공학석사 학위논문

Chemical Space Embedding for FDA Approved Drugs Using Cascade Autoencoder

순차적 오토인코더 기반 FDA 승인 약물들의 화학 공간 임베딩

2021 년 8 월

서울대학교 대학원

컴퓨터 공학부

김 정 우

Chemical Space Embedding for FDA Approved Drugs Using Cascade Autoencoder

순차적 오토인코더 기반 FDA 승인 약물들의 화학 공간 임베딩

지도교수 김 선

이 논문을 공학석사 학위논문으로 제출함

2021 년 8 월

서울대학교 대학원

컴퓨터 공학부

김 정 우

김정우의 공학석사 학위논문을 인준함

2021 년 8 월

위 원 장 장병탁

부위원장 김선

위 원 황승원

Abstract

Chemical Space Embedding for FDA Approved Drugs Using Cascade Autoencoder

Jungwoo Kim
Department of Computer Science & Engineering
College of Engineering
Seoul National University

Drug discovery requires decade of expensive efforts to meet sufficient needs. Computer-Aided drug discovery (CADD) is an emerging field of study that aims to systematically reduce the time and cost of a new durg development by adapting computer science to identify structural and physical properties of chemical compounds used as drugs and derive new drug candidates with similar characteristics. In particular, it is most important to identify the characteristics of chemical compounds approved by the U.S. Food and Drug Administration (FDA). FDA approved chemical compounds are validated drugs in terms of toxicity, efficacy of drug and side effects. The question arises here how these chemical compounds are distributed in an embedding space. Traditionally, hand-crafted rule is the only way of constructing the chemical space. Traditional chemical compound representations have made it difficult to classify FDA approved chemical compounds. With the advent of the era of big data and artificial intelligence technology, deep learning is the leading technology that drives to build an embedding space. However, there is few adaptive methods to identify the embedding space of FDA approved chemical compounds.

In this work, I propose a framework that encodes features of FDA approved chemical compounds by constructing a discriminative embedding space. Various encoding methods were used to encode information from FDA approved chemical compounds. The proposed framework consists of three stacked deep autoencoder modules. The proposed framework effectively integrate the information of the chemical compounds by cascade modeling. Connected three autoencoder modules in cascade is used to continuously use latent representation learned from previous modules. Whether FDA approved chemical compounds have discriminative regions in the embedding space is well visualized by the proposed framework. And perform machine learning classification tasks to evaluate whether the latent representation effectively characterize the FDA approval information. The proposed framework incorporates complex representation information to understand the embedding of FDA drugs. Ultimately, the framework proposed in this paper can be used as an embedding method for determining whether or not new drug candidates will be approved.

Keywords: FDA Approved drug, Cascade Autoencoder, Chemical space embedding

Student Number: 2019-24822

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.1.1 Chemical space

Chemical space refers to an embedding space of all possible chemical compounds that satisfy a common boundary condition in terms of common structural and physical properties. Understanding chemical space is a key research topic in AI based drug discovery. However, characterization of the chemical space is a challenging problem that much has been veiled. By leveraging domain knowledge such as molecular weight or what functional group that chemical compound has to effectively build a chemical space, I can characterize the general properties of the chemical compounds bounded in the chemical space. In order to calculate the decision boundaries that make up the chemical space, it is important to convert chemical compounds to a computable format. Typically, A Simplified Molecular-Input Line-Entry System (SMILES) is used to represent chemical compound. As depicted in Figure 1.1, SMILES uses element symbols suca as C(carbon), O(oxygen) to represent structural connections of

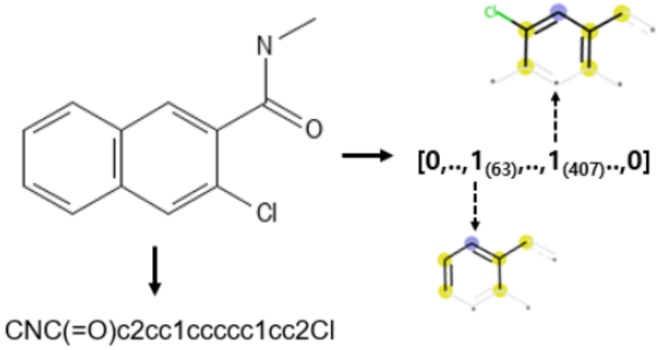

Figure 1.1: Chemical Compound to SMILES representation and ECFP

compounds as linear strings. Another method to represent chemical compound is an Extended Connectivity Fingerprint (ECFP) based on a Morgan Fingerprint. These representations can input to deep learning models by converting chemical compounds to computable format. are na computable form that is directly fed into machine learning models.

In order to build chemical embedding space, deep autoencoder is introduced as it enables manifold learning through non-linear dimensionality reduction. In this perspective, deep autoencoder can effectively extract latent features of a chemical compound to characterize the decision boundary of a chemical space I want.

1.1.2 FDA Approval of chemical drugs

When developing a new pharmaceuticals, new drug candidates must undergo pre-clinical stage, clinical pharmacology, clinical investigation, clinical trials and post marketing surveillance post marketing clinical trials in order to evaluate whether the candidate is effective and safe enough for the purpose of treatment. Based on the expensive investigations, pharmaceutical companies apply to the FDA for approval. As such, the process of getting a new drug

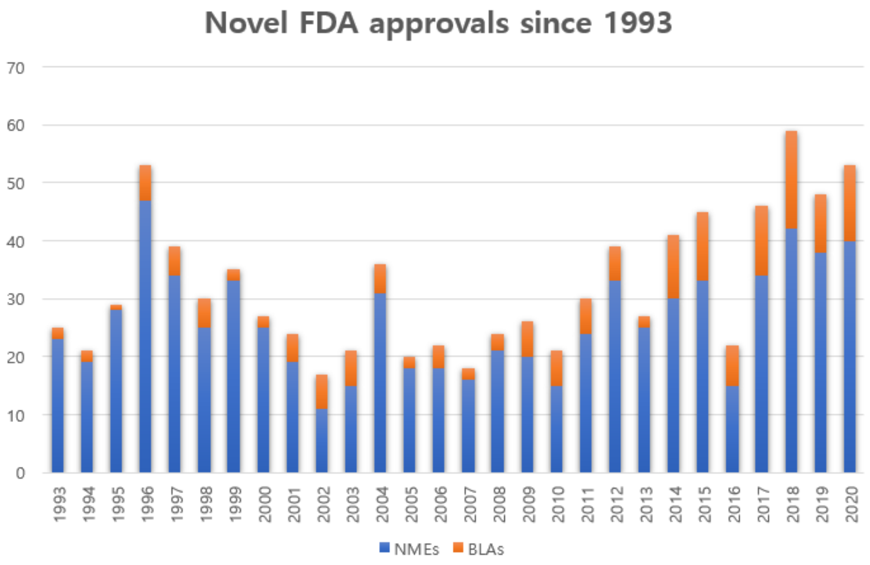

Figure 1.2: Novel FDA Approvals since 1993

approved by FDA takes a long time. And traditionally, developing a new drug takes 10.5 years on average from clinical trials to approval and finding new drug candidates takes an average of 5 years. As shown in Figure 1.2 (Mullard, 2021), average number of novel FDA approvals annually is less than 60. Recently, cheminformatics researchers try to find common properties of chemical compounds that can be a new drug candidate. For example, Ligand-Based Virtual Screening (LBVS) is a method of calculate similarity based on fingerprint or molecular descriptor, based on the idea that chemical compounds with similar properties have similar target-binding affinity(Xia *et al.*, 2014). From this perspective, I attempt to map compounds with similar properties to similar locations in the embedding space. In particular, FDA-approved drugs are drugs that have been tested for toxicity and side effects, etc., it is crucial to understand what chemical properties in existing approved drugs have to increase the likelihood of being approved.

