

**QUASI-SUPERVISED STRATEGIES FOR  
COMPOUND-PROTEIN INTERACTION  
PREDICTION**

**A Thesis Submitted to  
the Graduate School of Engineering and Sciences of  
İzmir Institute of Technology  
in Partial Fulfillment of the Requirements for the Degree of**

**MASTER OF SCIENCE**

**in Electronics and Communications Engineering**

**by  
Onur ÇAKI**

**July 2021  
İZMİR**

## **ACKNOWLEDGEMENT**

First and foremost, I would like to express my gratitude to my advisor Prof. Dr. Bilge Karaçalı, not only for his excellent guidance but also for his continued trust and patience through my master's study.

I also appreciate Assoc. Prof. Serhat Tozburun and Dr. Ibrahim Akkaya, who are my former lab lead and member, for their support and encouragement through this journey.

I would like to thank my thesis committee members, Assist. Prof. Dr. Ezgi Karaca and Assist. Prof. Dr. Mehmet Serkan Apaydın, for their valuable contributions to my thesis.

Finally, I would like to express my warmest thanks to my dear mother, Kadriye Çakı, my dear father, Kenan Çakı, and to my aunts and uncles for their endless support throughout my whole life.

# ABSTRACT

## QUASI-SUPERVISED STRATEGIES FOR COMPOUND-PROTEIN INTERACTION PREDICTION

In-silico prediction of compound-protein interaction using computational methods preserves its importance in various pharmacology applications because the wet-lab experiments are time-consuming, laborious and costly. Most machine learning methods proposed to that end approach this problem with supervised learning strategies in which known interactions are labeled as positive and the rest are labeled as negative. However, treating all unknown interactions as negative instances may lead to inaccuracies in real practice since some of the unknown interactions are bound to be positive interactions waiting to be identified as such. In this study, we propose to address this problem using the Quasi-Supervised Learning algorithm. In this framework, potential interactions are predicted by estimating the overlap between two datasets: a true positive dataset which consists of compound-protein pairs with known interactions and an unknown dataset which consists of all the remaining compound-protein pairs. The potential interactions are then identified as those in the unknown dataset that overlap with the interacting pairs in the true positive dataset in terms of the associated similarity structure between interacting pairs. Experimental results on GPCR and Nuclear Receptor datasets show that the proposed method can identify actual interactions from all possible combinations.

# ÖZET

## BİLEŞİK-PROTEİN ETKİLEŞİMİ TAHMİNİ İÇİN YARI-GÜDÜMLÜ YAKLAŞIMLAR

Laboratuvar ortamında gerçekleştirilen bileşik-protein etkileşimi belirleme deneylerinin zaman alıcı, zahmetli ve maliyetli olması nedeniyle, hesaplamalı yöntemler kullanarak dijital ortamda bileşik-protein etkileşimi tahmini önemini korumaktadır. Bu amaçla geliştirilen pek çok yapay öğrenme yöntemi bu probleme bilinen etkileşimlerin pozitif, eldeki geri kalan bütün etkileşimlerin ise negatif olarak etiketlendiği güdümlü öğrenme stratejileri ile yaklaşmıştır. Fakat bilinmeyen etkileşimler açığa çıkarılmayı bekleyen pozitif etkileşimleri de barındıracağından, bilinmeyen bütün etkileşimleri negatif örnek olarak ele almak gerçek uygulamalarda hatalı sonuçlara yol açacaktır. Bu çalışmada, bu problemin Yarı-Güdümlü Öğrenme Algoritması ile çözülmesi amaçlanmaktadır. Bu çerçevede olası etkileşimler iki veri kümesinin örtüşümü kestirilerek tahmin edilir: Etkileştikleri bilinen bileşik-protein çiftlerinden oluşan gerçek pozitif veri kümesi ve geri kalan diğer bütün bileşik-protein çiftlerinden oluşan bilinmeyen veri kümesi. Gerçek pozitif veri kümesindeki etkileşen çiftlerle ilgili yapısal benzerlik açısından örtüşen bilinmeyen veri kümesindeki bileşik-protein çiftleri potansiyel etkileşimler olarak tanımlanır. GPCR ve Nuclear Receptor veri kümeleri üzerindeki deneysel sonuçlar, amaçlanan yöntemin bütün olası çiftlerden gerçek etkileşimleri saptayabildiğini göstermektedir.

# TABLE OF CONTENTS

LIST OF FIGURES .....	vii
LIST OF TABLES .....	viii
CHAPTER 1. INTRODUCTION .....	1
1.1. Computational Approaches for CPI prediction .....	2
1.2. Problem Definition .....	3
1.3. Thesis Roadmap .....	4
CHAPTER 2. BACKGROUND .....	6
2.1. Chemical Space .....	6
2.2. Genomic Space .....	12
2.3. Feature-based vs Similarity-based Methods .....	13
2.4. Literature Review of Similarity-Based Methods .....	13
CHAPTER 3. MATERIALS AND METHODS .....	17
3.1. Dataset .....	17
3.2. Similarity Measurements .....	18
3.2.1. Compound-Compound Similarity Measurements .....	18
3.2.1.1. Graph-based methods .....	18
3.2.1.2. SMILES-based methods .....	23
3.2.1.3. Molecular Fingerprints-based methods .....	29
3.2.2. Protein-Protein Similarity Measurements.....	34
3.2.3. Pairwise Kernel Method .....	35
3.3. Quasi-Supervised Learning Algorithm.....	36
3.3.1. Efficient Numerical Computation of the Posterior Probability Estimations.....	38
3.4. Kolmogorov-Smirnov method .....	41
CHAPTER 4. RESULTS .....	43

CHAPTER 5. CONCLUSION .....	52
REFERENCES .....	54

# LIST OF FIGURES

<b><u>Figure</u></b>	<b><u>Page</u></b>
Figure 1. Schematic diagram of the proposed method .....	5
Figure 2. Molfile of Isotretinoin .....	8
Figure 3. SMILES code of Atropine sulfate with its 2D graph representation .....	11
Figure 4. Supervised learning framework of CPI prediction.....	15
Figure 5. Train and test splitting scheme to evaluate three different cases .....	15
Figure 6. Fundamental terms in the graph theory .....	19
Figure 7. Calculation of the association graph .....	20
Figure 8. 451. bit of the ECFP4 fingerprints of Epinephrine and Levodopa.....	30
Figure 9. The highlighted substructures that correspond to the ring attribute C8x-C8x-C8y-C8x-C8y-C8y,1-6 (A) and the vicinity attribute C1b-C1c-C8y,2-O1a (B) in KCF-S fingerprints of Epinephrine .....	33
Figure 10. The highlighted substructures that correspond to the ring attribute C8x-C8x-C8y-C8x-C8y-C8y,1-6 (A) and the vicinity attribute C1b-C1c-C6a,2-N1a (B) in KCF-S fingerprints of Levodopa.....	33
Figure 11. The alignment of homeodomain region of homeobox genes from Mouse and Human using Needleman Wunsch (global alignment) (A) and Smith-Waterman (local alignment) (B) [46] .....	35
Figure 12. Posterior probability distributions of compound-protein pairs in the Nuclear Receptor dataset to belong to the true positive dataset and their Kolmogorov-Smirnov Analysis for (a) KCF-S Fingerprints, (b) Maximum Common Substructure (RDkit), and (c) LINGO based TF-IDF Cosine Similarity .....	46
Figure 13. The plot of the cost functions, E(n) for (a) KCF-S Fingerprints, (b) Maximum Common Substructure (RDkit), and (c) LINGO based TF-IDF Cosine Similarity .....	47
Figure 14. Posterior probability distributions of compound-protein pairs in the GPCR dataset to belong to the true positive dataset and their Kolmogorov-Smirnov Analysis for (a) LINGO based TF-IDF Cosine Similarity, (b) LINGOsim (q=3), and (c) LINGOsim (q=4) .....	48
Figure 15. The plot of the cost functions, E(n) for (a) LINGO based TF-IDF Cosine Similarity, (b) LINGOsim (q=3), and (c) LINGOsim (q=4) .....	49

## LIST OF TABLES

<b><u>Table</u></b>	<b><u>Page</u></b>
Table 1. Different representations of Isotretinoin.....	7
Table 2. Most frequently used string characters in SMILES [23] .....	10
Table 3. The list of 20 amino acids.....	12
Table 4. Datasets of Yamanishi [28] .....	17
Table 5. The list of 68 KEGG atom types .....	21
Table 6. SMILES representations and 2D structure diagrams of Epinephrine and Levodopa .....	24
Table 7. LINGOs with their corresponding frequencies in the SMILES strings of Epinephrine and Levodopa .....	25
Table 8. A reduced list of substructures that can be extracted from Epinephrine using KCF-S fingerprint attributes .....	31
Table 9. Performance comparison of compound similarity measure methods.....	44
Table 10. The list of top 40 predicted positive interactions in the unknown dataset $C_0$ for the Nuclear Receptor Dataset.....	50
Table 11. The list of top 40 predicted positive interactions in the unknown dataset $C_0$ for the GPCR Dataset .....	51



# CHAPTER 1

## INTRODUCTION

Identification of compound-protein interactions (CPI) plays an essential role in a wide range of pharmacological applications. Most drugs are small chemical compounds that modulate the biological activities of their target proteins by interacting with them. In the initial phases of the drug discovery process, a druggable target protein of interest is identified and validated. Effective interactions between drug candidate compounds and the target protein that induce the desired biological effect are then detected. Thousands of compounds that exhibit sufficient pharmacological potential can move to the next phases of the drug discovery process, such as optimization, preclinical testing, and clinical development [1]. At the end of the drug discovery and development process that takes several years and costs approximately \$2.6 billion, only a few of all candidate drugs achieve approval [2]. Between 2015-2019, 44 drugs per year on average were approved by the FDA, which corresponds to a 0.01% approval rate [3]. Since there are many diseases that remain to be solved, and many new diseases are emerging every day, the drug discovery process needs to be accelerated. There is also a strong incentive to screen drug candidate compounds against many target proteins instead of only one target protein. A number of studies have shown that complex diseases such as cancer and Alzheimer's Disease are associated with multiple targets necessitating the elucidation of the interaction profiles of candidate drugs with many target proteins [4]. Drugs may also inadvertently interact with other off-target proteins. Detection of such interactions allows prediction and analysis of undesired side-effects of drugs [5]. Finally, CPI prediction is a key part of drug repositioning, i.e., discovering new clinical usage of existing drugs. Since these drugs have already passed many time-consuming and costly processes, drug repositioning attracts researchers in the pharmaceutical industry with increasing interest. [6].

