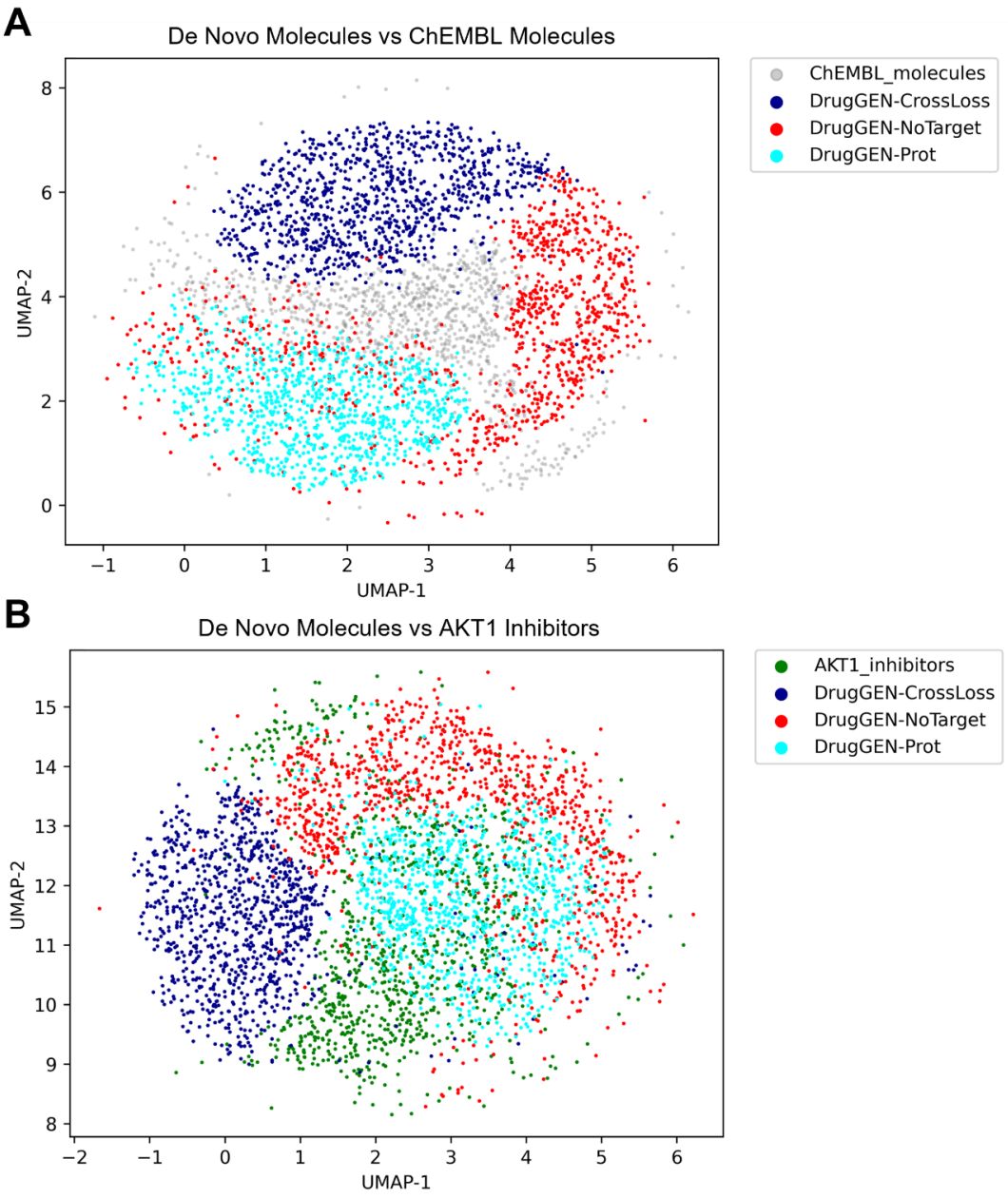

Figure 4.4. UMAP embeddings of the DrugGEN-NoTarget, DrugGEN-CrossLoss, and DrugGEN-Prot models against ChEMBL molecules and AKT1 inhibitors on separate planes.

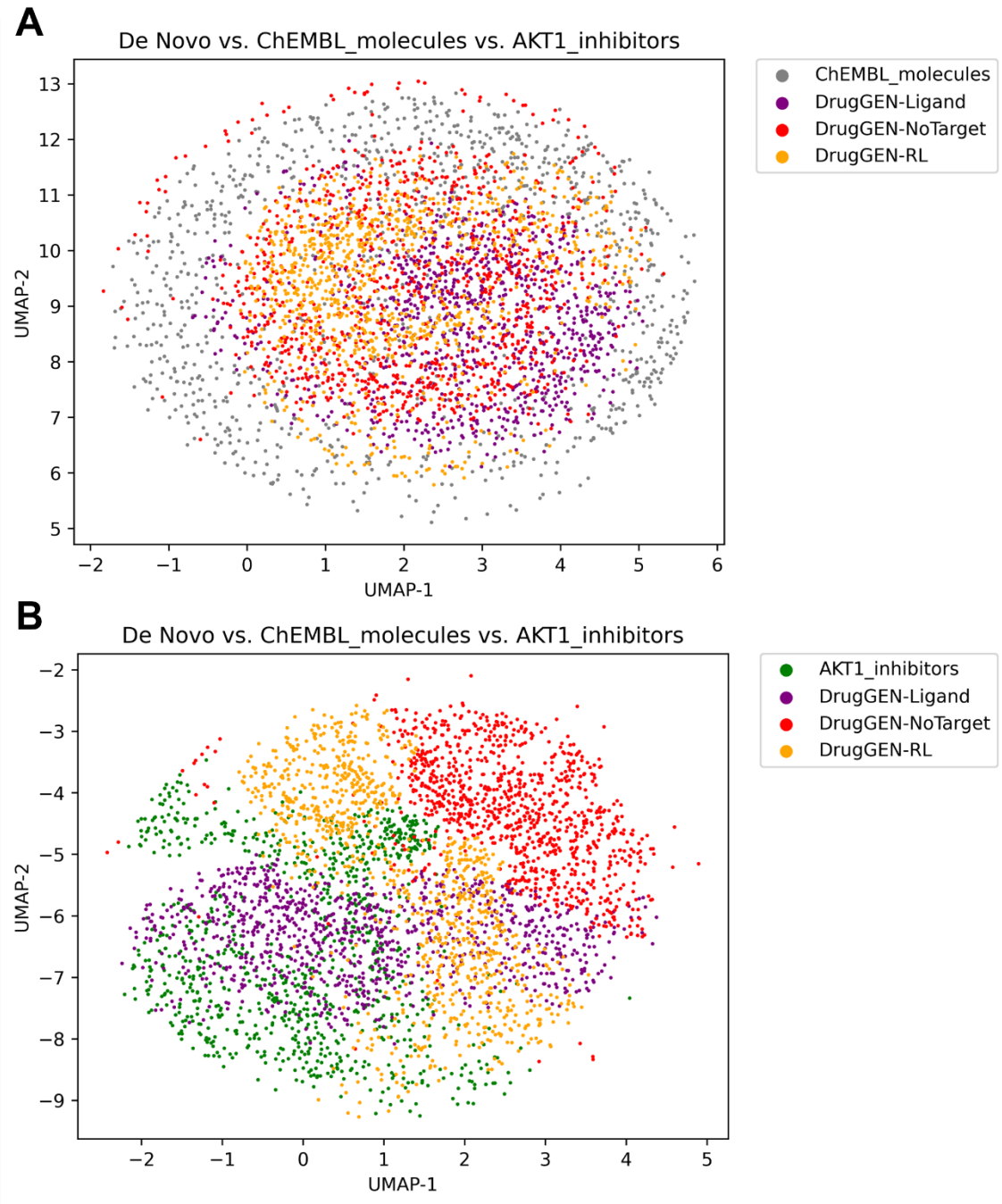

Figure 4.5. UMAP embeddings of the DrugGEN-NoTarget, DrugGEN-Ligand, and DrugGEN-RL models against ChEMBL molecules and AKT1 inhibitors on separate planes.