In this work, I try to characterize the chemical space by encoding properties of chemical compound and analyze what decision boundaries FDA approved drugs have in the chemical space. I apply deep learning technology to embed the properties inherent in FDA approached drug discriminatively. From the view of virtual screening, if I were able to represent a discriminative region in the embedding space by drug characteristics, the cost of finding new drug candidates could be reduced.

1.2 Current Method and Limitation

With the advent of the era of big data and artificial intelligence, Deep Learning technology combined with chemical compound data is applied to many tasks, such as drug target interaction prediction, physical property prediction and representation learning. Under the assumption that similar chemical compounds have similar pharmacological effects, research is actively underway to identify and predict the structural properties and physical properties of chemical compound. Understanding the sophisticated properties of chemical compound structure will allow us to produce chemical compounds with the properties we want.

In the representation learning task, in general, unsupervised deep learning methods are used based on generative models to capture meaningful properties hidden in chemical compound. Most methods aim to create a representation that encodes the desired characteristics using 2-dimensional structure of chemical compound. For example, a model applied Word2Vec technique with the substructure of chemical compound as unique word(Jaeger *et al.*, 2018), a model for representation learning by decomposing the structure of chemical compound into a junction tree(Jin *et al.*, 2018), and some models encode how chemical compounds are distributed using variational autoencoder(Winter *et al.*, 2019).

However, prior methods only encode fragmented information, So I proposed integrated framework to combine multiple information of chemical compound.

1.3 Problem Statement and Contributions

Considering diverse information in chemical compound, I aim to integrate different types of separately learned chemical compound representation to create integrated representation from a sophisticated perspective and to find the decision boundary that FDA approved drugs have in the embedding space. As follow, there are my two main goals in this study.

- I estabilish a discriminative embedding space of FDA approved drugs and discontinued drugs. In order to achieve this goal, I propose a framework based on cascade autoencoder to construct a discriminative chemical embedding space by extracting latent feature.
- I evaluate my framework through traditional machine learning classification tasks on whether my model effectively learned feature.

INPUT The input of the proposed framework is an information vectors extracted from three pretrained models.
OUTPUT The output of the proposed framework is the latent feature vector of chemical compound. In this work, I perform principal component analysis to visualize latent feature vectors from the proposed framework. And I perform machine learning classification tasks to verify that the proposed framework effectively encodes information from the chemical compound.

CONTRIBUTION

- I proposed a novel cascade autoencoder model for characterizing chemical embedding space using multiple information of chemical compound.

- Based on my framework, I extracted latent representation of chemical compound that contributes to classify whether FDA approved. And improved classification performance compared with traditional method.

- I visualized the created chemical embedding spaces. Even if I did not characterize the chemical embedding space well, I attempted to increase the possibility.

Chapter 2

Related Works

In this Chapter, I introduce the state-of-the-art methods in cascade autoencoder and chemical space embedding.

2.1 Cascade Autoencoder

Cascade model is the movement of information in a top-down manner. Cascade model leverages the representation from the previous stage in the next stage to construct a better representation. Cascade modeling is commonly used in the field of image processing and video detection. Diverse methods using cascade autoencoder have been proposed. Cascade models of adversarial autoencoder and convolutional autoencoder were used to detect video anomalies(Li *et al.*, 2020). The first module proactively identifies abnormal video cuboids, then the second module classifies specific abnormal patches in each abnormal cuboid. And there is also a cascade autoencoder for multi-label classification of scene data.(Law and Ghosh, 2019). In the field of image processing, cascade marginalized denoising autoencoder and non-negative sparse autoencoder was proposed to solve hyperspectral image unmixing problem(Guo *et al.*, 2015). In addition, cascade denoising autoencoders have also been used to solve noise reduction in single-particle Cryo-EM images(Lei and Yang, 2020).

2.2 Chemical Space Embedding Methods

The research of chemical space has traditionally been addressed in the field of combinatorial chemistry. Combinatorial chemistry consists of chemical synthesis, which allows the preparation of numerous compounds in a single process. These composite libraries can be made from mixtures, individual compound sets, or chemical structures produced by computer software. Traditional methods are mapping algorithms that rely on expert's hand-crafted rule. A typical example is ChemGPS, which performs principal component analysis on a pre-defined molecular structure descriptor to create a druglike chemical space.(Oprea and Gottfries, 2001). There are also ChemGPS-NPs that have improved ChemGPS by adding natural product classes(Rosén *et al.*, 2009).

Recently, not only has the popularity of deep learning methods but also the increasing power of computation has led to more and more research on deep learning methods applied in the field of cheminformatics. Unlike traditional hand-crafted methods, non-linear representation learning has become possible, research is being conducted to interpret chemical spaces in diverse ways. For examples, there are a model that uses autoencoder to characterize a hyperbolic space using chemical compound representation and drug hierarchy together(Yu *et al.*, 2020), a model to encode a molecular structure to build chemical embedding space using hypergraph variational autoencoder with metric learning(Koge *et al.*, 2021). And even in the field of material discovery, there is a model combined CNN with density function theory to embed crystal site.(Choubisa *et al.*, 2020).

Chapter 3

Methods and Materials

3.1 Notation and Problem Definition

Throughout this paper, I use uppercase characters to denote matrices and lowercase characters denote vectors. Unless specifically specified, the notations used in this paper are shown in the Table 3.1. And I define the set of definitions required to understand the paper.

Definition 1.(*Chemical compound, Drug Molecule*) Chemical compound is a chemical substance composed of two or more different chemically bonded chemical elements. Molecules are the smallest unit particles with the chemical properties of each substance. The definition of a molecule contains a chemical compound as a larger concept. However, researchers in the field of cheminformatics generally use chemical compounds and drug molecules in the same meaning, so in this paper I use chemical compounds and drug molecules in the same meaning.

Definition 2.(*Encoding, vector representation*) The definition of encoding is to encode/encrypt information. Encoding refers to the encoding methodol-

Table 3.1: Commonly Used Notations

Notations	Descriptions
D	the dimension of input.
d	the dimension of latent feature.
$x_i \in [0,1]^D$	i-th input vector.
$\widetilde{x_i} \in [0,1]^D$	i-th output vector.
$z_i \in [0,1]^d$	i-th latent vector.
y_i	class label.
c_{y_i}	i-th class center in d dimension.
λ	loss ratio.
n	the number of data points.
m	mini-batch size.
$\|\|$	concatenate function.
L_C	center loss.
L_{rec}	reconstruction loss.
W, θ	Learnable model parameters.

ogy, and vector presentation refers to the encoding output. In this paper, I define methods for converting chemical compounds to vector presentation as encoding.