Experimental validation of compound-protein interactions in laboratory environments remains time-consuming, laborious, and extremely costly, even when using high-throughput screening technologies. As a result, only a small number of experimentally validated interacting compound-protein pairs exist compared to the large

numbers of compounds and proteins: There are ~568 million protein sequences and ~96 million compounds in the databases of NCBI Entrez system against only ~1.2 million recorded interactions [7]. In recent years, there has been growing interest in using computational tools for CPI prediction, fueled by studies with promising results [8]. In-silico prediction of CPI aims to narrow the search space for future wet-lab experiments by suggesting the most likely interactions, thereby accelerating pharmacological research processes, decreasing costs, and increasing research productivity [8].

### **1.1. Computational Approaches for CPI prediction**

There are three main computational approaches in virtual screening for potential compound-protein interactions. Structure-based approaches aim to utilize the 3D structure of a target protein to determine whether or not a compound would interact with the target protein [9]. The disadvantage of this approach is that obtaining the 3D structure of a target protein may not always be possible, especially for membrane proteins such as Ion Channels and GPCRs [10]. In ligand-based approaches, potential interactions are identified by comparing the structure of compounds that are known to interact with a target protein against candidate compounds. [11]. Based on the universal expectation that shared structural elements may indicate shared functional characteristics, this approach becomes unfeasible if the target protein of interest has few or no known interactions. Finally, chemogenomics approaches address the issues associated with the earlier two approaches [10]. The idea behind this approach is again that the compounds that have similar structure would tend to interact with same or similar proteins, but unlike the ligand-based approaches, information that comes from both compounds and proteins are considered simultaneously. A chemogenomics approach seeks to reveal the correlations between the chemical space and the genomic space by relating the chemical structure information of compounds with amino acid sequence information of proteins. In this way, these approaches aim to compensate for the lack of known interactions of target proteins by considering the known interactions of similar proteins, and to develop a unifying prediction model for the whole compound-protein data at hand. It eliminates the bottleneck in the target protein identification by allowing large scale screening of compounds against the entire protein data of interest.

## 1.2. Problem Definition

To date, various machine learning-based methods have been proposed for CPI prediction using a chemogenomics approach [12]–[14]. Although the learning rule of algorithms and the ways to represent pertinent information may differ, the general principle can be described briefly as follows. We have mainly three different types of data: the list of compounds to be screened  $X_c = \{c_1, c_2, \dots, c_n\}$ , the list of proteins to be screened,  $X_p = \{p_1, p_2, \dots, p_m\}$ , and experimentally validated interaction information between some compound-protein pairs among all possible compound-protein pairs,  $X = \{c_1p_1, c_1p_2, \dots, c_np_{m-1}, c_np_m\}$ . A machine learning model is constructed to predict interaction profiles of a given compound-protein pair using a group of labeled pairs, chemical information obtained from compound data, and genomic information obtained from protein data. The inherent issue with current machine learning approaches is that a true negative dataset of known non-interacting compound-protein pairs does not exist, as positive interactions dominate the literature and the databases. Most machine learning methods that are developed so far address the CPI prediction problem within a Supervised Learning framework in which known interactions are labeled as positive, and everything else is labeled as negative. However, treating the compound-protein pairs that have no known interactions as negative leads to unrealistic recognition models as these pairs undoubtedly include some positive interactions that are as of yet unknown. Since supervised strategies require a true negative dataset to contrast with the true positive dataset of known interactions, the only option is to manufacture true negative datasets from pairings of existing compounds and proteins. This, however, entails several additional issues: Firstly, since many different true negative datasets can be manufactured based on different principles of non-interaction, classifier outputs incur a conditional bias on the selected true negative dataset and differ depending on the choice of the true negative dataset. Secondly, the presence of the unknown positive interactions in manufactured true-negative datasets of effectively untested interactions contaminates the inferred interaction recognition mechanism. There is a notable lack of studies to tackle this problem in a realistic manner [13].

In the absence of a validated true negative dataset of non-interacting compound-protein pairs, the only viable option for an unbiased and uncontaminated machine learning strategy is to contrast the set of previously untested interactions containing all

possible pairings between the compounds and proteins at hand with the true positive dataset of interacting pairs and seek those pairs in the unknown and effectively untested dataset that differentiate from the rest towards the positive interactions. This thesis proposes to address this problem using the Quasi-Supervised Learning Algorithm (QSL) [15] that does not require a true-negative dataset to contrast with the true-positive dataset. The proposed framework in this study contrasts the true positive dataset with known compound-protein interactions against the untested dataset of all possible pairings while recognizing that it contains a mixture of interacting and non-interacting compound-protein pairs. For machine learning purposes, we define a similarity between compound-protein pairs from protein-protein similarity and compound-compound similarity measures and apply the quasi-supervised learning algorithm on the combined similarity measure to calculate estimates for the posterior probability of a given compound-protein pair to belong to the true positive dataset, for all pairs in both datasets. Finally, we determine the optimal threshold for predicted positive interactions in the untested dataset using Kolmogorov-Smirnov statistics applied on posterior probability estimates.

### **1.3. Thesis Roadmap**

This thesis is organized as follows. Chapter 2 provides background information on CPI prediction, a comparison of feature vector-based and similarity-based machine learning methods for CPI prediction, and a review of related studies on similarity-based machine learning methods. Chapter 3 introduces CPI datasets used in this study. We then provide descriptions for each operational block of the schematic diagram of the proposed method shown in Figure 1. We describe the details of the proposed method for CPI prediction using the quasi-supervised learning algorithm in the next section along with the various techniques with which we characterize the similarity between compound and protein pairs. Chapter 4 presents the results of a comparative analysis of the proposed method across different configurations of alternative similarity measures. Chapter 5 concludes the thesis with a discussion on a general evaluation of the proposed method and several potential extensions for future work.