4.5. Failed Model Designs

The noise input approach proved ineffective as the sparse nature of the graph data hindered the learning process. A limited number of valid molecules were obtained, while the system also generated atoms that were not properly connected to each other, resulting in the formation of non- connected molecular structures. The adjacency matrix, which represents the graph structure, is two-dimensional and contains a significant number of zero values, with nearly 90% of the matrix being filled with zeros. As the matrix size increases, the sparsity becomes exponentially more pronounced. This abundance of null data within the matrix poses challenges for neural networks to learn effectively (116, 117). This lack of effectiveness becomes apparent during the training process as well. Despite the divergence in loss values between the generator and discriminator networks, there is a notable absence of improvement in performance metrics as training progresses. Frequently, the validity of the system approaches zero. Furthermore, it is crucial to acknowledge that relying solely on performance metrics for assessing model efficacy is not consistently fruitful. Some unsuccessful models exhibit high validity scores, yet the molecules they generate lack connectivity, producing disjointed carbon atoms. Sparse data in graph generation remains a persistent problem that has not been fully resolved to date. To overcome this issue, a modification was made to the input of GAN1 by changing it from Gaussian noise to real molecule input. This adjustment facilitated the learning process of GAN1, enabling successful molecule generation. By utilizing real molecule input instead of noise, the model's ability to learn and generate molecules was improved. GNN discriminator proved to be unstable, resulting in a lack of effective competition between the generator and discriminator within the GAN network. The instability observed in GAN models led to the decision of changing the discriminator to a multilayer perceptron (MLP) instead. By employing an MLP discriminator, the generative network became more stable, allowing for more effective competition between the generator and discriminator components of the GAN. This modification helped improve the overall performance and training stability of the DrugGEN model. Using classical

Transformer approach came with some problems. Transformer network was designed to process text-data (94). Using graph data with the vanilla transformer attention did not work as intended and created mostly unconnected graphs through training. After implementing graph transformer architecture to DrugGEN system, attention module was able to attend both annotation and adjacency matrices creating a message-passing like process inside attention module. This allowed DrugGEN to process and modify molecules better than vanilla transformer approach.

5. DISCUSSION

The assessment of DrugGEN's performance in designing de novo molecules involved the use of various benchmarking metrics. Validity, uniqueness, internal diversity (IntDiv), and novelty were employed to evaluate the quality and diversity of the generated molecules. Physicochemical property value distribution plots were utilized to compare the properties of the de novo molecules with those of real molecules. UMAP embeddings were employed to visualize and compare the structural similarities or differences between the de novo and real molecules. The performance evaluation showed that DrugGEN exhibited high scores in terms of validity, uniqueness, novelty, and internal diversity. The physicochemical comparison demonstrated that the distributions of targeted de novo molecules resembled those of real inhibitors, while non-targeted de novo molecules were similar to the training dataset. The UMAP analysis revealed the approximate structural distribution of the de novo molecules, with DrugGEN-Prot showing significant overlap with AKT1 inhibitors. These findings highlight DrugGEN's generative capabilities and its ability to produce molecules with desired properties.

There are two commonly used benchmarking strategies for molecule generative models, namely the MOSES and Guacamol benchmarks (96, 113). These benchmarks incorporate various metrics such as Validity, Uniqueness, Novelty, FCD, similarity, logP, and others. These metrics evaluate the generative capabilities of models and the physicochemical properties of the generated molecules. While benchmarking metrics provide initial insights, it is crucial to consider the specific objectives and applications of a generative model beyond these metrics. Different models may prioritize different aspects, such as validity, physicochemical optimization, or targeting specific biological activities.

In the case of the DrugGEN model, its focus is primarily on the chemical perspective of the generative process. DrugGEN performs exceptionally well in generating structurally diverse molecules while ensuring novelty. Additionally, the model

successfully reproduces the physicochemical properties of AKT1 inhibitors based on metrics such as QED, logP, and SA. However, relying solely on quantitative metrics is insufficient to determine the potential of a generated molecule to become a drug candidate. Goal-oriented models evaluate their generated molecules using these metrics, but it's important to note that these metrics are approximations and do not correspond to experimentally validated results (99). Furthermore, these metrics are inadequate for assessing a molecule's affinity for any specific target. Currently, there are no metrics available for benchmarking across target-specific generative models. Indeed, certain target-specific models utilize docking experiment results as a means of comparative analysis, although it's important to note that not all models use the same target (114). Traditional docking methods typically require expert knowledge, where specialists study the target and design experiments specifically tailored to that target. In contrast, modern deep learning-based docking techniques can be executed in a blind manner without prior knowledge of the target (115). To enable fair comparisons and accurate measurement of the capacity of generative models, there is a need for universal metrics and quantitative calculations. These metrics should provide a standardized framework for evaluating the performance of different generative models. The development of such metrics would help establish fair and consistent benchmarks across various generative models, allowing for meaningful comparisons and assessments. By incorporating universal metrics and quantitative calculations, the field of generative modeling can advance towards more objective and reliable evaluations.

The DrugGEN model excels in generating novel molecules compared to other methods. Many of the compared models generate molecules that already exist in their training datasets, which is not desirable. In contrast, the MARS model exhibits low novelty, likely due to its generation process that relies on fragment modification without allowing for atom-level changes (105). One of the key advantages of the DrugGEN model is its implementation at the atom resolution for both molecules and proteins. This atom-level resolution enables modifications to be made at the individual atom and bond level.

As a result, the model has a higher probability of creating novel molecules since the modifications are performed at such a detailed level. This atom-level flexibility enhances the chances of generating molecules that are structurally unique and different from those present in the training dataset.

The compound datasets used to train the DrugGEN models were derived from ChEMBL and DrugBank. The ChEMBL dataset initially consisted of approximately 1.6 million molecules. However, for training purposes, the dataset was truncated to include only molecules with a maximum of 45 heavy atoms, resulting in a remaining dataset of approximately 1.4 million molecules. While this dataset size may not be considered large for training a complex system, it was found to be sufficient for effectively training the DrugGEN models using these molecules. In contrast, the AKT inhibitor dataset, derived from DrugGEN, was curated to include inhibitors of AKT1, AKT2, and AKT3. This was done to increase the dataset size. The training dataset specifically contained only 2,754 molecules that were AKT-specific inhibitors. The limited size of this dataset posed a challenge for the discriminator in the GAN system. GANs rely on extensive training data to effectively learn and avoid overfitting (118). When the discriminator network was implemented using GNN, the training process was unstable, and the generator system struggled to learn effectively. To address this issue, a simpler MLP network was employed as the discriminator. This change significantly improved the stability of the GAN system and facilitated a more robust and stable learning process. The use of an MLP discriminator mitigated the potential overfitting caused by the limited AKT inhibitor dataset, enabling the DrugGEN model to train more effectively. Curating a dataset that has more inhibitors might come with an additional weight on the discriminator to suppress overfitting issues.

The generator network in the DrugGEN model is built upon the graph transformer architecture (90). This architecture enables the simultaneous processing of both annotation and adjacency matrices within an attention module. This allows for the

representation of bond information between atoms in the network. Various approaches exist for generating attention based on the adjacency and annotation matrices. One particular study utilizes a node-edge interaction module, where attention is computed through matrix multiplication between node and edge matrices (119). However, in this study, only the node features are updated, and there is no iterative optimization of the edge features, unlike in the DrugGEN model. In the DrugGEN model, both the node and edge features are updated through the learning process. Not only are the node features updated based on the created attention weights, but the attention weights themselves are further modified to generate an updated adjacency matrix for the next layer. This iterative updating of both the node and edge components ensures that both aspects of the molecular graph are dynamically adjusted during the learning process, leading to enhanced representation and generation capabilities.

Training GAN systems, including stacked systems like DrugGEN, can be challenging due to stability issues and the potential for mid-training collapse. The traditional training schema of GANs involves a min-max game between the generator and discriminator, but vanilla GANs are susceptible to inherent problems associated with their architecture (120, 121).

One of the challenges in GAN training is achieving convergence. Since GANs are gradient-based systems, each model updates independently, making it difficult to reach convergence. There is a dilemma in the learning process where a poorly trained discriminator fails to provide meaningful feedback to the generator, hindering its own training. On the other hand, if the discriminator learns too well, the gradients can become extremely small, resulting in vanishing gradients that impact the overall model (122).

Another issue that can arise in GAN systems is mode collapse, where the generator repeatedly generates the same or similar data that can deceive the discriminator network (123). To address some of these problems, the Wasserstein GAN

(WGAN) model was proposed. WGAN uses the Earth-Mover distance (Wasserstein-1) to measure the cost of transforming one probability distribution into another during training. WGAN has been reported to train more stably and mitigate mode collapse issues (124).

In addition to WGAN, another technique called gradient penalty (GP) was proposed to enhance the stability of GAN training. GP enforces the Lipschitz constraint by directly constraining the gradient norm of the critic's output with respect to its input. This soft constraint and penalty on the gradient norm for random samples help ensure stable training and address issues related to vanishing and exploding gradients (125). By incorporating these advancements, the DrugGEN model benefits from improved training stability and addresses some of the inherent challenges associated with GAN training.