3.2 Chemical Compound Encoding Process

In this work, I applied diverse encoding methods that used SMILES representation of FDA approved and discontinued drugs. I combined traditional representation encoding method such as Morgan Fingerprint and deep learning based representation encoding technology. The descriptions of the methods used are described below.

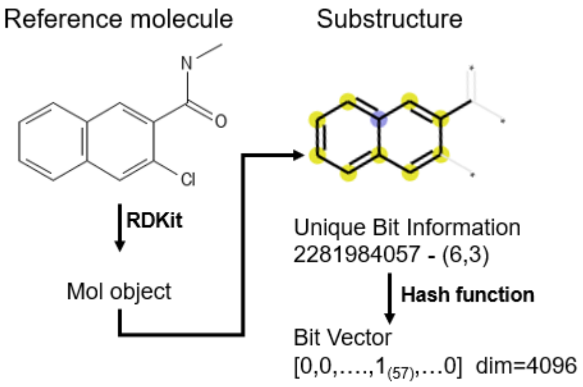

Figure 3.1: An Example of Generating Morgan Fingerprints

3.2.1 Morgan Fingerprints

Morgan Fingerprint is a method of encoding based on Morgan's algorithm in 1969(Morgan, 1965). All substructures under the k-nearest neighbor size that the chemical compound has are stored as unique bit information in a predefined manner(Rogers and Hahn, 2010). To encode unique bit information into a vector of a certain length, map bit information to the vector using hash function. In this case, too small vector size can lead to bit collision problems where different substructures are mapped to the same column information because they are generated using hash function. I use RDKit(Landrum *et al.*, 2006) library to convert the SMILES string of the chemical compound to mol object, and use mol object to generate the Morgan Fingerprint. As shown in Figure 3.1, unique bit information 2281984057 of the chemical compound indicates the substructure of the 3-nearest neighbor from 6^{th}atom, that is mapped to the 57^{th} component of the vector when converted to bit vector.

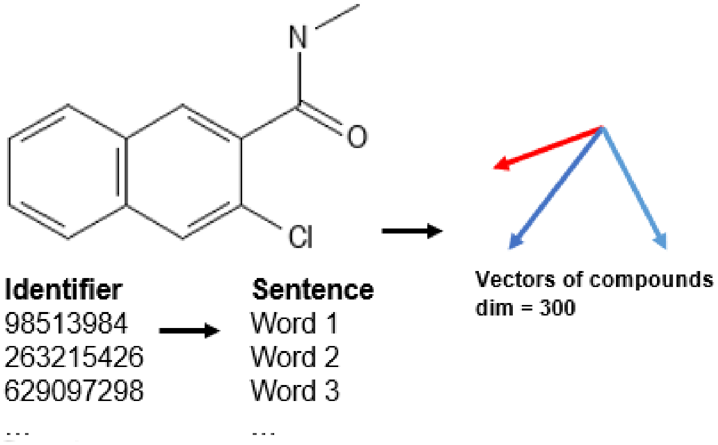

Figure 3.2: Mol2vec

3.2.2 Mol2vec

Mol2vec is a Word2Vec(Mikolov *et al.*, 2013) based substructure encoding method of chemical compound. Generating the aforementioned Morgan Fingerprint in the presence of a given chemical compound provides unique bit information. Considering this bit information as a word, the entire chemical compound described as a sentence of bit information. In other words, many substructures are mapped in unique word to generate vector representation. In this work, I used a pre-trained model using ZINC version 15(Irwin and Shoichet, 2005) and ChEMBL version 23(Gaulton *et al.*, 2012) databases. When creating corpus, each substructure up to 1-nearest neighbor was mapped into unique word. As shown in Figure 3.2, I set output dimension to 300 to convert one SMILES data into a vector representation of 300 dimension.

3.2.3 Junction Tree Variational Autoencoder

Another SMILES encoding method is the junction tree variational autoencoder. This method is molecular graph based chemical compound encoding

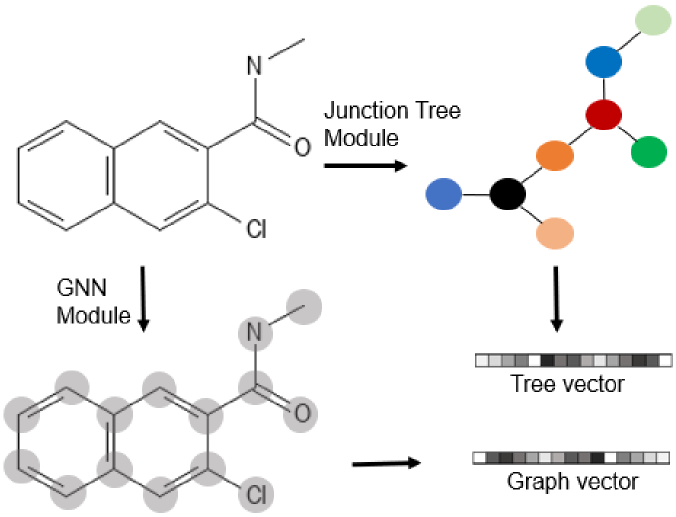

Figure 3.3: Juction Tree Variational Autoencoder

model. As shown in Figure 3.3, it consists of two modules: the Molecular Graph Neural Network (GNN) and the Junction Tree encoding module. Molecular GNN generates mol objects from SMILES presentation and then makes mol objects a molular graph. Where node is atom, edge is bond, node attributes encode the properties of each atom, edge attributes contain information such as bond order, and so on. Using this generated graph, perform a message passing GNN task. Similar to GNN module, Junction tree encoding module makes tree information message passing network. To create a clique that constitutes a Junction Tree, large chemical databases were used to perform tree decomposition on chemical compounds. And applying tree decomposition, chemical compounds are broken down into ring-based substructures to create cliques. Based on this clique, the chemical compound is made into a junction tree. The generated junction tree can be obtained through the gated recurrent unit based message passing network(Jin *et al.*, 2018). In this work, the model was trained using the ChEMBL database(Gaulton *et al.*, 2012), Moses database(Polykovskiy *et al.*, 2020), FDA approved drug database, and among the information extracted from the model, only tree vector of chemical compound is used to my framework.

3.2.4 Continuous and Data-Driven Descriptors Variational Autoencoder

Finally, I use the Continuous and Data-Driven Descriptors(CDDD) Variational Autoencoder model to encode information about the distribution of chemical compounds. Variational Autoencoder is a method for learning parameters that define the distribution of training data, that can be used to estimate the distribution of chemical compounds(Winter *et al.*, 2019). In this work, I vectorize the distribution information of chemical compound using the corresponding method to use it as input feature.