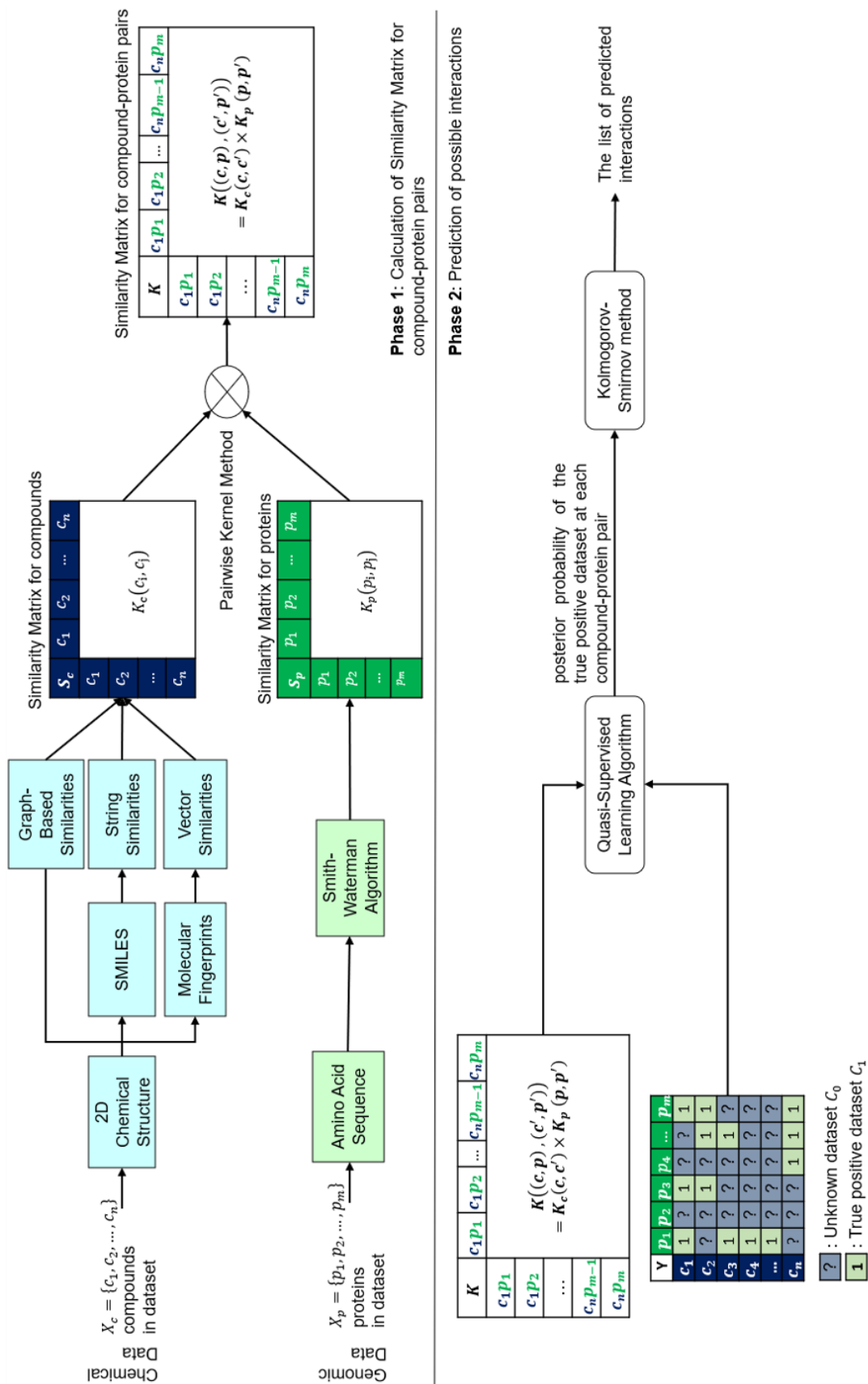


Figure 1. Schematic diagram of the proposed method

## CHAPTER 2

### BACKGROUND

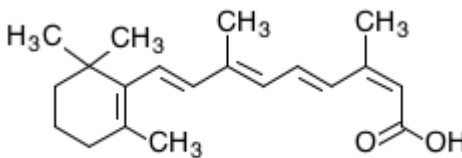
In compound-protein interaction (CPI) prediction tasks, the domain of interest consists of two different types of data: the compound data,  $X_c = \{c_1, c_2, \dots, c_n\}$  which construct chemical space and the protein data,  $X_p = \{p_1, p_2, \dots, p_m\}$  which construct genomic space.

#### 2.1. Chemical Space

In the literature, a chemical compound is represented in various ways such as Trivial Name, chemical formula, IUPAC name, 1D line notations, and 2D chemical structure diagram [16]. Trivial Name is an arbitrary or semi-arbitrary name that identifies some of the most used compounds. IUPAC (The International Union of Pure and Applied Chemistry) names standardize to assign names to the compound names under pre-defined rules. Chemical Formula (Molecular formula) describes the type and the number of atoms in a compound by character and numbers, but it does not include the structural information. 2D chemical structure diagrams are the 2D graphical representations of the structure of compounds that explains how atoms are connected to each other by which bond type without the 3D spatial position. Although 3D structure diagrams also exist, 2D chemical structure diagrams are still the most common way to represent a compound. It is often referred to as the “natural language of the chemist” [16]. 1D line notations such as SMILES encodes the structural information of compounds as a linear sequence of string characters under a specific rule. Table 2 shows the different representations of Isotretinoin.

The interaction between a drug candidate compound and a protein occurs when the compound binds to the protein via specific locations on the compound and the protein [17]. Since the common substructures may correspond to common bioactivity, the chemogenomics approaches focus on exploring the structure information of compounds such as shared substructures, ring systems, and topology to evaluate their bioactivities [10].

Table 1. Different representations of Isotretinoin  
 (Source: [https://www.kegg.jp/dbget-bin/www\\_bget?dr:D00348](https://www.kegg.jp/dbget-bin/www_bget?dr:D00348))

Trivial Name	Isotretinoin
Chemical Formula	C <sub>20</sub> H <sub>28</sub> O <sub>2</sub>
IUPAC name	(2Z,4E,6E,8E)-3,7-dimethyl-9-(2,6,6-trimethylcyclohexen-1-yl)nona-2,4,6,8-tetraenoic acid
2D chemical structure diagram	 <p>D00348</p>
SMILES	<chem>C\C(\C=C\C1=C(C)CCCC1(C)C)=C/C=C/C(/C)=C\C(O)=O</chem>

The structural information of a compound can be represented using various techniques in computer-readable form for storing and processing. The content of represented information may differ depending on the representation technique. The analogy between 2D chemical structure diagrams and topological graphs provides us the 2D representation of structural information in a computer-readable form using graph theory: The structural information is represented by undirected graphs in which atoms are mapped into vertices and bonds are mapped into edges. In this way, the structural information of compounds can be stored and processed in a computer environment via connection tables. Furthermore, many useful algorithms derived from graph theory can be applied to find common patterns between the compounds that exhibit similar bioactivity. Despite information losses due to the 3D nature of compounds, the most common way to process chemical information in pharmacological applications is to utilize 2D representation for two reasons [18]: First, it is sufficient for expert chemists whenever manual interpretation is required. Second, many studies have shown that 2D approaches often achieve better results in bioactivity prediction tasks than 3D approaches despite their simplicity. In that spirit, the Molfile format [19] (developed by MDL information systems, now BIOVIA, <https://www.3ds.com/productservices/biovia/>) has become a standard file format for storing and transferring connection tables in chemoinformatic applications [16]. Figure 2 provides the Molfile format description of Isotretinoin. The counts line indicates how many atoms and bonds construct the molecule. The atom block contains the lists of atoms indexed by the Morgan Algorithm [20] that is used to achieve canonicalization. Moreover, this block conveys 2D spatial coordinates

```

22 22 0 0 0 0 0 0 0 0 0999 V2000
20.0200 -21.8400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
18.7600 -21.1400 0.0000 C 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0
20.0200 -23.2400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
21.2100 -21.1400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
17.5700 -21.8400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
19.8800 -20.3000 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
17.5000 -20.2300 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
18.7600 -23.9400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
21.2100 -23.9400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
22.4000 -21.8400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
17.5700 -23.2400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
23.5900 -21.1400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
24.8500 -21.8400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
23.5900 -19.7400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
26.0400 -21.1400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
27.2300 -21.8400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
28.4200 -21.1400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
29.6800 -21.8400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
28.4200 -19.7400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
29.6800 -23.2400 0.0000 C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
30.8700 -23.9400 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
28.4200 -23.9400 0.0000 O 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 2 1 0 0 0
1 3 2 0 0 0
1 4 1 0 0 0
2 5 1 0 0 0
2 6 1 0 0 0
2 7 1 0 0 0
3 8 1 0 0 0
3 9 1 0 0 0
4 10 2 0 0 0
5 11 1 0 0 0
10 12 1 0 0 0
12 13 2 0 0 0
12 14 1 0 0 0
13 15 1 0 0 0
15 16 2 0 0 0
16 17 1 0 0 0
17 18 2 0 0 0
17 19 1 0 0 0
18 20 1 0 0 0
20 21 1 0 0 0
20 22 2 0 0 0
8 11 1 0 0 0
M END

```

Counts Line

Atom block

Bond block

Figure 2. Molfile of Isotretinoin  
(Source: [https://www.kegg.jp/dbget-bin/www\\_bget?-f+m+drug+D00348](https://www.kegg.jp/dbget-bin/www_bget?-f+m+drug+D00348))



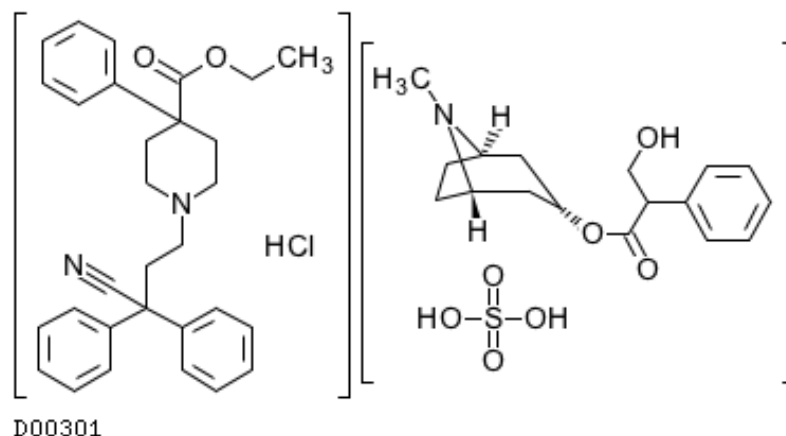
and additional information such as nonstandard isotope and valance for each atom. The bond block specifies bond types and describes how atoms are connected to each other. This block also includes additional stereochemistry information. In this way, we can distinguish between molecules whose atoms are bonded in the same way but have different 3D spatial positions. Such molecules are referred to as stereoisomers. Note that hydrogens are omitted in these representations. Since hydrogen can have only a single bond, it can be inferred from the remaining part of the graph. In this study, we retrieved the structural information of compounds in MOL file format.