The DrugGEN model employs the WGAN-GP training system to enhance stability and prevent collapse during GAN training. This training approach allows the stacked system to be trained in a more stable manner. By aggregating the losses of both GAN1 and GAN2, the learning process becomes more end-to-end, enabling the DrugGEN model to allocate different tasks to each GAN while maintaining a shared learning system. As a result, the DrugGEN model is capable of generating novel molecules and target-specific molecules through this combined training approach.

The DrugGEN model has demonstrated superior performance compared to other models in the comparison. Despite its computational complexity, the DrugGEN system outperformed other models in terms of generating highly diverse and novel molecules. A direct comparison can be made between the ORGAN model and DrugGEN-NoTarget, as neither model does not consider target features and do not generate target-specific molecules. However, it is evident that DrugGEN-NoTarget excels in generating molecules that are both highly novel and diverse, while also maintaining a similar level of heavy atom count as the ORGAN model. The ORGAN model relies on SMILES representation and does not consider spatial connectivity features in representing

molecules. Additionally, ORGAN utilizes LSTM as the generator and CNN as the discriminator module. In contrast, the DrugGEN model employs a more recent architecture for the generator network, which potentially enhances its understanding of molecular structures. As observed in the metrics, DrugGEN performs significantly better than ORGAN, despite using the same training method and dataset. This direct comparison highlights DrugGEN's improved production efficiency using GANs.

When compared to MolGPT, which utilizes a transformer decoder architecture, DrugGEN performs similarly in most aspects except for novelty score. MolGPT has a limitation in that it sometimes generates molecules that already exist in its training dataset. During training, MolGPT uses real molecules and during the generation process, it employs scaffolds and molecular descriptors to generate new molecules. However, the paper does not mention a scaffold split, where scaffolds present in the training data are not used during inference. This lack of scaffold split may hinder the novelty of the molecules generated by MolGPT.

In contrast, DrugGEN employs a training data split that allows for the use of test sets that were not seen during training. This enables DrugGEN to generate novel molecules that are not present in the training set. Additionally, the DrugGEN system utilizes a dual system to guide the molecular generation process specifically towards molecules that would interact with AKT1. This introduces an additional layer of complexity and manipulation of molecules to modify their structures and physicochemical properties. Overall, while DrugGEN and MolGPT share similarities in their transformer-based architectures, DrugGEN's training data split and target-specific generation mechanism give it an advantage in terms of generating novel molecules that interact with AKT1.

When comparing DrugGEN to RL-based models such as QADD, molDQN, BIMODAL, MARS, and REINVENT, it is important to focus on their generative efficiency and novelty. These RL models aim to generate molecules with specific desired properties

or optimize certain physicochemical properties. In terms of generative efficiency, DrugGEN does not employ any additional mechanisms to specifically optimize molecules based on desired properties. Therefore, comparing DrugGEN to these RL models based on physicochemical properties may not be meaningful. Instead, it would be more appropriate to evaluate their generative efficiency in terms of novelty and diversity.

In this regard, DrugGEN outperforms these RL models in terms of novelty since a significant portion of the molecules generated by these RL models already exist in their respective training sets. This indicates that these RL models struggle to generate truly novel molecular candidates. In contrast, DrugGEN excels at generating novel molecules, as it employs a training data split that allows for the generation of molecules not present in the training set. Therefore, when considering the generative efficiency and novelty of the models, DrugGEN demonstrates superiority over RL-based models such as QADD, molDQN, BIMODAL, MARS, and REINVENT.

The analysis of DrugGEN-Prot's performance in approximating AKT1 inhibitors without direct exposure to the molecules is certainly impressive. By incorporating target features within the generator network, DrugGEN-Prot is able to enhance the learning process of physicochemical properties specific to AKT1 inhibitors. The ability of DrugGEN-Prot to replicate the physicochemical characteristics required for an AKT1 inhibitor is a significant accomplishment. It demonstrates that target-specific molecule generation can be achieved by leveraging protein features and indirect guidance from molecular data.

However, it is worth noting that the complex process of incorporating protein features may slightly affect the synthetic accessibility of the generated molecules. This suggests that while DrugGEN-Prot excels at replicating the physicochemical needs of AKT1 inhibitors, there might be a trade-off in terms of synthetic feasibility. Overall, the performance of DrugGEN-Prot in replicating the physicochemical properties of AKT1 inhibitors through the incorporation of target features is a remarkable achievement,

highlighting the potential of utilizing protein features for target-specific molecule generation.

The UMAP embeddings further support the findings of the physicochemical analysis. The distribution of molecules in the UMAP embeddings reveals important insights into their structural similarities and relationships. In the case of DrugGEN-Prot, the fact that the generated molecules are located in the same planar area as the AKT1 inhibitors, despite not directly observing the inhibitors during training, suggests that DrugGEN-Prot was able to generate molecules that are both novel and exhibit similar physicochemical characteristics based on MACCS fingerprints. This indicates the successful incorporation of target features into the generation process. Similarly, DrugGEN-Ligand and DrugGEN-RL show a mixed distribution with AKT1 inhibitors, indicating that molecules generated from these models possess some structural similarities to AKT1 inhibitors. On the other hand, DrugGEN-CrossLoss appears to have a separated cluster, suggesting that the overall generation process for this model does not produce molecules that resemble AKT1 inhibitors based on MACCS fingerprints. Overall, the UMAP embeddings provide visual evidence that supports the physicochemical analysis findings. They demonstrate the ability of DrugGEN-Prot, DrugGEN-Ligand, and DrugGEN-RL to generate molecules that share certain characteristics with AKT1 inhibitors, while highlighting the distinct distribution of molecules generated by DrugGEN-CrossLoss.

To sum up, this study provides different options and capabilities within the DrugGEN system, allowing users to tailor their molecule generation approach based on their specific needs and available data. The DrugGEN-Ligand, DrugGEN-CrossLoss, and DrugGEN-RL models are suitable when protein data is not available or not necessary for the generation process. It relies solely on the validated inhibitors of the selected molecules to generate novel molecules. The DrugGEN-CrossLoss model can be utilized when there are limitations in computational resources. It employs a single GAN instead

of the stacked GANs used in other models, providing a more resource-efficient option. The DrugGEN-RL model is designed for generating molecules with different scaffolds, enabling the design of molecules with diverse core structures. DrugGEN-Prot is the default and most successful model among the variations. It incorporates target-specific features by incorporating protein data, allowing the generation of novel and diverse molecules that exhibit structural and physicochemical similarities to existing inhibitors.

Each model has its own strengths and can be chosen based on the specific requirements and objectives of the molecule generation task. The DrugGEN system provides a flexible and adaptable approach for generating molecules, offering various options to suit different scenarios.

6. CONCLUSION

In this study, we introduced the DrugGEN system, which combines GANs and the graph transformer architecture to automatically design target-specific drug candidate molecules. The primary objective of DrugGEN was to generate inhibitor candidates based on the selected target. The system encompasses multiple models aimed at exploring target-centric generation capabilities. The DrugGEN models demonstrate comparable or superior performance to state-of-the-art models in terms of performance metrics, indicating their high efficiency and capacity for molecule generation. We conducted analyses on physicochemical metrics such as QED, SA, and logP, demonstrating that DrugGEN models can generate de novo molecules with molecular characteristics shared with real inhibitors of the AKT1 protein. Additional computational analyses were performed to assess the target-specific properties of these de novo molecules, which revealed their AKT1 targeting potential.

Using graph representation carries the burden of computational complexity. High complexity leads to increased memory usage and slower training compared to text-based methods. Additionally, this complexity imposes a limit on the maximum number of atoms that can be processed within the DrugGEN model. While DrugGEN can handle molecules with heavy atom counts up to 45, processing larger molecules would exponentially increase the computational resource requirements.

The DrugGEN model has limitations regarding the overall generation process. These models rely either on protein features or existing inhibitor data. However, for targets that lack any protein or inhibitor data, there is no feasible way to generate drug candidates.

The DrugGEN model has been tested exclusively against a single target, AKT1. This particular target benefits from having available protein data and inhibitor data from open sources. However, modeling other targets with minimal inhibitor data can be challenging

since the discriminator network may become overfit to the limited data within a short timeframe. In such cases, molecular data can be constructed using inhibitors from the same protein family. However, this approach would restrict the specificity of the generated molecules.