3.3 Model Architecture

In this section, I introduce novel architecture that is mainly composed of deep autoencoder module.The deep autoencoder is a deep learning architecture that can efficiently perform dimension reduction, creating a reduced latent representation from input data. I construct a cascade autoencoder to create a three-step latent representation, and used diverse information from one chemical compound sequentially with different inputs for each module. that is applied with principal component analysis to create an embedded space.

3.3.1 Autoencoder Module

As show in Figure 3.4, I apply Multi Layer Perceptron (MLP) based stacked autoencoder. Stacked Autoencoder has a Deep Belief Network structure and trains greedy layer-wise.(Bengio *et al.*, 2007). I designed both encoder and decoder based on MLP with a two layers. For a one-layer MLP, the latent

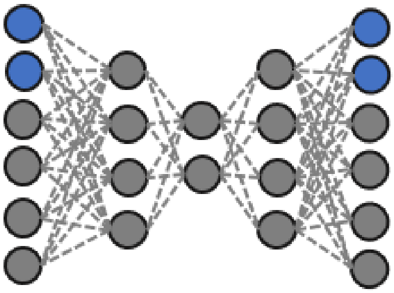

Figure 3.4: Architecture of Autoencoder Module

representation matrix $Z \in \mathrm{R}^{n \times d_z}$ is computed as

$$Z = f(X) = \sigma(\sigma(XW_{e1} + B_{e1})W_{e2} + B_{e2}) \tag{3.1}$$

where $W_{e1}, W_{e2} \in \mathbb{R}^{m \times d}$ are weight matrices, $B_{e1}, B_{e2} \in \mathbb{R}^{m \times d}$ are bias matrices, m is the size of mini batch, d_z is the size of the latent representation. σ is an activation function. In decoder, the latent representation is transformed to reconstructed input vector. Reconstructed input vector $\widetilde{x_i}$ is computed as

$$\widetilde{X} = g(f(X)) = \sigma(\sigma(ZW_{d1} + b_{d1})W_{d2} + b_{d2}) \tag{3.2}$$

$W_{e1}, W_{e2} \in \mathbb{R}^{m \times d}$ are weight matrices, $B_{e1}, B_{e2} \in \mathbb{R}^{m \times d}$ are bias matrices. And I use autoencoder with tied weights model to avoid overfitting problem. An autoencoder with tied weights has decoder weights that are the transpose of the encoder weights. This is a form of parameter sharing, which reduces the number of parameters of the model. Advantages of tying weights are that increases training speed and reduces risk of overfitting. And it can yield comparable performance than without weight tying in many cases(Li and Nguyen, 2018). Therefore, tied weight relationship is expressed as follows.

$$W_{d1} = W_{e2}^T, W_{d2} = W_{e1}^T \tag{3.3}$$

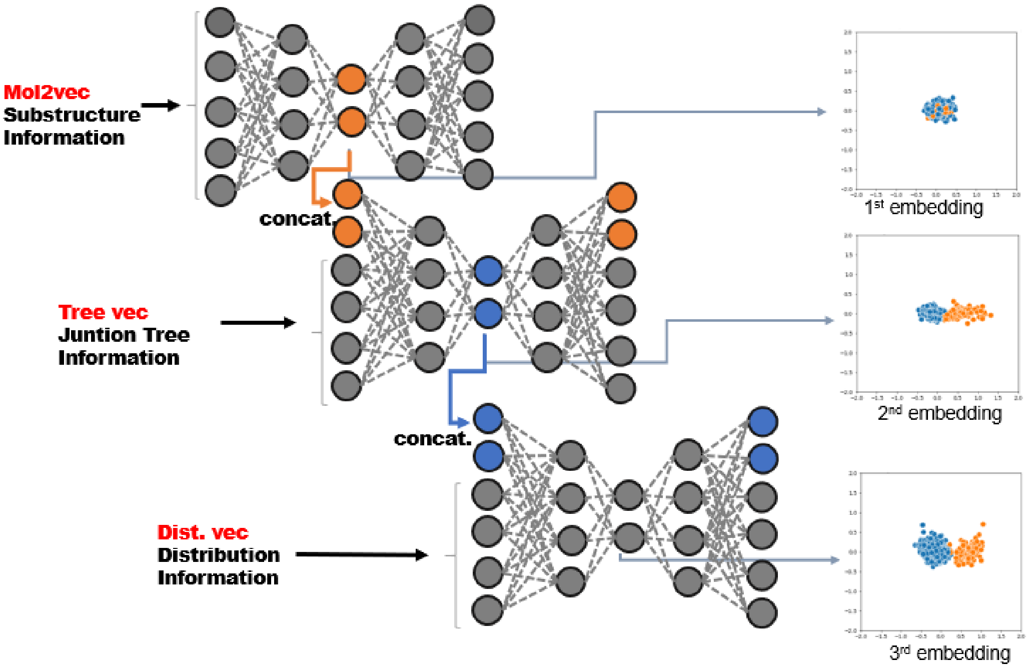

Figure 3.5: Model Architecture

3.3.2 Cascade Autoencoder

As show in Figure 3.5, Cascade autoencoder consists of three similar deep stacked autoencoder with tied weights module. Following methods were used to implement Cascade model. The latent representation of the previous module was concatenated to the input representation of the next module. It consists of three stages of modules, with late presentation operations at stage 2 and stage 3 as follows.

$$Z^{(2)} = f(X^{(2)}||Z^{(1)})) = \sigma((\sigma((X^{(2)}||Z^{(1)})W_{e1}^{(2)} + B_{e1}^{(2)})W_{e2}^{(2)} + B_{e2}^{(2)}))$$

$$\mathrm{Z}^{(3)} = f(X^{(3)}||Z^{(2)})) = \sigma((\sigma((X^{(3)}||Z^{(2)})W_{e1}^{(3)} + B_{e1}^{(3)})W_{e2}^{(3)} + B_{e2}^{(3)}))(3.4)$$

When designing architecture in this way, diverse inputs can be put into the model, and the information loss can be reduced by concatenating compressed information from the previous module. For better performance, I use batch normalization and dropout techniques. Both methods also have the effect of reducing risk of overfitting. And I use nn.Sigmoid() activation function for matching scale of the latent vector and scale of the input data for the following modules.

3.4 Loss function, Optimizer

When designing deep learning models, the most important thing is to define objective functions, and diverse optimization schemes can be applied when updating parameters to minimize them. To train my model properly, I use anadequate loss functions and optimizers.