The structural information can also be converted to computer-readable form through 1D line notations. This study used the Simplified Molecular Input Line Entry System (SMILES) for encoding the structural information of compounds using string characters via conversion from the Molfile format [21]. Table 2 presents the most frequently used string characters in SMILES representations. These characters encode the same information with connection tables. This representation allows us to use text mining algorithms for screening chemical space with time and memory storage advantages over the graph-based algorithms. We used the molconverter console program of JCHEM (developed by ChemAxon, <https://www.chemaxon.com/>) to convert Molfiles into canonical SMILES. The program defines SMILES by following Daylight's SMILES specification rules [22]: Each non-hydrogen atom in the organic subset (C, N, S, O, P, Cl, F, B, Br, and I) are denoted independently by their standard symbols while the other ones are denoted by their standard symbols in square brackets (e.g. [Ag], [Cu]). Square brackets are also used to specify the atoms that have charges other than normal, and isotopic specifications of the atoms (e.g. [Br-], [12C]). Hydrogen atoms are omitted except in special cases. Single, double, triple, and aromatic bonds between atoms are indicated by '-', '=', '#', and ':' symbols, respectively. Note that single and aromatic bonds between consecutive atom characters are omitted. Branches are represented by surrounding the characters in branches with parentheses. The branch specified by parentheses is always on the left. Ring structures are constructed by breaking one bond in the ring and assigning an integer to adjacent atoms connected by the broken bond. The atoms in aromatic rings are indicated by lower-case characters. Disconnected substructures in a compound are separated from the main part with the "." symbol. Since the stereoisomers may have different bioactivity characteristics due to the 3D nature of the compounds [24], it is important to represent stereoisomerism in which atoms in compounds are bonded to each other in the same way, but they have different spatial

Table 2. Most frequently used string characters in SMILES [23]

no	symbol	definition
1	C	nonaromatic carbon atoms
2	c	aromatic carbon atoms
3	N	nonaromatic nitrogen atoms
4	n	aromatic nitrogen atoms
5	O	nonaromatic oxygen atoms
6	o	aromatic oxygen atoms
7	S	nonaromatic sulfur atoms
8	s	aromatic sulfur atoms
9	F	fluorine atoms
10	Cl	chlorine atoms
11	Br	bromine atoms
12	I	iodine atoms
13	P	nonaromatic phosphorus atoms
14	p	aromatic phosphorus atoms
15	B	boron atoms
16	"X"	any other atoms
17	-	single bonds
18	=	double bonds
19	#	triple bonds
20	[	Nonorganic elements, charges, isotopes
21	-	negative charges
22	+	positive charges
23	H	explicit hydrogen atoms
24	(	acyclic branching points
25	1	nonfused ring system
26	2	bicyclic systems
27	3	tricyclic systems
28	4	tetracyclic systems
29	5	pentacyclic systems
30	6	hexacyclic systems
31	7	heptacyclic systems
32	8	octacyclic systems
33	9	nonacyclic systems
34	%	higher order ring systems

positions in space. SMILES can also encode for the stereoisomerism information depicted in 2D diagrams. The characters “/” and “\” are used to represent the arrangement around the double bonds. Chirality of a molecule can be expressed in SMILES by specifying the stereocenter with “@” or “@@” characters depending on clockwise or anti-clockwise ordering neighbor atoms of the stereocenter, respectively. The SMILES representation is illustrated in Figure 3 for Atropine sulfate.



“Cl.OS(O)(=O)=O.CN1[C@H]2CC[C@@H]1C[C@@H](C2)OC(=O)C(CO)c1cccc1.CCOC(=O)C1(CCN(CCC(C#N)(c2cccc2)c2cccc2)CC1)c1cccc1.”

Figure 3. SMILES code of Atropine sulfate with its 2D graph representation  
(Source: [https://www.kegg.jp/dbget-bin/www\\_bget?dr:D00301](https://www.kegg.jp/dbget-bin/www_bget?dr:D00301))

Lastly, the structural information of a compound can be represented by molecular fingerprints that encode the structure of compounds into fixed-length bit-vectors depending on whether a substructure occurs in a compound or not. These substructures are defined to evaluate molecules in a manner of a purpose. A study by Sawada et al. (2014) [25] investigated the performance of different types of fingerprints in compound-protein interaction prediction problems. Chapter 4 will provide a more detailed explanation of the molecular fingerprints used in this study.

## 2.2. Genomic Space

Proteins are the macromolecules that carry out the most vital tasks in organisms, such as activate or inhibit biochemical reactions, forming structures in cells and organisms, transmission of molecules, and signaling [26]. A series of small organic compounds called amino acids form proteins as a stretched polymer chain by joining with

each other in a sequence. The amino acid chain folds to form the unique 3D structure of the protein, which is the main element that determines the functions of the protein [26]. Previous studies have confirmed that the amino acid sequence of the protein is the primary determinant of its 3D structure, and by extension its function and biochemical properties [26]. Table 3 presents 20 different amino acids with their 3-letter name and 1-letter symbol representations. The amino acid sequence of a protein is usually represented as a 1D linear sequence of the 1-letter symbols of amino acids. In this way, the proteins can easily be converted into computer-readable forms for storing, delivering, and processing in FASTA file format.

Table 3. The list of 20 amino acids

no	Amino Acid	3-Letter Name	Symbol
1	Alanine	Ala	A
2	Arginine	Arg	R
3	Asparagine	Asn	N
4	Aspartic Acid	Asp	D
5	Cysteine	Cys	C
6	Glutamine	Gln	Q
7	Glutamic Acid	Glu	E
8	Glycine	Gly	G
9	Histidine	His	H
10	Isoleucine	Ile	I
11	Leucine	Leu	L
12	Lysine	Lys	K
13	Methionine	Met	M
14	Phenylalanine	Phe	F
15	Proline	Pro	P
16	Serine	Ser	S
17	Threonine	Thr	T
18	Tryptophan	Trp	W
19	Tyrosine	Tyr	Y
20	Valine	Val	V

As mentioned before, compound-protein interactions highly depend on the binding sites on compounds and proteins determined largely by their structure. Based on the strong correlation between the amino acid sequence and the structure of a protein [26], chemogenomics approaches often utilize the amino acid sequences of proteins to build machine learning algorithms for CPI prediction [10]. In this way, chemogenomics approaches aim to predict the possible interactions of the proteins for which the 3D structures are not available. Note that amino acid sequences of proteins are readily accessible but their 3D structures are not: There are ~568 million amino acid sequences

of proteins in the databases of NCBI Entrez system, while only 147.217 proteins have published 3D structures [7]. Furthermore, amino acid sequences allow us to perform large-scale screening of genomic space by leveraging various fast and efficient text-mining algorithms. In this study, our protein data consists of amino acid sequences of the proteins as is the case in most chemogenomics studies.

### **2.3. Feature-based vs Similarity-based Methods**

The machine learning methods have been developed for CPI prediction can be categorized further into feature vector-based approaches and similarity-based approaches. In feature-vector based approaches, compound-protein pairs are represented by fixed-length feature vectors that are used as input to a machine learning algorithm. The feature vector for a compound-protein pair is calculated by combining the two feature vectors, one from the genomic space, and the other from the chemical space. A machine learning algorithm is then constructed with a fixed number of parameters in order to predict an interaction label from the combined feature vector of a given compound-protein pair. However, feature extraction is a challenging process especially when it comes to CPI prediction due to the complex relationships between the chemical and the genomic spaces. Since many factors may affect the establishment of interaction between a given compound-protein pair, fixed-length vectors may not adequately reflect the critical pharmacological properties. In similarity-based approaches, machine learning algorithms can be constructed to evaluate compound-compound similarities and protein-protein similarities to predict interactions of compound-protein pairs. A well-defined similarity function that calculates the similarity between samples and a number of labeled data are sufficient to perform training and prediction [27]. It is also important to note that this strategy is inherently suitable for a chemogenomics approach as structural similarities that are key for molecular interaction may not necessarily be represented adequately through numeric features [12].

### **2.4. Literature Review of Similarity-Based Methods**

In recent years, there has been an increasing amount of literature on CPI prediction using a similarity-based chemogenomic approach [12]–[14]. The greater part of the

literature approaches this problem with supervised strategies in which the compound-protein pairs that have known interaction are labeled as positive “1”, and the rest are labeled as negative “0”. Although the similarity calculation methods and the machine learning rules may differ, the general principle of the supervised learning framework in the similarity-based scheme for CPI prediction is as shown in Figure 4. The machine learning model contrasts true positive training samples against true negative training samples by leveraging compound-compound similarities and protein-protein similarities to predict the label of a given compound-protein pair. The trained model is then evaluated on the compound-protein pair samples that are not used in the training process for three different problems of CPI prediction [28]: new drug prediction in which the model tries to predict the interaction profile of a compound that has no known interaction, the new target prediction in which the model tries to predict the interaction profile of a protein that has no known interaction, and interacting pair prediction in which the model tries to predict additional interactions of a compound or a protein that already has at least one known interaction in the dataset. Train and test set splitting for each case is shown in Figure 5 where “?” indicates the test samples.

Yamanashi et al (2008) approached the CPI prediction problem as link prediction in a bipartite graph. They used compound-compound similarities and protein-protein similarities to embed them into a pharmacological vector space in which the Euclidean distances between linked vectors are minimized [28]. Jacob and Vert (2008) developed a pairwise kernel method to obtain a similarity matrix for compound-protein pairs from similarities between compounds and similarities between proteins [29]. They then trained a Support Vector Machines (SVM) classifier using this similarity matrix as a kernel matrix. Laarhoven et al. (2011) treated interaction profiles of each protein and each compound as binary feature vectors. They constructed similarity matrices from these vectors using a Gaussian kernel and integrated them with a compound similarity matrix and a protein similarity matrix. A predicted interaction score matrix was calculated from these combined similarities using Regularized Least Squares (RLS) [30]. Laarhoven and Marchiori (2013) later expanded Gaussian Interaction Profile kernels with a Weighted Nearest Neighbor approach to predict interactions for new proteins and compounds for which no interactions exist [31]. Gönen (2012) combined non-linear dimensionality reduction and matrix factorization to project compounds and proteins into a unified low-dimensional space through their similarity matrices and to estimate an interaction matrix in this space [32]. Zheng et al. (2013) used Collaborative Matrix Factorization to estimate