As future aspects of this study, DrugGEN system can be re-implemented to process text-based data using either SMILES, or SELFIES (127). This way both molecular data and protein data can be handled using smaller representations which will a relief on computational complexity and memory usage. Also, generative model benchmarks are designed to measure text-based generation performance better than geometric ones which would be a better way for optimizing the generation system. Even though text-based method does not encode any spatial features, geometric features can be added as vector codes to molecules to give additional information to the system. This way molecule generation system can be optimized.

DrugGEN model only trained for a single target however, training this model for multiple targets is also possible. Using multiple targets and corresponding molecular inhibitors, systems can be trained to learn specifics of being a molecular inhibitor by looking at protein data. This might be proven useful when studying target that does not have any or a few known inhibitors. A system that can learn to design inhibitors based on target data and existing respective molecular inhibitors, can theoretically design molecular inhibitors for any given protein data.

7. REFERENCES

1. Wang M, Wang Z, Sun H, Wang J, Shen C, Weng G, et al. Deep learning approaches for de novo drug design: An overview. Current Opinion in Structural Biology [Internet]. 2022 [cited 2023 May 21];72:135–44. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0959440X21001433

2. Ou-Yang S sheng, Lu J yan, Kong X qian, Liang Z jie, Luo C, Jiang H. Computational drug discovery. Acta Pharmacol Sin [Internet]. 2012 [cited 2023 May 21];33(9):1131–40. Available from: https://www.nature.com/articles/aps2012109

3. Chen H, Engkvist O, Wang Y, Olivecrona M, Blaschke T. The rise of deep learning in drug discovery. Drug Discovery Today [Internet]. 2018 [cited 2023 May 21];23(6):1241–50. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1359644617303598

4. Lavecchia A. Deep learning in drug discovery: opportunities, challenges and future prospects. Drug Discovery Today [Internet]. 2019 [cited 2023 May 21];24(10):2017–32. Available from: https://linkinghub.elsevier.com/retrieve/pii/S135964461930282X

5. Zhang L, Tan J, Han D, Zhu H. From machine learning to deep learning: progress in machine intelligence for rational drug discovery. Drug Discovery Today [Internet]. 2017 [cited 2023 May 21];22(11):1680–5. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1359644616304366

6. Gawehn E, Hiss JA, Schneider G. Deep Learning in Drug Discovery. Mol Inf [Internet]. 2016 [cited 2023 May 21];35(1):3–14. Available from: https://onlinelibrary.wiley.com/doi/10.1002/minf.201501008

7. Meyers J, Fabian B, Brown N. De novo molecular design and generative models. Drug Discovery Today [Internet]. 2021 [cited 2023 May 21];26(11):2707–15. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1359644621002531

8. Tong X, Liu X, Tan X, Li X, Jiang J, Xiong Z, et al. Generative Models for De Novo Drug Design. J Med Chem [Internet]. 2021 Oct 14 [cited 2023 May 21];64(19):14011–27. Available from: https://pubs.acs.org/doi/10.1021/acs.jmedchem.1c00927

9. Cheng T, Li Q, Zhou Z, Wang Y, Bryant SH. Structure-Based Virtual Screening for Drug Discovery: a Problem-Centric Review. AAPS J [Internet]. 2012 [cited 2023 May 27];14(1):133–41. Available from: http://link.springer.com/10.1208/s12248-012-9322-0

10. Leelananda SP, Lindert S. Computational methods in drug discovery. Beilstein J Org Chem [Internet]. 2016 Dec 12 [cited 2023 May 27];12:2694–718. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5238551/

11. Sun M, Zhao S, Gilvary C, Elemento O, Zhou J, Wang F. Graph convolutional networks for computational drug development and discovery. Briefings in Bioinformatics [Internet]. 2020 May 21 [cited 2023 May 27];21(3):919–35. Available from: https://academic.oup.com/bib/article/21/3/919/5498046

12. Tiwari A, Singh S. Computational approaches in drug designing. In: Bioinformatics [Internet]. Elsevier; 2022 [cited 2023 May 27]. p. 207–17. Available from: https://linkinghub.elsevier.com/retrieve/pii/B9780323897754000109

13. Neil D, Segler M, Guasch L, Ahmed M, Plumbley D, Sellwood M, et al. Exploring Deep Recurrent Models with Reinforcement Learning for Molecule Design. 2022 Feb 10 [cited 2023 May 27]; Available from: https://openreview.net/forum?id=HkcTe-bR-&source=post_page-----a942a1c523a3----------

14. Gasteiger J. Chemoinformatics: a new field with a long tradition. Anal Bioanal Chem [Internet]. 2006 [cited 2023 May 27];384(1):57–64. Available from: http://link.springer.com/10.1007/s00216-005-0065-y

15. Vanhee P, Van Der Sloot AM, Verschueren E, Serrano L, Rousseau F, Schymkowitz J. Computational design of peptide ligands. Trends in Biotechnology [Internet]. 2011 [cited 2023 May 27];29(5):231–9. Available from: https://linkinghub.elsevier.com/retrieve/pii/S016777991100014X

16. Durrant JD, McCammon JA. Molecular dynamics simulations and drug discovery. BMC Biol [Internet]. 2011 [cited 2023 May 27];9(1):71. Available from: https://bmcbiol.biomedcentral.com/articles/10.1186/1741-7007-9-71

17. Sanchez-Lengeling B, Aspuru-Guzik A. Inverse molecular design using machine learning: Generative models for matter engineering. Science [Internet]. 2018 Jul 27 [cited 2023 May 27];361(6400):360–5. Available from: https://www.science.org/doi/10.1126/science.aat2663

18. Goh GB, Hodas NO, Vishnu A. Deep learning for computational chemistry. J Comput Chem [Internet]. 2017 Jun 15 [cited 2023 May 27];38(16):1291–307. Available from: https://onlinelibrary.wiley.com/doi/10.1002/jcc.24764

19. Venkatasubramanian V, Chan K, Caruthers JM. Computer-aided molecular design using genetic algorithms. Computers & Chemical Engineering [Internet]. 1994 [cited 2023 May 27];18(9):833–44. Available from: https://linkinghub.elsevier.com/retrieve/pii/0098135493E00233

20. Andricopulo AD, Montanari CA. Structure-Activity Relationships for the Design of Small-Molecule Inhibitors. Mini Reviews in Medicinal Chemistry. 2005 Jun 1;5(6):585–93.

21. Böhm HJ, Flohr A, Stahl M. Scaffold hopping. Drug Discovery Today: Technologies [Internet]. 2004 [cited 2023 May 27];1(3):217–24. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1740674904000460

22. Nicolaou CA, Watson IA, LeMasters M, Masquelin T, Wang J. Context Aware Data-Driven Retrosynthetic Analysis. J Chem Inf Model [Internet]. 2020 Jun 22 [cited 2023 May 27];60(6):2728–38. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.9b01141

23. Choudhury C, Arul Murugan N, Priyakumar UD. Structure-based drug repurposing: Traditional and advanced AI/ML-aided methods. Drug Discovery Today [Internet]. 2022 [cited 2023 May 27];27(7):1847–61. Available from: https://linkinghub.elsevier.com/retrieve/pii/S135964462200112X

24. Stahl M, Guba W, Kansy M. Integrating molecular design resources within modern drug discovery research: the Roche experience. Drug Discovery Today [Internet]. 2006 [cited 2023 May 27];11(7–8):326–33. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1359644606000092

25. Speck-Planche A. Recent advances in fragment-based computational drug design: tackling simultaneous targets/biological effects. Future Medicinal Chemistry [Internet]. 2018 Sep 1 [cited 2023 May 27];10(17):2021–4. Available from: https://www.future-science.com/doi/10.4155/fmc-2018-0213

26. Sridharan B, Goel M, Priyakumar UD. Modern machine learning for tackling inverse problems in chemistry: molecular design to realization. Chem Commun [Internet]. 2022 [cited 2023 May 27];58(35):5316–31. Available from: http://xlink.rsc.org/?DOI=D1CC07035E

27. Tarca AL, Carey VJ, Chen X wen, Romero R, Drăghici S. Machine Learning and Its Applications to Biology. Lewitter F, editor. PLoS Comput Biol [Internet]. 2007 Jun 29 [cited 2023 May 27];3(6):e116. Available from: https://dx.plos.org/10.1371/journal.pcbi.0030116

28. El Naqa I, Murphy MJ. What Is Machine Learning? In: El Naqa I, Li R, Murphy MJ, editors. Machine Learning in Radiation Oncology [Internet]. Cham: Springer International Publishing; 2015 [cited 2023 May 27]. p. 3–11. Available from: http://link.springer.com/10.1007/978-3-319-18305-3_1

29. Shinde PP, Shah S. A Review of Machine Learning and Deep Learning Applications. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) [Internet]. Pune, India: IEEE; 2018 [cited 2023 May 27]. p. 1–6. Available from: https://ieeexplore.ieee.org/document/8697857/

30. Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge, Massachusetts: The MIT Press; 2016. 775 p. (Adaptive computation and machine learning).