3.4.1 Reconstruction Loss

In general, autoencoder calculates the reconstruction loss. I have to decide that loss function to use according to the input representation, and I used mean square error because I encode input with a minmax scale from 0 to 1 continuous value. Mean Square Error (MSE) is the square of the error with the prediction value for each input x_i. MSE loss is computed as

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x_i})^2 \tag{3.5}$$

3.4.2 Metric Loss

Metric learning is a method of measuring distance between data points. I learn metrics that make class label-specific for data that are not easily classified with traditional features. To this end, I define a distance function in the embedding space, using a metric called center loss. Center loss is a classification metric developed in the field of face recognition that trains data in an embedded space in a discretionary manner(Wen *et al.*, 2016). A class center is set for each class in the Mini batch, and samples belonging to the class are placed close to the class center. Center loss can be used as a clustering loss.

$$L_C = \frac{1}{2}\sum_{i=1}^{N} \|v_i - c_{y_i}\|_2^2 \tag{3.6}$$

3.4.3 Optimizer

In this work, I use different optimizers for reconstruction loss and metric loss. First, I use a stochastic gradient description (SGD) to the metric loss(Kiefer *et al.*, 1952). SGD performs gradient descent in mini batches. In the case of Center loss, SGD is a suitable optimizer because it trains the label away from the mini-batch. Secondly, Adam optimizer was applied for reconstruction loss. Adam optimizer is a method of updating parameters to different sizes by applying moment and decaying average of gradients(Kingma and Ba, 2014). It has the advantage of much faster convergence than SGD. And I apply L2 norm regularizer to optimizers as above for reducing overfitting problem.

3.5 Principal Component Analysis

Principal Component Analysis is a traditional machine learning technique for dimension reduction. A high-dimensional vector is represented by a linear combination of a low-dimensional vector. Typically, the number of principal components is set to 2, 3 for representation in 2D or 3D space. In this work, I used 3D PCA to draw embedding spaces in three-dimensional space, that is the output of each encoder module in cascade autoencoder.

3.6 Machine Learning Classifiers

I performed traditional machine learning classification tasks using 3^{rd} latent representationof my framework. I introduce the machine learning classifier model used below.

3.6.1 Support Vector Machine

Support vector machine is one of machine learning model, a supervised learning model for pattern recognition, data analysis, and are mainly used for classification and regression(Cortes and Vapnik, 1995). Given a set of data belonging to either category, the SVM algorithm builds a non-probabilistic binary linear classification model that determines which category the new data belongs to based on the given dataset. If the data are not classified by linear function, I solve it by designing an SVM structure that defines the appropriate kernel function. In this work, to avoid aforementioned problem, I use Radius Basis Function (RBF) kernel.

3.6.2 Naive Bayes

In the field of machine learning, the Naïve Bayes Classification is a type of probability classifier that applies the Bayes Theorem, which assumes independence between features. The advantages of Naive Bayes are as follows. First, in some probability models, the Naive Bayes classification can be trained very efficiently in supervised learning environments. Second, the amount of training data for estimating parameters required for classification is very small. Third, despite its simple design and simple assumptions, the Naive Bayes classification works well in many complex real-world situations.

3.6.3 Random Forset

Random Forest is an algorithm for improving the shortcomings of decision trees in a method first introduced by Leo Breiman in 2001, that combines multiple decision trees into a single model . Random forest is a method of adding randomness to the sample variable of each bootstrap in the bagging model. This allows us to have more diverse hyperplanes than conventional bagging models and maximize the advantages of ensembel models, improving predictive classification and accuracy over conventional methods. In the case of a decision tree, it is a very unstable model, that combines these trees to achieve a final conclusion by voting the results of several decision trees. I used random forest classifier to conduct model evaluation through classification accuracy and so on.

3.6.4 Adaboost

Adaboost is a machine learning meta-algorithm developed by Yoav Freund and Robert Schapire. Adaboost is ensemble method that it's core concept is to train different weak classifiers on the same dataset and combine weak classifiers to form strong classifiers. The strong classifier, created by combining the weak classifier, is computed as

$$f(x) = \sum_{i=1}^{T} \alpha_t h_t(x) \tag{3.7}$$

where α is In this work, I use decision tree as a weak classifier.

Chapter 4

Experiments

4.1 Datasets

4.1.1 Datasets for pre-trained model

In order to pretrain chemical compound encoding methods, I use several large chemical compound database. The most commonly used databases in Chem-informatics are the ZINC Clean Lead database(Irwin and Shoichet, 2005) and the ChEMBL database(Gaulton *et al.*, 2012). The original ZINC database has 4,591,276 molecules in total but I used MOSES dataset, that filtered ZINC by several criteria(Polykovskiy *et al.*, 2020). In ChEMBL database, I filtered dataset by molecules with molecular weight less than 500. I use these two large chemical databases to pre-train Mol2vec, Junction Tree Variational Autoencoder, and Variational Autoencoder. Following table 4.1 shows the summary of these databases.

Table 4.1: Statistic of Large Chemical Databases

Database	# of unique chemical compound
ChEMBL	1,961,462
ZINC	4,591,276
Moses	1,936,962

Table 4.2: Statistic of FDA approval dataset

Dataset	Status	# of unique chemical compound
FDA Approved small molecules	Approved	1447
	Discontinued	338
Filtered dataset	Approved	904
	Discontinued	198

4.1.2 FDA Approved and Discontinued dataset

To create a chemical embedding space of FDA-approved drugs, I used FDA-approved, discontinued data among diverse drug dataset(Douguet, 2018)(Siramshetty *et al.*, 2016). In the work, I filtered data by small molecule. The dataset consists of SMILES and ATC code information for FDA approached, discontinued small molecule, 1447 approached data and 338 discontinued data. Before training my model, I constructed 904 FDA approached and 198 discontinued data by filtering only the drug with one ATC code. Table 4.2 shows the summary of this dataset.

4.2 Model Training Hyper Parameter Settings

I describe model parameters and hyperparameter setting that I set to effectively train my model.

4.2.1 The dimension of input data

I generated input in 3 ways from the SMILES string of Chemical compound. Each method is pre-trained using moses dataset and ChEMBL dataset. The dimension of Morgan Fingerprint is set to 512, the dimension of Mol2vec is set to 300, the dimension of tree vector from Junction tree VAE is set to 400, the dimension of Continuous data driven VAE vector is set to 512. The input data of 1^{st} module is 812 dimension, concatenating Mol2vec vector and Morgan Fingerprint. The input data of the 2^{nd} module is 912 dimensions, concatenating tree vector and Morgan Fingerprint.The input data of the 3^{rd} module is 1024 dimensions, concatenating distribution vector and Morgan Fingerprint.

4.2.2 Model Training

I design autoencoder module as tied autoencoder to equalize the weight of the hidden layer of the encoder and the hidden layer of the decoder. To train a model with optimal results, I set the hyper parameters as follows: The dimension of 1^{st} hidden layer in Encoder is set to 256, the dimension of latent layer is set to 8 and the dimension of 1^{st} hidden layer in Decoder is set to 256. Dropout probability is set to 0.4 and batch size is set to 32. I use Adam Optimizer for reconstruction loss and set initial learning rate and weight decay to 10^{-5} and 10^{-6}. I use Stochastic Gradient Descent optimizer for center loss and set initial learning rate and weight decay to 10^{-5} and 10^{-6}. And the ratio of reconstruction loss to center loss, λ, is set to 0.01. The size rate of training set, validation set and test set is split into 8:1:1.

4.2.3 Embedding and Evaluation method

I extract the latent presentation of chemical compound using a model trained by the above method. When I use Principal Component Analysis to embed chemical space, the number of principal components is set to 3. And for classifying latent representation, I use Support Vector machine, Naive Bayes, Random Forest, Adaboost classifier. I use accuracy, balanced accuracy, average precision-recall and f1 score to evaluate model.