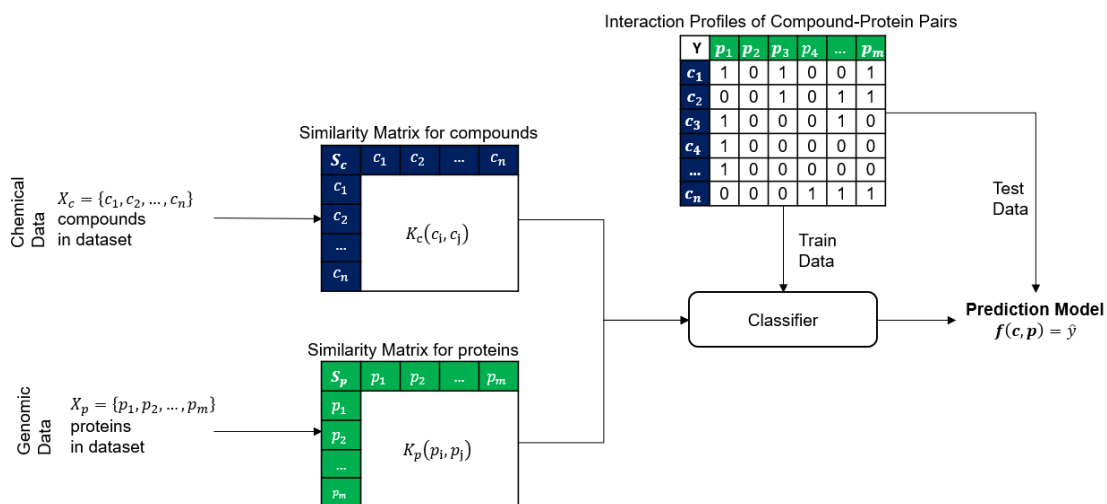


Figure 4. Supervised learning framework of CPI prediction

New Drug Prediction						
Y	$p_1$	$p_2$	$p_3$	$p_4$	...	$p_m$
$c_1$	1	0	1	0	0	1
$c_2$	0	0	1	0	1	1
$c_3$	1	0	0	0	1	0
$c_4$	1	0	0	0	0	0
...	?	?	?	?	?	?
$c_n$	?	?	?	?	?	?

New Target Prediction						
Y	$p_1$	$p_2$	$p_3$	$p_4$	...	$p_m$
$c_1$	1	0	1	0	?	?
$c_2$	0	0	1	0	?	?
$c_3$	1	0	0	0	?	?
$c_4$	1	0	0	0	?	?
...	0	1	1	0	?	?
$c_n$	1	1	0	1	?	?

New Interaction Prediction						
Y	$p_1$	$p_2$	$p_3$	$p_4$	...	$p_m$
$c_1$	?	0	1	0	0	?
$c_2$	0	?	1	0	1	1
$c_3$	1	0	0	?	?	0
$c_4$	1	0	0	0	0	0
...	0	?	1	0	0	1
$c_n$	?	1	0	1	1	?

Figure 5. Train and test splitting scheme to evaluate three different cases

a binary interaction matrix between compounds and proteins such that the latent features of the matrix approximate protein and compound similarity matrices [33].

In all these studies, the CPI prediction problem is addressed within a Supervised Learning framework that requires a true negative dataset of known non-interacting compound-protein pairs. Since there are only a few experimentally validated and documented non-interacting compound-protein pairs in the literature, the methods mentioned above resort to manufactures true negative datasets by treating all unknown interactions as negative samples. As mentioned in Chapter 1.3., constructing a true negative dataset with the pairs that do not have any validated interaction profiles causes inaccurate predictive results since these negative samples unquestionably include many hidden true positives waiting to be detected.

We propose to address this problem using the Quasi-Supervised Learning Algorithm in which the learning algorithm contrasts a true positive dataset with all unlabeled compound-protein pairs with no known interaction [15]. In this framework, we collect the compound-protein pairs at hand into two different datasets: a true positive dataset which consists of compound-protein pairs with known interactions, and an unknown dataset which consists of all the remaining compound-protein pairs. The Quasi-Supervised Learning Algorithm then calculates the posterior probability of the true positive dataset at each compound-protein pair using the asymptotic properties of nearest neighbor classification rule, compound-compound similarity measures, and protein-protein similarity measures. We then identify the potential interactions as those in the unknown dataset that are beyond the threshold posterior value that is determined by Kolmogorov-Smirnov method. In other words, we estimate the overlap between the true positive dataset and the unknown dataset in terms of the associated similarity structure.



## CHAPTER 3

### MATERIALS AND METHODS

#### 3.1. Dataset

In this study, we used the publicly available dataset published by Yamanishi et al. [28]. In this widely referenced paper, the authors point out that screening all compound-protein pairs is computationally infeasible, and construct a modular dataset to build machine learning models separately for four major protein classes (i.e. enzymes, ion channels, GPCRs, and nuclear receptors) which are commonly considered as drug targets. This dataset has since become a benchmark in CPI prediction studies [13]. The interaction information between compound-protein pairs were retrieved from DrugBank [34], KEGG [35], BRENDA [36], and SuperTarget [37] databases by Yamanishi. Table 4 shows the number of proteins and compounds and known interactions between all possible compound-protein pairs for each protein class dataset in the collection. We applied our proposed framework on Nuclear Receptor and GPCR dataset separately. We could not use Enzyme and Ion Channels dataset due to limitations of the processing power at hand.

Table 4. Datasets of Yamanishi [28]

Protein Class Dataset	Compounds	Proteins	Interactions
Enzyme	445	664	2926
Ion Channels	210	204	1476
GPCR	223	95	635
Nuclear Receptor	54	26	90

## 3.2. Similarity Measurements

In this chapter, we introduce the methods with which we evaluated compound-compound and protein-protein similarities. We used thirteen different methods to quantify the similarity between compounds and one method to quantify the similarity between proteins. Thus, we obtain thirteen different similarity matrices,  $\mathbf{S}_c \in R^{n \times n}$ , from compound data,  $X_c = \{c_1, c_2, \dots, c_n\}$ , for chemical space and one similarity matrix,  $\mathbf{S}_p \in R^{m \times m}$ , from protein data,  $X_p = \{p_1, p_2, \dots, p_m\}$ , for genomic space through the methods described below.

### 3.2.1. Compound-Compound Similarity Measurements

We retrieved the chemical structural information of compounds in Molfile format from KEGG DRUG database [35]. Similarity matrices for chemical space, denoted by a matrix  $\mathbf{S}_c$  of compound-compound similarity, are constructed using a variety of methods to evaluate the similarity between different compounds. The methods used in this study can be classified into three main categories: Graph based methods, SMILES based methods and Molecular Fingerprints based methods.

#### 3.2.1.1. Graph-Based Methods

SIMCOMP [38] algorithm was used to calculate the chemical structural similarity between compounds. This algorithm treats the 2D structure of compounds as graphs in which atoms are mapped to vertices and bonds are mapped to edges. The vertices are labelled with 68 KEGG atom types instead of the usual atomic species. Table 5 provides the list of 68 KEGG atom types. The KEGG atom types consist of three letters: The first letter corresponds to the element symbol of atom, while the second and third letters indicate its hierarchical classification depending on its hybrid orbital and atomic environments. For example, C denotes the element symbol of carbon, 1 represents sp<sup>3</sup> hybridization and a indicates a methyl Carbon in C1a. These microenvironments are defined in order to distinguish molecules in a biochemical manner in addition to their structures [38].

The algorithm finds the maximum common subgraph between two compound graphs and then calculates a similarity score using the Jaccard coefficient, defined by

$$S_c(G_1, G_2) = \frac{|G_1 \cap G_2|}{|G_1 \cup G_2|} = \frac{MCS(G_1, G_2)}{|G_1| + |G_2| - MCS(G_1, G_2)} \quad (3.1)$$

where the intersection and the union operations between graphs  $G_1$  and  $G_2$  are defined as maximum common subgraph and the nonredundant subgraph, respectively. In addition, the  $|\cdot|$  operator calculates the cardinality of its argument graph.

The challenge here is to find the maximum common subgraph. The graph theoretical explanation for this task is as follows: The complete graph is a graph in which each vertex is connected to all other vertices in the graph through edges. A clique is a complete subgraph. The maximal clique is a clique that cannot be enlarged by adding vertices. The Figure 6 illustrates these terms.

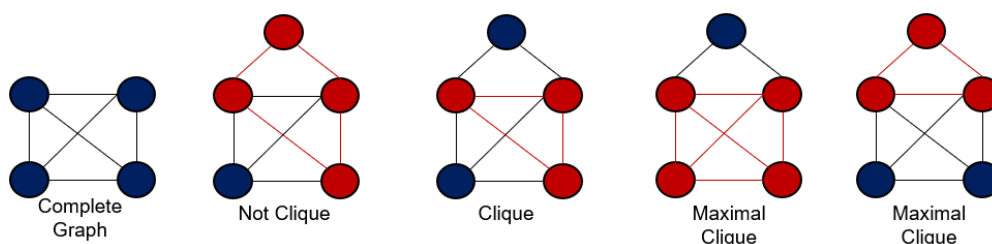


Figure 6. Fundamental terms in the graph theory

Given two graphs  $G_1(V_1, E_1)$ ,  $v_{1_i} \in V_1$ ,  $e_{1_{ij}} \in E_1$  and  $G_2(V_2, E_2)$ ,  $v_{2_k} \in V_2$ ,  $e_{2_{kl}} \in E_2$ , the algorithm firstly finds the association graph  $G(V, E)$  between them that includes all possible matching between vertices in  $G_1$  and  $G_2$ . The vertex set of the association graph is obtained by the cartesian product of two vertex sets  $V = V_1 \otimes V_2$ , with elements  $v_{ik} \in V$ . The edge set is defined by setting edges between vertices  $v_{ik}$  and  $v_{jl}$  by following adjacency conditions:

- $v_{1_i} \in V_1$  is adjacent to  $v_{1_j} \in V_1$  in  $G_1$ , and  $v_{2_k} \in V_2$  is adjacent to  $v_{2_l} \in V_2$  in  $G_2$  or
- $v_{1_i} \in V_1$  is not adjacent to  $v_{1_j} \in V_1$  in  $G_1$ , and  $v_{2_k} \in V_2$  is not adjacent to  $v_{2_l} \in V_2$  in  $G_2$

Figure 7 illustrates the calculation of an association graph from two graphs with their adjacency matrices where dashed lines in the association graph denote the matching of the non-adjacent vertices. The index  $v_{ik}$  can be regarded as indicating the row of the adjacency matrix of the association graph in which blue color denotes the nodes coming from the first graph  $v_i$  and the green color denotes the nodes coming from the second graph  $v_k$ , whereas  $v_{jl}$  indicates the column of the adjacency matrix. Maximal cliques in the association graph, which corresponds to common subgraphs between two graphs, are detected by the Bron-Kerbosch algorithm from the adjacency matrix of the association graph. Each vertex in maximal cliques is assigned a weight based on the labels of the matching vertices using the function, defined by

$$w(v_{ik}) = \begin{cases} 1 & p(v_{1_i}) = p(v_{2_k}) \\ 0.5 & p(v_{1_i}) \neq p(v_{2_k}) \text{ and } a(v_{1_i}) = a(v_{2_k}) \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

where the function  $p$  returns the KEGG atom type and the function  $a$  returns the atom species of its argument vertex. The maximal clique for which the summation of the weights of the vertices is highest is identified as the Maximum Common Subgraph.