31. Li Z, Liu F, Yang W, Peng S, Zhou J. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Trans Neural Netw Learning Syst [Internet]. 2022 [cited 2023 May 27];33(12):6999–7019. Available from: https://ieeexplore.ieee.org/document/9451544/

32. Krizhevsky A. One weird trick for parallelizing convolutional neural networks [Internet]. arXiv; 2014 [cited 2023 May 27]. Available from: http://arxiv.org/abs/1404.5997

33. Sutskever I, Martens J, Hinton G. Generating text with recurrent neural networks. In: Proceedings of the 28th International Conference on International Conference on Machine Learning. Madison, WI, USA: Omnipress; 2011. p. 1017–24. (ICML'11).

34. Yu Y, Si X, Hu C, Zhang J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. Neural Computation [Internet]. 2019 [cited 2023 May 31];31(7):1235–70. Available from: https://direct.mit.edu/neco/article/31/7/1235-1270/8500

35. Mahmud M, Kaiser MS, Hussain A, Vassanelli S. Applications of Deep Learning and Reinforcement Learning to Biological Data. IEEE Trans Neural Netw Learning Syst [Internet]. 2018 [cited 2023 May 31];29(6):2063–79. Available from: https://ieeexplore.ieee.org/document/8277160/

36. Mater AC, Coote ML. Deep Learning in Chemistry. J Chem Inf Model [Internet]. 2019 Jun 24 [cited 2023 May 31];59(6):2545–59. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.9b00266

37. Bond-Taylor S, Leach A, Long Y, Willcocks CG. Deep Generative Modelling: A Comparative Review of VAEs, GANs, Normalizing Flows, Energy-Based and Autoregressive Models. IEEE Trans Pattern Anal Mach Intell [Internet]. 2022 Nov 1 [cited 2023 May 31];44(11):7327–47. Available from: https://ieeexplore.ieee.org/document/9555209/

38. Jørgensen PB, Schmidt MN, Winther O. Deep Generative Models for Molecular Science. Mol Inf [Internet]. 2018 [cited 2023 May 31];37(1–2):1700133. Available from: https://onlinelibrary.wiley.com/doi/10.1002/minf.201700133

39. Gonog L, Zhou Y. A Review: Generative Adversarial Networks. In: 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA) [Internet]. Xi'an, China: IEEE; 2019 [cited 2023 May 31]. p. 505–10. Available from: https://ieeexplore.ieee.org/document/8833686/

40. Creswell A, White T, Dumoulin V, Arulkumaran K, Sengupta B, Bharath AA. Generative Adversarial Networks: An Overview. IEEE Signal Process Mag [Internet]. 2018 [cited 2023 May 31];35(1):53–65. Available from: http://ieeexplore.ieee.org/document/8253599/

41. Aggarwal A, Mittal M, Battineni G. Generative adversarial network: An overview of theory and applications. International Journal of Information Management Data Insights [Internet]. 2021 [cited 2023 May 31];1(1):100004. Available from: https://linkinghub.elsevier.com/retrieve/pii/S2667096820300045

42. Lopez-Alvis J, Laloy E, Nguyen F, Hermans T. Deep generative models in inversion: The impact of the generator's nonlinearity and development of a new approach based on a variational autoencoder. Computers & Geosciences [Internet]. 2021 [cited 2023 May 31];152:104762. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0098300421000698

43. Girin L, Leglaive S, Bie X, Diard J, Hueber T, Alameda-Pineda X. Dynamical Variational Autoencoders: A Comprehensive Review. FNT in Machine Learning [Internet]. 2021 [cited 2023 May 31];15(1–2):1–175. Available from: http://arxiv.org/abs/2008.12595

44. Lee M, Min K. MGCVAE: Multi-Objective Inverse Design via Molecular Graph Conditional Variational Autoencoder. J Chem Inf Model [Internet]. 2022 Jun 27 [cited 2023 May 31];62(12):2943–50. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.2c00487

45. De Cao N, Kipf T. MolGAN: An implicit generative model for small molecular graphs [Internet]. arXiv; 2022 [cited 2023 May 31]. Available from: http://arxiv.org/abs/1805.11973

46. Xue D, Gong Y, Yang Z, Chuai G, Qu S, Shen A, et al. Advances and challenges in deep generative models for de novo molecule generation. WIREs Comput Mol Sci [Internet]. 2019 [cited 2023 May 31];9(3). Available from: https://onlinelibrary.wiley.com/doi/10.1002/wcms.1395

47. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Applying and improving AlphaFold at CASP14. Proteins [Internet]. 2021 [cited 2023 May 31];89(12):1711–21. Available from: https://onlinelibrary.wiley.com/doi/10.1002/prot.26257

48. Lopez R, Gayoso A, Yosef N. Enhancing scientific discoveries in molecular biology with deep generative models. Molecular Systems Biology [Internet]. 2020 [cited 2023 May 31];16(9):e9198. Available from: https://www.embopress.org/doi/10.15252/msb.20199198

49. Simidjievski N, Bodnar C, Tariq I, Scherer P, Andres Terre H, Shams Z, et al. Variational Autoencoders for Cancer Data Integration: Design Principles and Computational Practice. Front Genet [Internet]. 2019 Dec 11 [cited 2023 May 31];10:1205. Available from: https://www.frontiersin.org/article/10.3389/fgene.2019.01205/full

50. Caterini AL, Doucet A, Sejdinovic D. Hamiltonian Variational Auto-Encoder. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2018 [cited 2023 May 31]. Available from: https://proceedings.neurips.cc/paper/2018/hash/3202111cf90e7c816a472aaceb72b0df-Abstract.html

51. Kingma DP, Welling M. An Introduction to Variational Autoencoders. MAL [Internet]. 2019 Nov 27 [cited 2023 May 31];12(4):307–92. Available from: https://www.nowpublishers.com/article/Details/MAL-056

52. Liu Q, Allamanis M, Brockschmidt M, Gaunt A. Constrained Graph Variational Autoencoders for Molecule Design. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2018 [cited 2023 May 31]. Available from: https://proceedings.neurips.cc/paper/2018/hash/b8a03c5c15fcfa8dae0b03351eb1742f-Abstract.html

53. Yan X, Yang J, Sohn K, Lee H. Attribute2Image: Conditional Image Generation from Visual Attributes. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision – ECCV 2016 [Internet]. Cham: Springer International Publishing; 2016 [cited 2023 May 31]. p. 776–91. Available from: http://link.springer.com/10.1007/978-3-319-46493-0_47

54. Song T, Sun J, Chen B, Peng W, Song J. Latent Space Expanded Variational Autoencoder for Sentence Generation. IEEE Access [Internet]. 2019 [cited 2023 May 31];7:144618–27. Available from: https://ieeexplore.ieee.org/document/8853312/

55. Gómez-Bombarelli R, Wei JN, Duvenaud D, Hernández-Lobato JM, Sánchez-Lengeling B, Sheberla D, et al. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. ACS Cent Sci [Internet]. 2018 Feb 28 [cited 2023 May 31];4(2):268–76. Available from: https://pubs.acs.org/doi/10.1021/acscentsci.7b00572

56. Ghojogh B, Ghodsi A, Karray F, Crowley M. Factor Analysis, Probabilistic Principal Component Analysis, Variational Inference, and Variational Autoencoder: Tutorial and Survey [Internet]. arXiv; 2022 [cited 2023 May 31]. Available from: http://arxiv.org/abs/2101.00734

57. Lee M, Seok J. Controllable Generative Adversarial Network. IEEE Access [Internet]. 2019 [cited 2023 Jun 12];7:28158–69. Available from: https://ieeexplore.ieee.org/document/8641270/

58. Mirza M, Osindero S. Conditional Generative Adversarial Nets [Internet]. arXiv; 2014 [cited 2023 Jun 12]. Available from: http://arxiv.org/abs/1411.1784

59. Celard P, Iglesias EL, Sorribes-Fdez JM, Romero R, Vieira AS, Borrajo L. A survey on deep learning applied to medical images: from simple artificial neural networks to generative models. Neural Comput & Applic [Internet]. 2023 [cited 2023 Jun 12];35(3):2291–323. Available from: https://link.springer.com/10.1007/s00521-022-07953-4