4.2.4 Comparison Models

To evaluate the performance of my framework, I compare my framework's performance with other models. I compare my framework with traditional machine learning methods and with single autoencoder to show the effectiveness of the cascade model.

- **Traditional Machine Learning Method**: I perform machine learning classification tasks using only the Morgan Fingerprint created by SMILES string of chemical compound as input. Morgan Fingerprint of 512 dimension vectors and FDA approved/discontinued labels are used to perform SVM, NB, RF, Adaboost classification task.

- **Information vector**: To evaluate whether pre-trained models I used effectively encodes information, I perform a machine learning classification tasks with only the information vectors extracted from pre-trained models using SMILES string of chemical compound. I compare performance when using only Mol2vec, tree vector, distribution vector and concatenated vector.

- **Single Autoencoder**: I compare the results with when I use only single autoencoder to show that Cascade modeling is useful. In this case, I perform machine learning classification tasks using a latent vector extracted when each encoding vector used as input for the single autoencoder.

Chapter 5

Results

In this chapter, I report the results of experiments. First, I visualize the chemical embedding space obtained through my framework, and compare the performance with the traditional method. And I report results one by one on the benefits of my framework.

5.1 Visualization of Chemical Embedding Space

As shown in Figure 5.1, I visualize the chemical embedding space using latent representation of each autoencoder module. I plot the chemical embedding space with latent representation from each module using Principal Component Analysis, (a), (c) and (e) are the results for the training data, (b), (d) and (f) are the results for the test data. As shown in Figure 5.1(a), (e), the embedding space is more discriminative in 3^{rd} module than in 1^{st} module. This is due to multiple information I used sequentially as input to my framework. More information is encoded into latent vectors in the third module. And I also obtain more discriminative embedding space from 3rd module due to calculate

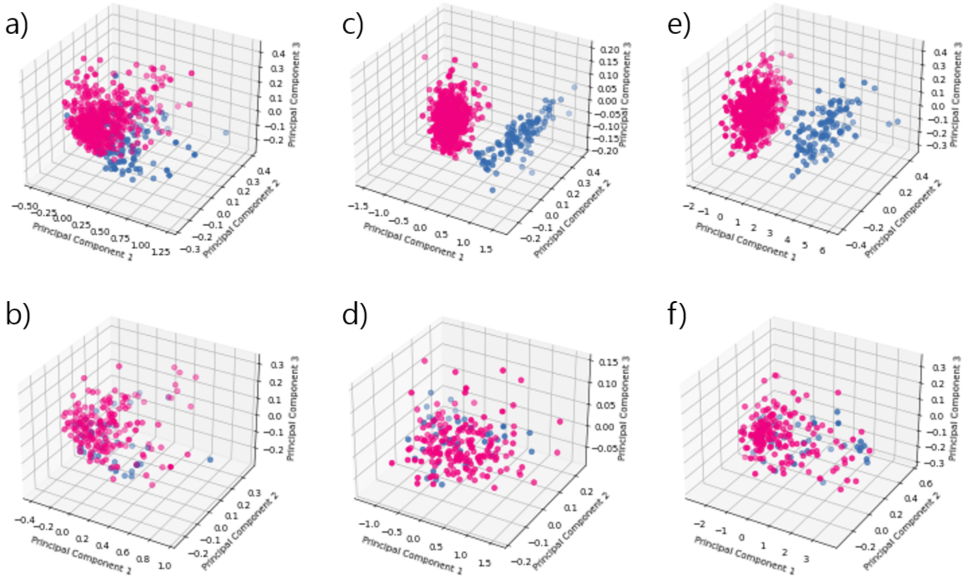

Figure 5.1: Visualization of chemical embedding space. (a) Embedding space is created by latent representation of training data of the first module. (b) use latent representation of test data of the first module. (c) Use latent representation of training data of the second module. (d) Use latent representation of test data of the second module. (e) Use latent representation of training data of the third module. (d) Use latent representation of test data of the third module.

metric loss repeatedly. When I use the test data, I find that there was an overfitting issue. However, as shown in Figure 5.1(f), it is encouraging that I identified a little discriminative regions.

5.2 Performance Comparisons with Traditional Machine Learning Method

I compare performance of my framework with the result of using traditional method. The traditional method is the result of machine learning classification tasks using Morgan Fingerprint representation vector as input. In my

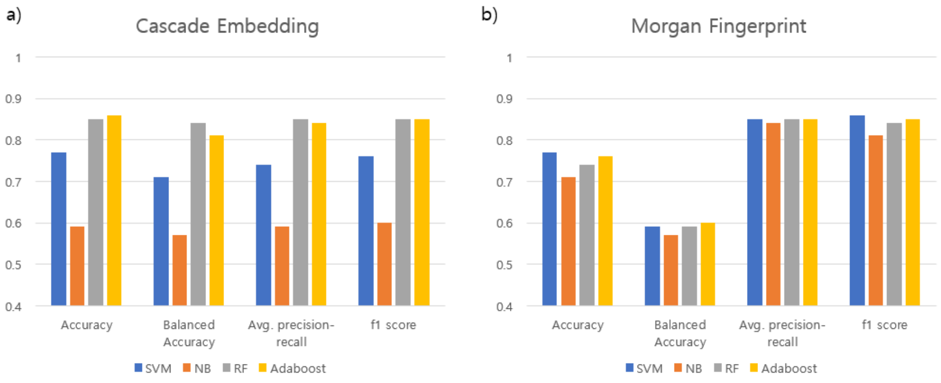

Figure 5.2: Performance Comparisons with Traditional Machine Learning Method. (a)Performance of my framework. (b)Performance of traditional method. I use 512 dimensional Morgan Fingerprint as representation.

framework, machine learning classification tasks were performed using the latent representation from the 3^{rd} module. As shown in Figure 5.2, the results obtained using my framework were outperformed in most machine learning classifier models than traditional method.

5.3 Performance of using each input representation

To see how useful the information vectors I used in my framework are, I perform machine learning classification tasks using only each information vector as input. As shown in Figure 5.3, I performed machine learning tasks using each input vector individually 5.3 (a), (b), (c). Compared to the results using Morgan Fingerprint, there was no significant improvement. It means that each input representation can't encode a common feature of FDA approved drugs. This is because each encoding method encodes only certain information from the chemical compound. So I integrated and combined the diverse information of chemical compound.