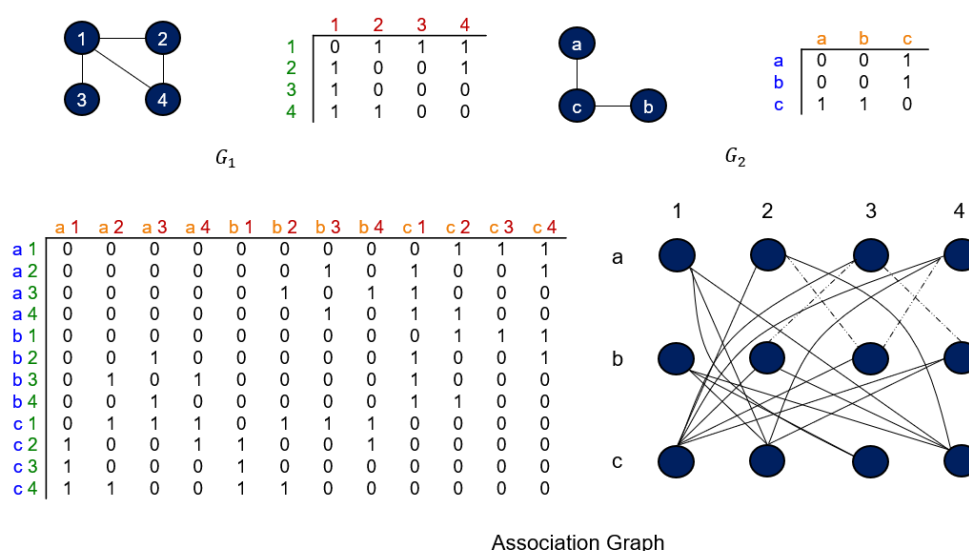


Figure 7. Calculation of the association graph

Table 5. The list of 68 KEGG atom types  
 (Source: <https://www.genome.jp/kegg/reaction/KCF.html>)

Atom	Functional group	Atom type	Description
C	Alkane	C1a	R-CH3
		C1b	R-CH2-R
		C1c	R-CH(-R)-R
		C1d	R-C(-R)2-R
	Cyclic alkane	C1x	ring-CH2-ring
		C1y	ring-CH(-R)-ring
		C1z	ring-C(-R)2-ring
	Alkene	C2a	R=CH2
		C2b	R=CH-R
		C2c	R=C(-R)2
	Cyclic alkene	C2x	ring-CH=ring
		C2y	ring-C(-R)=ring or ring-C(=R)-ring
	Alkyne	C3a	R≡CH
		C3b	R≡C-R
	Aldehyde	C4a	R-CH=O
	Ketone	C5a	R-C(=O)-R
	Cyclic ketone	C5x	ring-C(=O)-ring
	Carboxylic acid	C6a	R-C(=O)-OH
		Carboxylic ester	C7a
	C7x		ring-C(=O)-O-ring
Aromatic ring	C8x	ring-CH=ring	
	C8y	ring-C(-R)=ring	
Undefined C	C0		
N	Amine	N1a	R-NH2
		N1b	R-NH-R
		N1c	R-N(-R)2
		N1d	R-N(-R)3+
	Cyclic amine	N1x	ring-NH-ring
		N1y	ring-N(-R)-ring
	Imine	N2a	R=N-H
		N2b	R=N-R
	Cyclic imine	N2x	ring-N=ring
		N2y	ring-N(-R)+=ring
	Cyan	N3a	R≡N
	Aromatic ring	N4x	ring-NH-ring
		N4y	ring-N(-R)-ring
N5x		ring-N=ring	
N5y		ring-N(-R)+=ring	

(cont. on next page)

Table 5. (cont.)

N	Undefined N	N0	
O	Hydroxy	O1a	R-OH
		O1b	N-OH
		O1c	P-OH
		O1d	S-OH
	Ether	O2a	R-O-R
		O2b	P-O-R
		O2c	P-O-P
		O2x	ring-O-ring
	Oxo	O3a	N=O
		O3b	P=O
		O3c	S=O
	Aldehyde	O4a	R-CH=O
	Ketone	O5a	R-C(=O)-R
		O5x	ring-C(=O)-ring
	Carboxylic acid	O6a	R-C(=O)-OH
Ester	O7a	R-C(=O)-O-R	
	O7x	ring-C(=O)-O-ring	
Undefined O	O0		
S	Thiol	S1a	R-SH
	Thioether	S2a	R-S-R
		S2x	ring-S-ring
	Disulfide	S3a	R-S-S-R
		S3x	ring-S-S-ring
	Sulfate	S4a	R-SO <sub>3</sub>
Undefined S	S0		
P	Attached to other elements	P1a	P-R
	Attached to oxygen	P1b	P-O
Other	Halogens	X	F, Cl, Br, I
	Others	Z	

In our implementation of the algorithm, the maximum common subgraph was determined using RDKit open-source cheminformatics python library [39] where vertices are labelled by atomic species instead of 68 KEGG atoms. In addition to their types, vertices are also distinguished by their valance information and bonds are distinguished by their aromaticity and ring information.

### 3.2.1.2. SMILES-Based Methods

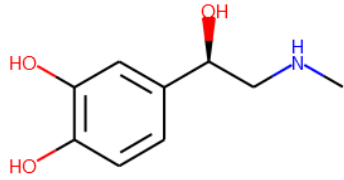
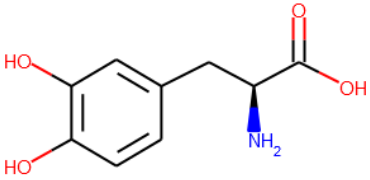
Simplified Molecular Input Line Entry System (SMILES) is a 1D string representation that encodes the structural information of compounds [21]. A Study by Öztürk et al. (2016) suggested that text similarity between two SMILES strings can be considered as a measure of structural similarity between two compounds for CPI prediction tasks [40]. They showed that similarity measures using various SMILES kernels performed as well as graph-based methods with an additional computational time advantage. We generated SMILES strings for each compound from their 2D structural information in Molfile format using JCHEM (developed by ChemAxon, <https://www.chemaxon.com/>). The program defines SMILES by following Daylight's SMILES specification rules [22], as summarized earlier in Chapter 2.1. We then constructed various SMILES kernels to calculate a similarity score between two SMILES strings. To this end, we used Normalized Longest Common Subsequence (NLCS), Combination of Longest Common Subsequence Models (CLCS), LINGO-q Similarity, LINGO Based Term Frequency (TF) Cosine Similarity, and LINGO Based Term Frequency-Inverse Document Frequency (TF-IDF) Cosine Similarity which are proposed by Öztürk et al. to calculate similarity between two compounds. For illustration purposes in the description below, we apply these kernels over the SMILES strings of Epinephrine and Levodopa shown in Table 6 along with their 2D structures.

In NLCS, the kernel finds the longest common subsequence between two SMILES. A similarity score between two SMILES strings is then calculated by cosine normalization as

$$\text{Similarity}_{NLCS}(S_1, S_2) = \frac{\text{len}(LCS(S_1, S_2))^2}{\text{len}(S_1) \times \text{len}(S_2)} \quad (3.3)$$

where  $LCS(S_1, S_2)$  denotes the longest common subsequence, and  $len(.)$  operator calculates the number of strings in its argument SMILES. For Epinephrine and Levodopa, the kernel function calculates a similarity score of 0.588 given the longest common subsequence “N[C@H](c1ccc(O)c(O)c1”.

Table 6. SMILES representations and 2D structure diagrams of Epinephrine and Levodopa

Epinephrine	Levodopa
	
<chem>CNC[C@H](O)c1ccc(O)c(O)c1</chem>	<chem>N[C@@H](Cc1ccc(O)c(O)c1)C(O)=O</chem>

Note that the longest common subsequence is not required to be consecutive. In order to achieve a more meaningful semantic similarity between two strings, Maximal Consecutive Longest Common Subsequence ( $MCLCS$ ) starting from the first character and character  $n$  are calculated.  $CLCS$  is then defined as the equal-weighted average of their cosine normalizations and  $NLCS$  as

$$Similarity_{CLCS}(S_1, S_2) = \frac{1}{3} (Similarity_{MCLCS_n} + Similarity_{MCLCS_1} + Similarity_{NLCS}) \quad (3.4)$$

where  $Similarity_{MCLCS_n}$  and  $Similarity_{MCLCS_1}$  are calculated as

$$Similarity_{MCLCS_n}(S_1, S_2) = \frac{len(MCLCS_n(S_1, S_2))^2}{len(S_1) \times len(S_2)} \quad (3.5)$$

and

$$Similarity_{MCLCS_1}(S_1, S_2) = \frac{len(MCLCS_1(S_1, S_2))^2}{len(S_1) \times len(S_2)} \quad (3.6)$$



respectively. In the case of Epinephrine and Levodopa, Maximal Consecutive Longest Common Subsequence (MCLCS) becomes ““c1ccc(O)c(O)c1”” providing a similarity score of 0.283.