60. Gm H, Gourisaria MK, Pandey M, Rautaray SS. A comprehensive survey and analysis of generative models in machine learning. Computer Science Review [Internet]. 2020 [cited 2023 Jun 12];38:100285. Available from: https://linkinghub.elsevier.com/retrieve/pii/S1574013720303853

61. Guo Z, Liu J, Wang Y, Chen M, Wang D, Xu D, et al. Diffusion Models in Bioinformatics: A New Wave of Deep Learning Revolution in Action [Internet]. arXiv; 2023 [cited 2023 Jun 12]. Available from: http://arxiv.org/abs/2302.10907

62. Zou H, Kim ZM, Kang D. Diffusion Models in NLP: A Survey [Internet]. arXiv; 2023 [cited 2023 Jun 12]. Available from: http://arxiv.org/abs/2305.14671

63. Sohl-Dickstein J, Weiss E, Maheswaranathan N, Ganguli S. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In: Proceedings of the 32nd International Conference on Machine Learning [Internet]. PMLR; 2015 [cited 2023 Jun 12]. p. 2256–65. Available from: https://proceedings.mlr.press/v37/sohl-dickstein15.html

64. S. Non-Denoising Forward-Time Diffusions. 2022 Jan 28 [cited 2023 Jun 12]; Available from: https://openreview.net/forum?id=oVfIKuhqfC

65. Tay Y, Dehghani M, Bahri D, Metzler D. Efficient Transformers: A Survey. ACM Comput Surv [Internet]. 2023 Jul 31 [cited 2023 Jun 12];55(6):1–28. Available from: https://dl.acm.org/doi/10.1145/3530811

66. Casola S, Lauriola I, Lavelli A. Pre-trained transformers: an empirical comparison. Machine Learning with Applications [Internet]. 2022 Sep 15 [cited 2023 Jun 12];9:100334. Available from: https://www.sciencedirect.com/science/article/pii/S2666827022000445

67. Kobyzev I, Prince SJD, Brubaker MA. Normalizing Flows: An Introduction and Review of Current Methods. IEEE Trans Pattern Anal Mach Intell [Internet]. 2021 Nov 1 [cited 2023 Jun 12];43(11):3964–79. Available from: https://ieeexplore.ieee.org/document/9089305/

68. Zhang J, Chen H. De Novo Molecule Design Using Molecular Generative Models Constrained by Ligand–Protein Interactions. J Chem Inf Model [Internet]. 2022 Jul 25 [cited 2023 Jun 12];62(14):3291–306. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.2c00177

69. Lim J, Ryu S, Kim JW, Kim WY. Molecular generative model based on conditional variational autoencoder for de novo molecular design. Journal of Cheminformatics [Internet]. 2018 Jul 11 [cited 2023 Jun 12];10(1):31. Available from: https://doi.org/10.1186/s13321-018-0286-7

70. Méndez-Lucio O, Baillif B, Clevert DA, Rouquié D, Wichard J. De novo generation of hit-like molecules from gene expression signatures using artificial intelligence. Nat Commun [Internet]. 2020 Jan 3 [cited 2023 Jun 12];11(1):10. Available from: https://www.nature.com/articles/s41467-019-13807-w

71. Uludoğan G, Ozkirimli E, Ulgen KO, Karalı N, Özgür A. Exploiting pretrained biochemical language models for targeted drug design. Bioinformatics [Internet]. 2022 Sep 16 [cited 2023 Jun 12];38(Supplement_2):ii155–61. Available from: https://academic.oup.com/bioinformatics/article/38/Supplement_2/ii155/6702010

72. Wang M, Hsieh CY, Wang J, Wang D, Weng G, Shen C, et al. RELATION: A Deep Generative Model for Structure-Based De Novo Drug Design. J Med Chem [Internet]. 2022 Jul 14 [cited 2023 Jun 12];65(13):9478–92. Available from: https://pubs.acs.org/doi/10.1021/acs.jmedchem.2c00732

73. Skalic M, Sabbadin D, Sattarov B, Sciabola S, De Fabritiis G. From Target to Drug: Generative Modeling for the Multimodal Structure-Based Ligand Design. Mol Pharmaceutics [Internet]. 2019 Oct 7 [cited 2023 Jun 12];16(10):4282–91. Available from: https://pubs.acs.org/doi/10.1021/acs.molpharmaceut.9b00634

74. Kadurin A, Nikolenko S, Khrabrov K, Aliper A, Zhavoronkov A. druGAN: An Advanced Generative Adversarial Autoencoder Model for de Novo Generation of New Molecules with Desired Molecular Properties in Silico. Mol Pharmaceutics [Internet]. 2017 Sep 5 [cited 2023 Jun 12];14(9):3098–104. Available from: https://pubs.acs.org/doi/10.1021/acs.molpharmaceut.7b00346

75. Martinelli DD. Generative machine learning for de novo drug discovery: A systematic review. Computers in Biology and Medicine [Internet]. 2022 [cited 2023 Jun 12];145:105403. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0010482522001950

76. Angermueller C, Pärnamaa T, Parts L, Stegle O. Deep learning for computational biology. Molecular Systems Biology [Internet]. 2016 [cited 2023 Jun 12];12(7):878. Available from: https://www.embopress.org/doi/10.15252/msb.20156651

77. Wainberg M, Merico D, Delong A, Frey BJ. Deep learning in biomedicine. Nat Biotechnol [Internet]. 2018 [cited 2023 Jun 12];36(9):829–38. Available from: https://www.nature.com/articles/nbt.4233

78. Manning BD, Cantley LC. AKT/PKB Signaling: Navigating Downstream. Cell [Internet]. 2007 [cited 2023 Jun 12];129(7):1261–74. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0092867407007751

79. George B, Gui B, Raguraman R, Paul AM, Nakshatri H, Pillai MR, et al. AKT1 Transcriptomic Landscape in Breast Cancer Cells. Cells [Internet]. 2022 Jul 25 [cited 2023 Jun 12];11(15):2290. Available from: https://www.mdpi.com/2073-4409/11/15/2290

80. Banno E, Togashi Y, De Velasco MA, Mizukami T, Nakamura Y, Terashima M, et al. Clinical significance of Akt2 in advanced pancreatic cancer treated with erlotinib. International Journal of Oncology [Internet]. 2017 [cited 2023 Jun 12];50(6):2049–58. Available from: https://www.spandidos-publications.com/10.3892/ijo.2017.3961

81. Stahl JM, Sharma A, Cheung M, Zimmerman M, Cheng JQ, Bosenberg MW, et al. Deregulated Akt3 Activity Promotes Development of Malignant Melanoma. Cancer Research [Internet]. 2004 Oct 1 [cited 2023 Jun 12];64(19):7002–10. Available from: https://aacrjournals.org/cancerres/article/64/19/7002/511766/Deregulated-Akt3-Activity-Promotes-Development-of

82. Shariati M, Meric-Bernstam F. Targeting AKT for cancer therapy. Expert Opinion on Investigational Drugs [Internet]. 2019 Nov 2 [cited 2023 Jun 12];28(11):977–88. Available from: https://www.tandfonline.com/doi/full/10.1080/13543784.2019.1676726

83. Krissinel E, Henrick K. Inference of Macromolecular Assemblies from Crystalline State. Journal of Molecular Biology [Internet]. 2007 [cited 2023 Jun 12];372(3):774–97. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0022283607006420

84. Burley SK, Berman HM, Bhikadiya C, Bi C, Chen L, Di Costanzo L, et al. RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. Nucleic Acids Research [Internet]. 2019 Jan 8 [cited 2023 Jun 12];47(D1):D464–74. Available from: https://academic.oup.com/nar/article/47/D1/D464/5144139

85. Du K, Tsichlis PN. Regulation of the Akt kinase by interacting proteins. Oncogene [Internet]. 2005 Nov 14 [cited 2023 Jun 12];24(50):7401–9. Available from: https://www.nature.com/articles/1209099

86. Addie M, Ballard P, Buttar D, Crafter C, Currie G, Davies BR, et al. Discovery of 4-Amino- N -[(1 S)-1-(4-chlorophenyl)-3-hydroxypropyl]-1-(7 H -pyrrolo[2,3- d]pyrimidin-4-yl)piperidine-4-carboxamide (AZD5363), an Orally Bioavailable, Potent Inhibitor of Akt Kinases. J Med Chem [Internet]. 2013 Mar 14 [cited 2023

Jun 12];56(5):2059–73. Available from: https://pubs.acs.org/doi/10.1021/jm301762v