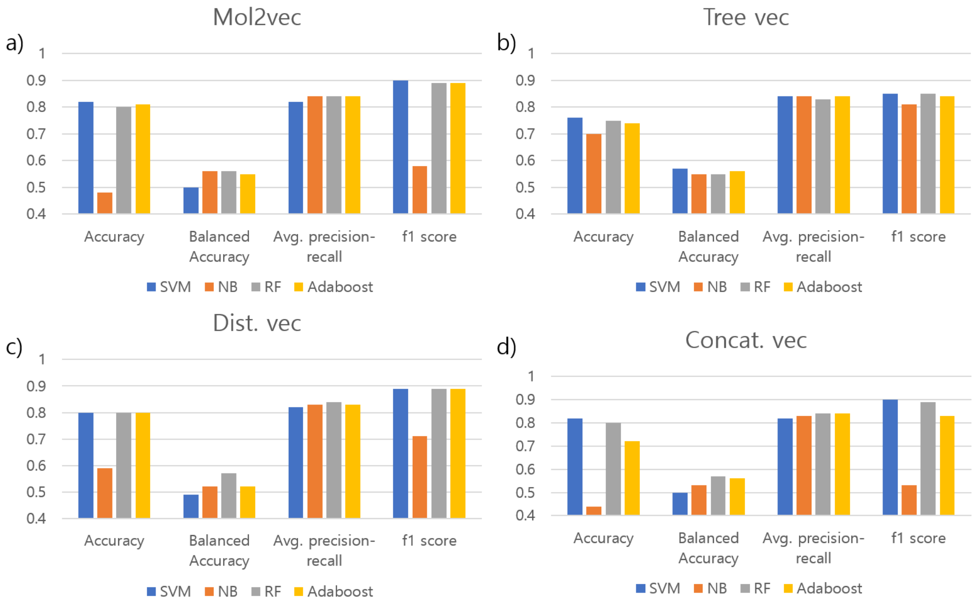

Figure 5.3: Performance using Input Representation only. Machine learning classification tasks were performed by (a)only using Mol2vec, (b)only using tree vector, (c)only using distribution vector, (d)only using concatenated vector

5.4 Effect of Cascade Modeling

I conduct additional experiment on performance when using only a single autoencoder. I report experimental results and find benefits of designing my framework as cascade model. As shown in Figure 5.4, I performed machine learning tasks using latent vector from single autoencoder that used each input representation individually. Compared to the result in section 5.3, single autoencoder improved about 10% in terms of balanced accuracy. Classification performance is improved by using only one autoencoder. Furthermore, as shown in Figure 5.2 (a), cascade model shows a 20% improvement in performance compared to using only a single autoencoder. I interpret these results in terms of ensemble method of chemical representation.

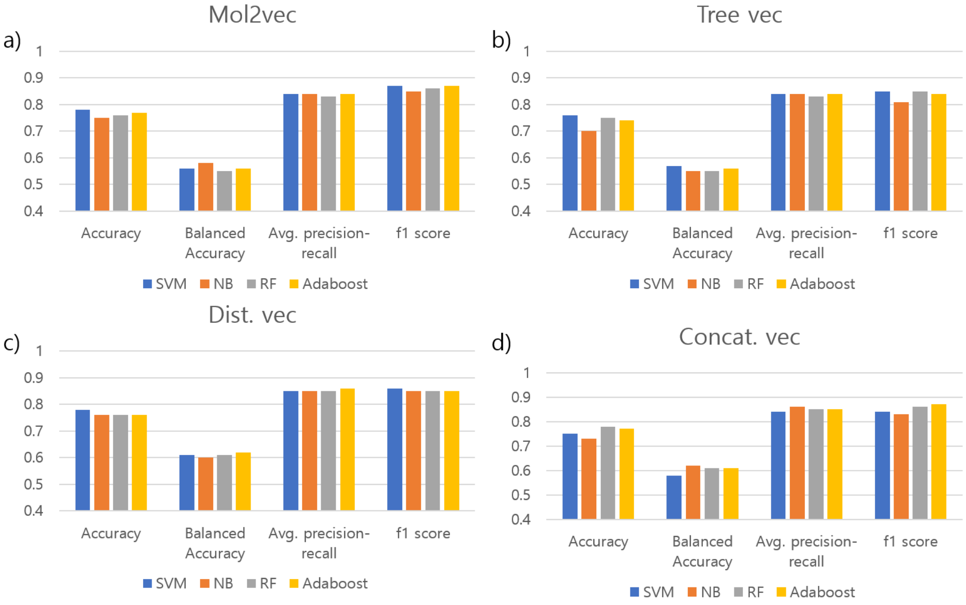

Figure 5.4: Performance using single autoencoder. Machine learning classification tasks were performed by latent representation of single autoencoder that is extracted from (a)only using Mol2vec, (b)only using tree vector, (c)only using distribution vector, (d)only using concatenated vector

Chapter 6

Conclusion

In this section, I will sum up the works in this paper and set future works for further improvement.

1. I built a framework that integrates encoded chemical representation in different ways. This effectively integrated diverse information of chemical compound. Compared to traditional method, my framework outperformed.

2. I visualize the chemical embedding space of FDA approved chemical compound and discontinued chemical compound.

3. Although classification power is obtained by effectively learning the representation of chemical compounds, I seek to solve the overfitting problem by using different encoding methods to improve discriminative power in embedding spaces, or by using information-specific deep learning architectures rather than MLP-based deep autoencoders.

4. To increase the size of the dataset, I would like to apply transfer learning

techniques. I would also like to develop an embedding method that can be applied to multi-label classification task.

Bibliography

Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H., *et al.* (2007). Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, **19**, 153.

Choubisa, H., Askerka, M., Ryczko, K., Voznyy, O., Mills, K., Tamblyn, I., and Sargent, E. H. (2020). Crystal site feature embedding enables exploration of large chemical spaces. *Matter*, **3**(2), 433–448.

Cortes, C. and Vapnik, V. (1995). Support vector machine. *Machine learning*, **20**(3), 273–297.

Douguet, D. (2018). Data sets representative of the structures and experimental properties of fda-approved drugs. *ACS medicinal chemistry letters*, **9**(3), 204–209.

Gaulton, A., Bellis, L. J., Bento, A. P., Chambers, J., Davies, M., Hersey, A., Light, Y., McGlinchey, S., Michalovich, D., Al-Lazikani, B., *et al.* (2012). Chembl: a large-scale bioactivity database for drug discovery. *Nucleic acids research*, **40**(D1), D1100–D1107.

Guo, R., Wang, W., and Qi, H. (2015). Hyperspectral image unmixing using autoencoder cascade. In *2015 7th Workshop on Hyperspectral Image and*

Signal Processing: Evolution in Remote Sensing (WHISPERS), pages 1–4. IEEE.

Irwin, J. J. and Shoichet, B. K. (2005). Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, **45**(1), 177–182.

Jaeger, S., Fulle, S., and Turk, S. (2018). Mol2vec: unsupervised machine learning approach with chemical intuition. *Journal of chemical information and modeling*, **58**(1), 27–35.

Jin, W., Barzilay, R., and Jaakkola, T. (2018). Junction tree variational autoencoder for molecular graph generation. In *International Conference on Machine Learning*, pages 2323–2332. PMLR.

Kiefer, J., Wolfowitz, J., *et al.* (1952). Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, **23**(3), 462–466.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koge, D., Ono, N., Huang, M., Altaf-Ul-Amin, M., and Kanaya, S. (2021). Embedding of molecular structure using molecular hypergraph variational autoencoder with metric learning. *Molecular informatics*, **40**(2), 2000203.

Landrum, G. *et al.* (2006). Rdkit: Open-source cheminformatics.