LINGO- $q$  stands for consecutive  $q$ -character substrings that can be created from a SMILES string [41]. For instance, Table 7 presents LINGOs( $q = 4$ ) that can be extracted from the SMILES strings of Epinephrine and Levodopa with their occurrence frequencies in SMILES. Note that all ring numbers must be replaced with 0s before the LINGO extraction process [41].

Table 7. LINGOs with their corresponding frequencies in the SMILES strings of Epinephrine and Levodopa

Epinephrine		Levodopa	
LINGO	Frequency	LINGO	Frequency
CNC[	1	N[C@	1
NC[C	1	[C@@	1
C[C@	1	C@@H	1
[C@H	1	@@H]	1
C@H]	1	@H](	1
@H](	1	H](C	1
H](O	1	](Cc	1
](O)	1	(Cc0	1
(O)c	3	Cc0c	1
O)c0	2	c0cc	1
)c0c	1	0ccc	1
c0cc	1	ccc(	1
0ccc	1	cc(O	1
ccc(	1	c(O)	2
cc(O	1	(O)c	2
c(O)	2	O)c(	1
O)c(	1	)c(O	1
)c(O	1	O)c0	1
		)c0)	1
		c0)C	1
		0)C(	1
		)C(O	1
		C(O)	1
		(O)=	1
		O)=O	1

A similarity function based on Tanimoto Coefficient, LINGOsim, then calculates a similarity score between the two SMILES strings using the unique LINGOs that are extracted from them as

$$LINGOsim(S_1, S_2) = \frac{1}{m} \sum_{i=1}^m 1 - \frac{|N_{S_1,i} - N_{S_2,i}|}{|N_{S_1,i} + N_{S_2,i}|} \quad (3.7)$$

where  $m$ ,  $N_{S_1,i}$ , and  $N_{S_2,i}$  indicate the total number of unique LINGOs in both SMILES along with the frequencies of the  $i^{th}$  LINGO in the first and the second SMILES strings, respectively. Therefore, the kernel function measures the similarity between Epinephrine and Levodopa as 0.287. We used  $q = 3, 4, 5$  as in the original study [40].

In order to calculate a similarity score between two SMILES strings, the SMILES strings can also be mapped into vectors whose common length equals the total number of unique LINGOs in the two strings. In Lingo-based Term Frequency (TF) cosine similarity, the TF of each unique LINGO reflects the occurrence frequency of the LINGO in SMILES and are collected into feature vectors.  $TF_{weight}$  of the  $i^{th}$  unique LINGO for a SMILES is then calculated as

$$TF_{weight,i} = \begin{cases} 1 + \log_{10} TF_i & \text{if } TF_i > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.8)$$

where  $TF_i$  denotes the frequency of the  $i^{th}$  LINGO in the corresponding SMILES string. The feature vectors in which each element corresponds to  $TF_{weight}$  of one of the unique LINGOs in the SMILES representation of Epinephrine and Levodopa, {0ccc, [C@H, O]C(, O)c0, C@H, )c(O, c0)C, @@H], C@@H, O)c(, (O)=, H](C, C[C@, c(O), (Cc0, )c0), CNC[, )c0c, N[C@, ccc(, cc(O, NC[C, (O)c, [C@@, ](Cc, )C(O, Cc0c, O)=O, c0cc, C(O), ](O), H](O, @H)](}, are calculated as

$$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1.301 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1.301 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1.477 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1.301 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1.301 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

respectively. The similarity score between two compound smiles is finally calculated by cosine similarity as

$$TF_{sim}(S_1, S_2) = \frac{V_{S_1} \cdot V_{S_2}}{\|V_{S_1}\| \times \|V_{S_2}\|} \quad (3.9)$$

Consequently, the kernel function measures the similarity between Epinephrine and Levodopa as 0.511.

In LINGO based TF-IDF similarity, the TF values of LINGOs are multiplied with their Inverse Document Frequency that reflects the occurrence frequency of the LINGOs

in the whole SMILES dataset and then collected into reference vectors. TF assigns higher values to more frequently occurring LINGOs in a SMILES, while IDF assigns lower values to more frequently occurred LINGOs in the dataset.  $IDF_{weight}$  the  $i^{th}$  unique LINGO for a SMILES string is calculated as

$$IDF_{weight,i} = \log_{10}\left(\frac{N}{IDF_i}\right) \quad (3.10)$$

where  $N$  corresponds to total SMILES in the dataset and  $IDF_i$  corresponds to the number of SMILES that contains the  $i^{th}$  unique LINGO in the dataset. In this setting, the feature vectors for Epinephrine and Levodopa are calculated as

$$\begin{bmatrix} 0.118 \\ 0.563 \\ 0 \\ 0.723 \\ 0.563 \\ 1.07 \\ 0 \\ 0 \\ 0 \\ 1.118 \\ 0 \\ 0 \\ 0.616 \\ 1.214 \\ 0 \\ 0 \\ 1.649 \\ 0.424 \\ 0 \\ 0.375 \\ 0.78 \\ 1.57 \\ 1.309 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.093 \\ 0 \\ 0.917 \\ 0.871 \\ 0.6 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0.118 \\ 0 \\ 0.83 \\ 0.556 \\ 0 \\ 1.07 \\ 0.871 \\ 0.624 \\ 0.624 \\ 1.118 \\ 0.705 \\ 0.95 \\ 0 \\ 1.214 \\ 1.144 \\ 1.746 \\ 0 \\ 0 \\ 1.348 \\ 0.375 \\ 0.78 \\ 0 \\ 1.153 \\ 0.624 \\ 1.394 \\ 0.901 \\ 0.456 \\ 0.563 \\ 0.093 \\ 0.491 \\ 0 \\ 0 \\ 0.6 \end{bmatrix}$$

respectively. Cosine similarity between the two vectors provides a similarity score between the two SMILES strings as 0.393.

These processes are applied to all possible SMILES pairs in the dataset to construct a similarity matrix,  $S_c$ , between all compounds.

### 3.2.1.3. Molecular Fingerprints based methods

Fingerprints encode the structure of compounds into fixed-length bit vectors depending on whether a substructure occurs in a compound or not. As opposed to LINGO where substructures are constructed from SMILES strings through  $n$  consecutive string characters, pre-defined substructures are designed for specific purposes or extracted from 2D structures directly. A study by Sawada et al. (2014) investigated the performance of different types of fingerprints and similarity functions in compound-protein interaction prediction problems [25]. In our study, we used Extended-Connectivity Fingerprints (ECFP) [42], Functional-Class Fingerprints (FCFP) [42], Molecular ACCess System (MACCS) fingerprints [43], KEGG Chemical Function and Substructures (KCF-S) descriptors [44] that have previously been identified as useful in CPI prediction [25].

ECFP describes the structure of a molecule by encoding substructures formed by a circular neighborhood of each atom within an atom radius into 1024-length binary vectors, an approach that is also known as a circular fingerprint, or Morgan Fingerprints [39]. In this study, we set the radius as 2 providing a maximum range between fingerprint atoms of 4 (ECFP4). The common substructure of Epinephrine and Levodopa, encoded into the 451<sup>th</sup> feature of their ECFP4 fingerprints, is shown in Figure 8. The blue circle denotes the center atom of the substructure, while the yellow ones indicate the aromatic atoms in the circular neighborhood. The non-aromatic neighbor atom of the center atom is represented without any specific color. FCFP is an extension of ECFP in which pharmacophore roles of atoms are also added to fingerprints to encode for functional substructural features instead of just atom environments. The cosine similarity then provides similarity scores between ECFP4 and FCFP4 fingerprints of Epinephrine and Levodopa as 0.453 and 0.535, respectively.



Figure 8. 451. bit of the ECFP4 fingerprints of Epinephrine and Levodopa

The MACCS fingerprints describe the structure of a molecule with a 166 length bit vector whose elements correspond to a substructure key. These publicly available substructure keys are developed by a private company (previously MDL Information Systems, now BIOVIA, <https://www.3ds.com/products-services/biovia/>) in order to calculate a molecular similarity. The cosine similarity between MACCS fingerprints of Epinephrine and Levodopa is measured as 0.742. We used the RDKit python library [39] to construct these fingerprints and calculated similarity scores between each fingerprint pair using cosine similarity.