87. Wishart DS, Feunang YD, Guo AC, Lo EJ, Marcu A, Grant JR, et al. DrugBank 5.0: a major update to the DrugBank database for 2018. Nucleic Acids Research [Internet]. 2018 Jan 4 [cited 2023 Jun 12];46(D1):D1074–82. Available from: http://academic.oup.com/nar/article/46/D1/D1074/4602867

88. Morris GM, Huey R, Lindstrom W, Sanner MF, Belew RK, Goodsell DS, et al. AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. J Comput Chem [Internet]. 2009 [cited 2023 Jun 12];30(16):2785–91. Available from: https://onlinelibrary.wiley.com/doi/10.1002/jcc.21256

89. Piana S, Lindorff-Larsen K, Dirks RM, Salmon JK, Dror RO, Shaw DE. Evaluating the Effects of Cutoffs and Treatment of Long-range Electrostatics in Protein Folding Simulations. Verma C, editor. PLoS ONE [Internet]. 2012 Jun 29 [cited 2023 Jun 12];7(6):e39918. Available from: https://dx.plos.org/10.1371/journal.pone.0039918

90. Dwivedi VP, Bresson X. A Generalization of Transformer Networks to Graphs [Internet]. arXiv; 2021 [cited 2023 Jun 12]. Available from: http://arxiv.org/abs/2012.09699

91. Vignac C, Krawczuk I, Siraudin A, Wang B, Cevher V, Frossard P. DiGress: Discrete Denoising diffusion for graph generation [Internet]. arXiv; 2023 [cited 2023 Jun 12]. Available from: http://arxiv.org/abs/2209.14734

92. Gaudelet T, Day B, Jamasb AR, Soman J, Regep C, Liu G, et al. Utilizing graph machine learning within drug discovery and development. Briefings in Bioinformatics [Internet]. 2021 Nov 5 [cited 2023 Jun 24];22(6):bbab159. Available from: https://academic.oup.com/bib/article/doi/10.1093/bib/bbab159/6278145

93. Zhu Y, Du Y, Wang Y, Xu Y, Zhang J, Liu Q, et al. A Survey on Deep Graph Generation: Methods and Applications [Internet]. arXiv; 2022 [cited 2023 Jun 24]. Available from: http://arxiv.org/abs/2203.06714

94. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention Is All You Need [Internet]. arXiv; 2017 [cited 2023 Jun 24]. Available from: http://arxiv.org/abs/1706.03762

95. McKenna M, Balasuriya N, Zhong S, Li SSC, O'Donoghue P. Phospho-Form Specific Substrates of Protein Kinase B (AKT1). Front Bioeng Biotechnol [Internet]. 2021 Feb 3 [cited 2023 Jun 24];8:619252. Available from: https://www.frontiersin.org/articles/10.3389/fbioe.2020.619252/full

96. Polykovskiy D, Zhebrak A, Sanchez-Lengeling B, Golovanov S, Tatanov O, Belyaev S, et al. Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models. Front Pharmacol [Internet]. 2020 Dec 18 [cited 2023 Jun 24];11:565644. Available from: https://www.frontiersin.org/articles/10.3389/fphar.2020.565644/full

97. Landrum G. RDKit Documentation [Internet]. 2019. Available from: https://buildmedia.readthedocs.org/media/pdf/rdkit/latest/rdkit.pdf

98. Mahmood O, Mansimov E, Bonneau R, Cho K. Masked graph modeling for molecule generation. Nat Commun [Internet]. 2021 May 26 [cited 2023 Jun 24];12(1):3156. Available from: https://www.nature.com/articles/s41467-021-23415-2

99. Bagal V, Aggarwal R, Vinod PK, Priyakumar UD. MolGPT: Molecular Generation Using a Transformer-Decoder Model. J Chem Inf Model [Internet]. 2022 May 9 [cited 2023 Jun 28];62(9):2064–76. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.1c00600

100. Wang M, Hsieh CY, Wang J, Wang D, Weng G, Shen C, et al. RELATION: A Deep Generative Model for Structure-Based De Novo Drug Design. J Med Chem [Internet]. 2022 Jul 14 [cited 2023 Jun 28];65(13):9478–92. Available from: https://pubs.acs.org/doi/10.1021/acs.jmedchem.2c00732

101. Olivecrona M, Blaschke T, Engkvist O, Chen H. Molecular de-novo design through deep reinforcement learning. J Cheminform [Internet]. 2017 [cited 2023 Jun 28];9(1):48. Available from: https://jcheminf.biomedcentral.com/articles/10.1186/s13321-017-0235-x

102. Xie Y, Shi C, Zhou H, Yang Y, Zhang W, Yu Y, et al. MARS: Markov Molecular Sampling for Multi-objective Drug Discovery [Internet]. arXiv; 2021 [cited 2023 Jun 28]. Available from: http://arxiv.org/abs/2103.10432

103. Grisoni F, Moret M, Lingwood R, Schneider G. Bidirectional Molecule Generation with Recurrent Neural Networks. J Chem Inf Model [Internet]. 2020 Mar 23 [cited 2023 Jun 28];60(3):1175–83. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.9b00943

104. Zhou Z, Kearnes S, Li L, Zare RN, Riley P. Optimization of Molecules via Deep Reinforcement Learning. Sci Rep [Internet]. 2019 Jul 24 [cited 2023 Jun 28];9(1):10752. Available from: https://www.nature.com/articles/s41598-019-47148-x

105. Fang Y, Pan X, Shen HB. De novo drug design by iterative multiobjective deep reinforcement learning with graph-based molecular quality assessment. Valencia A, editor. Bioinformatics [Internet]. 2023 Apr 3 [cited 2023 Jun 28];39(4):btad157. Available from:

https://academic.oup.com/bioinformatics/article/doi/10.1093/bioinformatics/btad157/7085596

106. Wei R, Mahmood A. Recent Advances in Variational Autoencoders With Representation Learning for Biomedical Informatics: A Survey. IEEE Access [Internet]. 2021 [cited 2023 Jun 28];9:4939–56. Available from: https://ieeexplore.ieee.org/document/9311619/

107. Dan Y, Zhao Y, Li X, Li S, Hu M, Hu J. Generative adversarial networks (GAN) based efficient sampling of chemical composition space for inverse design of inorganic materials. npj Comput Mater [Internet]. 2020 Jun 26 [cited 2023 Jun 28];6(1):84. Available from: https://www.nature.com/articles/s41524-020-00352-0

108. Ho J, Jain A, Abbeel P. Denoising Diffusion Probabilistic Models [Internet]. arXiv; 2020 [cited 2023 Jun 28]. Available from: http://arxiv.org/abs/2006.11239

109. Tan C, Gao Z, Li SZ. Target-aware Molecular Graph Generation [Internet]. arXiv; 2022 [cited 2023 Jun 28]. Available from: http://arxiv.org/abs/2202.04829

110. Kobyzev I, Prince SJD, Brubaker MA. Normalizing Flows: An Introduction and Review of Current Methods. IEEE Trans Pattern Anal Mach Intell [Internet]. 2021 Nov 1 [cited 2023 Jun 28];43(11):3964–79. Available from: https://ieeexplore.ieee.org/document/9089305/

111. Liu J, Kumar A, Ba J, Kiros J, Swersky K. Graph Normalizing Flows. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2019 [cited 2023 Jun 28]. Available from: https://proceedings.neurips.cc/paper/2019/hash/1e44fdf9c44d7328fecc02d677ed704d-Abstract.html

112. Durant JL, Leland BA, Henry DR, Nourse JG. Reoptimization of MDL Keys for Use in Drug Discovery. J Chem Inf Comput Sci [Internet]. 2002 Nov 1 [cited 2023 Jun 29];42(6):1273–80. Available from: https://pubs.acs.org/doi/10.1021/ci010132r

113. Brown N, Fiscato M, Segler MHS, Vaucher AC. GuacaMol: Benchmarking Models for de Novo Molecular Design. J Chem Inf Model [Internet]. 2019 Mar 25 [cited 2023 Jun 29];59(3):1096–108. Available from: https://pubs.acs.org/doi/10.1021/acs.jcim.8b00839

114. Liu M, Luo Y, Uchino K, Maruhashi K, Ji S. Generating 3D Molecules for Target Protein Binding [Internet]. arXiv; 2022 [cited 2023 Jun 29]. Available from: http://arxiv.org/abs/2204.09410

115. Yu Y, Lu S, Gao Z, Zheng H, Ke G. Do Deep Learning Models Really Outperform Traditional Approaches in Molecular Docking? [Internet]. arXiv; 2023 [cited 2023 Jun 29]. Available from: http://arxiv.org/abs/2302.07134