Law, A. and Ghosh, A. (2019). Multi-label classification using a cascade of stacked autoencoder and extreme learning machines. *Neurocomputing*, **358**, 222–234.

Lei, H. and Yang, Y. (2020). Cdae: A cascade of denoising autoencoders for noise reduction in the clustering of single-particle cryo-em images. *Frontiers in genetics*, **11**.

Li, N., Chang, F., and Liu, C. (2020). Spatial-temporal cascade autoencoder for video anomaly detection in crowded scenes. *IEEE Transactions on Multimedia*, **23**, 203–215.

Li, P. and Nguyen, P.-M. (2018). On random deep weight-tied autoencoders: Exact asymptotic analysis, phase transitions, and implications to training. In *International Conference on Learning Representations*.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Morgan, H. L. (1965). The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of Chemical Documentation*, **5**(2), 107–113.

Mullard, A. (2021). 2020 fda drug approvals. *Nature reviews drug discovery*.

Oprea, T. I. and Gottfries, J. (2001). Chemography: the art of navigating in chemical space. *Journal of combinatorial chemistry*, **3**(2), 157–166.

Polykovskiy, D., Zhebrak, A., Sanchez-Lengeling, B., Golovanov, S., Tatanov, O., Belyaev, S., Kurbanov, R., Artamonov, A., Aladinskiy, V., Veselov, M., Kadurin, A., Johansson, S., Chen, H., Nikolenko, S., Aspuru-Guzik, A., and Zhavoronkov, A. (2020). Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models. *Frontiers in Pharmacology*.

Rogers, D. and Hahn, M. (2010). Extended-connectivity fingerprints. *Journal of chemical information and modeling*, **50**(5), 742–754.

Rosén, J., Gottfries, J., Muresan, S., Backlund, A., and Oprea, T. I. (2009). Novel chemical space exploration via natural products. *Journal of medicinal chemistry*, **52**(7), 1953–1962.

Siramshetty, V. B., Nickel, J., Omieczynski, C., Gohlke, B.-O., Drwal, M. N., and Preissner, R. (2016). Withdrawn—a resource for withdrawn and discontinued drugs. *Nucleic acids research*, **44**(D1), D1080–D1086.

Wen, Y., Zhang, K., Li, Z., and Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer.

Winter, R., Montanari, F., Noé, F., and Clevert, D.-A. (2019). Learning continuous and data-driven molecular descriptors by translating equivalent chemical representations. *Chemical science*, **10**(6), 1692–1701.

Xia, J., Jin, H., Liu, Z., Zhang, L., and Wang, X. S. (2014). An unbiased method to build benchmarking sets for ligand-based virtual screening and its application to gpcrs. *Journal of chemical information and modeling*, **54**(5), 1433–1450.

Yu, K., Visweswaran, S., and Batmanghelich, K. (2020). Semi-supervised hierarchical drug embedding in hyperbolic space. *Journal of Chemical Information and Modeling*.

국문초록

신약 개발시 여러 조건들을 충족하는 약물을 발견하기 위해 수십년의 노력이 필요하다. 컴퓨터 보조 신약 개발(CADD)은 컴퓨터 과학을 적용시켜 약물로 사용되는 약물의 구조적 및 물리적 특성을 확인하고 유사한 특성을 가진 신약 후보를 도출함으로써 신약 개발의 시간과 비용을 체계적으로 절감하는 것을 목표로 하는 신흥 연구 분야이다. 특히 미국 식품의약국(FDA)이 승인한 약물의 특성을 확인하는 것이 가장 중요하다. FDA에서 승인한 약물들은 독성, 효능 및 부작용 측면에서 검증된 의약품이다. 이러한 약물들이 임베딩 공간 상에서 어떻게 분포되어 있는지에 대한 의문점에서 시작한다. 전통적으로는 전문가의 수작업으로 만든 규칙들로 화합물의 임베딩 공간을 구성했다. 전통적인 화합물 표현만으로는 FDA 승인 약물들을 분류하는 것이 어렵다. 빅데이터와 인공지능 기술의 발전으로 딥러닝을 이용해 임베딩 공간을 구축한다. 그러나 기존 연구들에선 FDA 승인 약물들의 임베딩 공간을 식별할 수 있는 적절한 방법이 없다.

본 연구에서는 FDA 승인 약물들의 특징을 인코딩하는 프레임워크를 사용해 차별적인 임베딩 공간을 구축하는 방법을 제안한다. 제안된 프레임워크는 3개의 순차적 딥 오토인코더 모듈로 구성된다. 제안된 프레임워크는 순차적 모델링을 통해 약물의 정보를 효과적으로 통합한다. 순차적으로 연결된 3개의 오토인코더 모듈을 사용하여 이전 모듈에서 학습한 잠재 표현을 지속적으로 사용한다. FDA 승인 화학 화합물이 임베딩 공간상에서 차별적인 영역을 가지고 있는지 여부는 제안된 프레임워크에 의해 시각화된다. 또한 잠재된 표현이 FDA 승인 정보를 효과적으로 특성화하는지 여부를 평가하기 위해 기계 학습 분류 작업을 수행한다. 궁극적으로, 본 논문에서 제안하는 프레임워크는 신약 후보자의 승인 여부를 결정하기 위한 임베딩 방법으로 사용될 수 있다.

주요어: FDA 승인 약물, 순차적 오토인코더, 화합물 공간 임베딩

학번: 2019-24822

감사의 글

2년이라는 시간 동안 화목한 연구실에서 석사 학위를 마칠 수 있었음에 정말 행복했습니다. 먼저 아낌없는 지도와 격려로 저를 이끌어 주신 김선 교수님께 감사드립니다. 학업적 지도 뿐만 아니라 사회 생활을 함에 있어 제가 앞으로 나아가야 할 방향의 귀감이 되어주신 김선 교수님께정말 감사드립니다. 그리고 아낌없이 조언해주신 장병탁 교수님, 황승원 교수님께도 감사드립니다.

다음으로 생물정보 및 생명정보 연구실 동료들에게 감사의 말을 전합니다. 특히 항상 아낌없이 조언해주신 임상수 박사님, 이상선 박사님께 감사드립니다. 그리고 여러 방면으로 도움을 준 연구실 구성원 여러분들께 감사합니다.

그리고 친구들에게 감사의 말을 전합니다. 컴퓨터 공부를 도와준 부루, 연빈에게 감사합니다. 자취 전 신세를 많이 졌던 박성민에게도 감사합니다. 그리고 찬우, 석호, 준호, 동녘, 민재, 한성, 지은 등에게 감사합니다. 항상 모든 고민을 공유하고 들어준 임성민에게도 감사합니다.

마지막으로 사랑하는 우리 가족에게 감사의 말을 전합니다. 항상 제가 하고 싶은 일을 응원해주시고, 지지해주시는 아버지, 이제 새신랑이 될 김동우 박사님, 그리고 새로운 길로 나아가는걸 누구보다도 기뻐하셨을 어머님께 정말 사랑한다고 전하고 싶습니다. 그리고 언제나 나의 편인 소영이에게 감사합니다.