116. Zhang Y, Zhang Y, Yan D, Deng S, Yang Y. Revisiting Graph-based Recommender Systems from the Perspective of Variational Auto-Encoder. ACM Trans Inf Syst [Internet]. 2023 Jul 31 [cited 2023 Jun 29];41(3):1–28. Available from: https://dl.acm.org/doi/10.1145/3573385

117. Teng Y, Wang L. Structured Sparse R-CNN for Direct Scene Graph Generation. In 2022 [cited 2023 Jun 29]. p. 19437–46. Available from: https://openaccess.thecvf.com/content/CVPR2022/html/Teng_Structured_Sparse_R-CNN_for_Direct_Scene_Graph_Generation_CVPR_2022_paper.html

118. Karras T, Aittala M, Hellsten J, Laine S, Lehtinen J, Aila T. Training Generative Adversarial Networks with Limited Data. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2020 [cited 2023 Jun 29]. p. 12104–14. Available from: https://proceedings.neurips.cc/paper/2020/hash/8d30aa96e72440759f74bd2306c1fa3d-Abstract.html

119. Chen J, Zheng S, Song Y, Rao J, Yang Y. Learning Attributed Graph Representations with Communicative Message Passing Transformer [Internet]. arXiv; 2021 [cited 2023 Jun 29]. Available from: http://arxiv.org/abs/2107.08773

120. Thanh-Tung H, Tran T, Venkatesh S. Improving Generalization and Stability of Generative Adversarial Networks [Internet]. arXiv; 2019 [cited 2023 Jun 29]. Available from: http://arxiv.org/abs/1902.03984

121. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X, et al. Improved Techniques for Training GANs. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2016 [cited 2023 Jun 29]. Available from: https://proceedings.neurips.cc/paper/2016/hash/8a3363abe792db2d8761d6403605aeb7-Abstract.html

122. Weng L. From GAN to WGAN [Internet]. arXiv; 2019 [cited 2023 Jun 29]. Available from: http://arxiv.org/abs/1904.08994

123. Zhang Z, Li M, Yu J. On the convergence and mode collapse of GAN. In: SIGGRAPH Asia 2018 Technical Briefs [Internet]. Tokyo Japan: ACM; 2018 [cited 2023 Jun 29]. p. 1–4. Available from: https://dl.acm.org/doi/10.1145/3283254.3283282

124. Arjovsky M, Chintala S, Bottou L. Wasserstein Generative Adversarial Networks. In: Proceedings of the 34th International Conference on Machine Learning [Internet]. PMLR; 2017 [cited 2023 Jun 29]. p. 214–23. Available from: https://proceedings.mlr.press/v70/arjovsky17a.html

125. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC. Improved Training of Wasserstein GANs. In: Advances in Neural Information Processing Systems

[Internet]. Curran Associates, Inc.; 2017 [cited 2023 Jun 29]. Available from: https://proceedings.neurips.cc/paper_files/paper/2017/hash/892c3b1c6dccd52936e27cbd0ff683d6-Abstract.html

126. Guimaraes GL, Sanchez-Lengeling B, Outeiral C, Farias PLC, Aspuru-Guzik A. Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models [Internet]. arXiv; 2018 [cited 2023 Jun 29]. Available from: http://arxiv.org/abs/1705.10843

127. Krenn M, Häse F, Nigam A, Friederich P, Aspuru-Guzik A. Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. Mach Learn: Sci Technol [Internet]. 2020 Dec 1 [cited 2023 Jun 29];1(4):045024. Available from: https://iopscience.iop.org/article/10.1088/2632-2153/aba947

128. Rifaioglu AS, Atas H, Martin MJ, Cetin-Atalay R, Atalay V, Doğan T. Recent applications of deep learning and machine intelligence on in silico drug discovery: methods, tools and databases. Briefings in Bioinformatics [Internet]. 2019 Sep 27 [cited 2023 Aug 9];20(5):1878–912. Available from: https://academic.oup.com/bib/article/20/5/1878/5062947

129. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. Commun ACM [Internet]. 2020 Oct 22 [cited 2023 Aug 9];63(11):139–44. Available from: https://dl.acm.org/doi/10.1145/3422622

8. APPENDIX

8.1. EK-1: Etik Kurul İzin Belgesi

HACETTEPE ÜNİVERSİTESİ
GİRİŞİMSEL OLMAYAN KLİNİK ARAŞTIRMALAR ETİK KURULU

KURUL KARARI

OTURUM TARİHİ	OTURUM SAYISI	KARAR SAYISI
21.02.2023	2023/03	2023/03-38
Araştırma Numarası : GO 22/115	**Değerlendirme Tarihi :** 18.01.2022	

Üniversitemiz Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü Yapay Zeka Mühendisliği Anabilim Dalı öğretim üyelerinden Doç. Dr. Tunca DOĞAN'ın sorumlu araştırmacı olduğu, Atabey ÜNLÜ'nün yüksek lisans tezi olan, GO 22/115 kayıt numaralı *"Derin Çizge Öğrenmesi ile İlaç Adayı Moleküllerin Otomatik Şekilde Tasarımı"* başlıklı araştırma önerisi gerekçe, amaç, yaklaşım ve yöntemleri dikkate alınarak incelenmiş olup, 22 Şubat 2023 – 22 Şubat 2024 tarihleri arasında geçerli olmak üzere etik açıdan **uygun bulunmuştur.**

Çalışma tamamlandığında sonuçlarını içeren bir rapor örneğinin Etik Kurulumuza gönderilmesi gerekmektedir.

Prof. Dr. Nüket PAKSOY ERBAYDAR Kurul Başkanı	Prof. Dr. Güzide Burça AYDIN Kurul Üyesi	Prof. Dr. Mehmet Özgür UYANIK Kurul Üyesi	Prof. Dr. Ayşe KİN İŞLER Kurul Üyesi
Prof. Dr. Sibel PEHLİVAN Kurul Üyesi	Prof. Dr. Burcu Balam DOĞU Kurul Üyesi	Prof. Dr. Tolga YILDIRIM Kurul Üyesi	Prof. Dr. Hande GÜNEY DENİZ Kurul Üyesi
Doç. Dr. Betül ÇELEBİ SALTIK Kurul Üyesi	Doç. Dr. Merve BATUK Kurul Üyesi	Doç. Dr. Gülten IŞIK KOÇ Kurul Üyesi	Dr. Öğr. Üyesi Müge DEMİR Kurul Üyesi
Dr. Öğr. Üyesi Burcu Ersöz ALAN Kurul Üyesi	**İZİNLİ** Av. Buket ÇINAR Kurul Üyesi		

8.2. EK-6: Tez Çalışması Orijinallik Raporu

turnitin

Dijital Makbuz

Bu makbuz ödevinizin Turnitin'e ulaştığını bildirmektedir. Gönderiminize dair bilgiler şöyledir:

Gönderinizin ilk sayfası aşağıda gönderilmektedir.

Gönderen:	Atabey Ünlü
Ödev başlığı:	Atabey Ünlü - Biyoenformatik YL Tezi (son sürüm)
Gönderi Başlığı:	Biyoenformatik YL Tezi
Dosya adı:	AU_Tez_10.08.2023.pdf
Dosya boyutu:	4.63M
Sayfa sayısı:	104
Kelime sayısı:	26,785
Karakter sayısı:	151,007
Gönderim Tarihi:	10-Ağu-2023 10:37ÖÖ (UTC+0300)
Gönderim Numarası:	2143872498


T.C.
REPUBLIC OF TURKEY
HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF HEALTH SCIENCES

AUTOMATED DESIGN OF DRUG CANDIDATE MOLECULES WITH DEEP GRAPH LEARNING

Atabey ÜNLÜ

Program of Bioinformatics
MASTER'S THESIS

Ankara
2023

Biyoenformatik YL Tezi

ORİJİNALLİK RAPORU

%15	%12	%10	%6
BENZERLİK ENDEKSİ	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCİ ÖDEVLERİ

BİRİNCİL KAYNAKLAR

1	openaccess.hacettepe.edu.tr:8080 İnternet Kaynağı	%2
2	www.mdpi.com İnternet Kaynağı	%1
3	assets.researchsquare.com İnternet Kaynağı	%1
4	acikbilim.yok.gov.tr İnternet Kaynağı	<%1
5	dokumen.pub İnternet Kaynağı	<%1
6	depot.ceon.pl İnternet Kaynağı	<%1
7	academic.oup.com İnternet Kaynağı	<%1
8	Submitted to Hacettepe University Öğrenci Ödevi	<%1
9	www.researchgate.net İnternet Kaynağı	<%1

9. CURRICULUM VITAE