One known drawback of fingerprints is that they encode for only the presence or absence of substructures and disregard the copy number for multiply present substructures. KCF-S addresses this problem using integer-valued vectors of counts instead of binary vectors: It treats the 2D chemical structure of a compound as a graph and characterizes the structure by an integer-valued vector in which each element of the vector corresponds to the number of distinct copies of a substructure that the compound possesses. Moreover, instead of atomic species such as C, H, O, N, P, and so on, it uses the 68 KEGG atoms. As mentioned in Chapter 3.2.1., these labels are designed to reflect physiochemical environmental information of the atom in order to distinguish molecules from a biochemical viewpoint. Substructures are constructed from the graph of a compound using seven chemical structural attributes: atom, bond, triplet, vicinity, ring, skeleton, and inorganic. Some substructures that can be created from Epinephrine using these attributes are summarized in Table 8. Two substructures extracted from Epinephrine and Levodopa are illustrated in Figure 9 and Figure 10, respectively. The substructures of Epinephrine shown in Figure 9 are also marked with bold characters in Table 8. The attributes of atom correspond to the number of each different labeled node that the compound graph possesses. The node pairs that are connected to each other through a chemical bound are collected into bond attributes. The triplet attributes are extracted from

Table 8. A reduced list of substructures that can be extracted from Epinephrine using KCF-S fingerprint attributes

String	Attribute Type	Level	Count
C	atom	Atom species	9
C8	atom	Atom class	6
O	atom	Atom species	3
C1	atom	Atom class	3
O1	atom	Atom class	3
C8y	atom	KEGG atom	3
C8x	atom	KEGG atom	3
O1a	atom	KEGG atom	3
C-C	bond	Atom species	8
C8-C8	bond	Atom class	6
C8x-C8y	bond	KEGG atom	4
C-O	bond	Atom species	3
C-N	bond	Atom species	2
C8-O1	bond	Atom class	2
C8y-C8y	bond	KEGG atom	1
C1b-N1b	bond	KEGG atom	1
C1a-N1b	bond	KEGG atom	1
C-C-C	triplet	Atom species	34
C8-C8-C8	triplet	Atom class	24
C-C-O	triplet	Atom species	12
C8-C8-O1	triplet	Atom class	8
C8y-C8x-C8y	triplet	KEGG atom	6
C8x-C8y-C8x	triplet	KEGG atom	6
C-N-C	triplet	Atom species	4
C1-C8-C8	triplet	Atom class	4
C1-N1-C1	triplet	Atom class	4
C8x-C8x-C8y	triplet	KEGG atom	4
C8x-C8y-C8y	triplet	KEGG atom	4
C-C-C,2-O	vicinity	Atom species	3
C8-C8-C8,2-O1	vicinity	Atom class	2
C8x-C8y-C8y,2-O1a	vicinity	KEGG atom	2
C-C-C,2-C	vicinity	Atom species	1
C1-C8-C8,2-C8	vicinity	Atom class	1
C1-C1-C8,2-O1	vicinity	Atom class	1
C1c-C8y-C8x,2-C8x	vicinity	KEGG atom	1
<b>C1b-C1c-C8y,2-O1a</b>	<b>vicinity</b>	<b>KEGG atom</b>	<b>1</b>
C-C-C-C-C,1-6	ring	Atom species	1
C8-C8-C8-C8-C8,1-6	ring	Atom class	1
<b>C8x-C8x-C8y-C8x-C8y-C8y,1-6</b>	<b>ring</b>	<b>KEGG atom</b>	<b>1</b>
C-C-C-C-C,2-C-C	skeleton	Atom species	1
C1-C1-C8-C8-C8-C8,3-C8-C8-6	skeleton	Atom class	1
C1b-C1c-C8y-C8x-C8x-C8y,3-C8x-C8y-6	skeleton	KEGG atom	1

three sequential nodes that are connected through a center atom. An attribute of vicinity represents a central node and all nodes that attach to the center node. This attribute includes many important functional groups. The cyclic substructures are encoded into the ring attributes. The carbon skeleton of the molecule, such as alkyl and aryl groups, is represented through the skeleton attribute. An inorganic attribute corresponds to a connected atom group without any carbon atom. Each substructure is constructed for three different levels separately: using the first letters of KEGG atom labels which denotes atomic species, the first two letters of KEGG atom labels which are referred to as atomic class, and KEGG atom labels.

The dimension of KCF-S fingerprint vectors equals the number of unique substructures listed in a database of substructures that can be extracted from the compounds. For example, we have 148680 total substructures for 223 compounds in GPCR dataset. We used KCF-Convoy python package [45] to construct fingerprint vectors and calculated the similarity between two  $m$ -dimension KCF-S fingerprint vectors using a weighted Tanimoto similarity as

$$KCF - S_{similarity}(V_1, V_2) = \frac{N_{12}}{N_1 + N_2 + N_{12}} \quad (3.11)$$

$$N_1 = \sum_{i=1}^m \max(0, v_{1,i} - v_{2,i}) \quad (3.12)$$

$$N_2 = \sum_{i=1}^m \max(0, v_{2,i} - v_{1,i}) \quad (3.13)$$

$$N_{12} = \sum_{i=1}^m \min(v_{1,i}, v_{2,i}) \quad (3.14)$$

where *max* and *min* operators return the maximum and the minimum value of their inputs, respectively. Therefore, the similarity score between Epinephrine and Levodopa is calculated as 0.420.



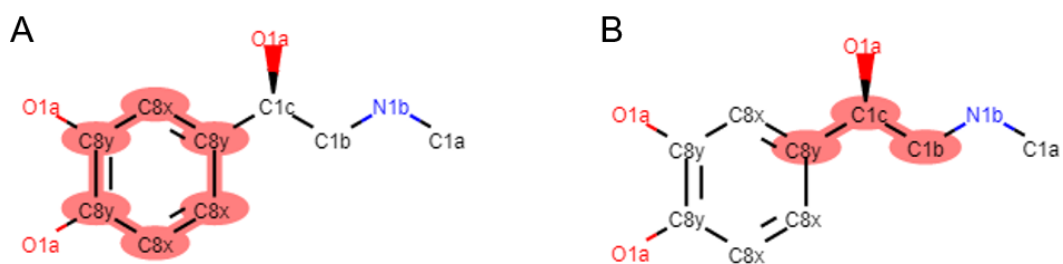


Figure 9. The highlighted substructures that correspond to the ring attribute C8x-C8x-C8y-C8x-C8y-C8y,1-6 (A) and the vicinity attribute C1b-C1c-C8y,2-O1a (B) in KCF-S fingerprints of Epinephrine

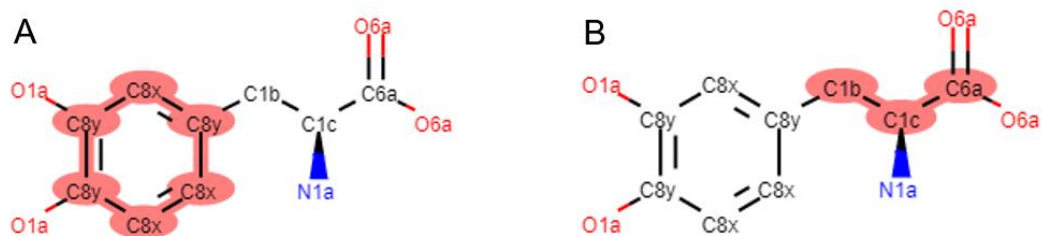


Figure 10. The highlighted substructures that correspond to the ring attribute C8x-C8x-C8y-C8x-C8y-C8y,1-6 (A) and the vicinity attribute C1b-C1c-C6a,2-N1a (B) in KCF-S fingerprints of Levodopa

### 3.2.2. Protein-Protein Similarity Measurements

We retrieved the amino acid sequences of all proteins in the datasets in FASTA format from KEGG GENES database [35]. The similarity between two amino acid sequences can be evaluated using various sequence alignment algorithms in which sequences are slid with respect to each other under a variety of preset rules until as many as possible amino acids are matched [46]. The fact that amino acid sequences are subject to mutations necessitates padding the sequences with gaps in suitable places, in order to obtain the most likely alignment. In addition, since some mutations are more likely than others, we must take account into the mutation propensity of different amino acids. These problems can be solved using dynamic programming in which an algorithm penalizes unmatched pairs and gaps, and rewards matching pairs with predetermined scores that also consider the probability of substitutions. The optimal alignment is obtained where the algorithm returns the maximum alignment score. Note that the best alignment is subjective concept and highly depends on scoring choice.

In this study, a similarity matrix for genomic space, denoted by a matrix  $\mathbf{S}_p$  of protein-protein similarity, was constructed by calculating the similarity between each protein pair in the dataset using Normalized Smith-Waterman Algorithm. Smith-Waterman is a sequence alignment algorithm based on a local alignment scoring strategy that searches similarities among the local regions without necessarily aligning entire sequences [47]. As a result, the algorithm returns an alignment score for the conserved regions between the two sequences [47]. Since these conserved regions can be responsible for common bioactivity and functional similarity, the Normalized Smith-Waterman Algorithm offers a more biologically significant assessment compared to global alignment. Figure 11 illustrates the difference between global and local alignment. Furthermore, Sawada et al. (2014) experimentally showed that the protein-protein similarities based on local alignment outperformed the protein-protein similarities based on global alignment in CPI prediction problem as expected [25]. In the implementation, the local alignment score is normalized as

$$\mathbf{S}_p(p_i, p_j) = \frac{SW(p_i, p_j)}{\sqrt{SW(p_i, p_i) \times SW(p_j, p_j)}} \quad (3.15)$$

in order to obtain a similarity score between 0 and 1, where  $SW(p_i, p_j)$  denotes the alignment score of the Smith-Waterman algorithm. In this study, we used the default values for the parameters of the algorithm as provided in Pairwise Sequence Alignment Tool of EMBOSS ([https://www.ebi.ac.uk/Tools/psa/emboss\\_water/](https://www.ebi.ac.uk/Tools/psa/emboss_water/)). BLOSUM62 scoring matrix, which is constructed using observed substitutions in the great number of conserved regions referred as blocks, is used to reflect the mutation propensity of different amino acids [48].



Figure 11. The alignment of homeodomain region of homeobox genes from Mouse and Human using Needleman Wunsch (global alignment) (A) and Smith-Waterman (local alignment) (B) [46]

### 3.2.3. Pairwise Kernel Method

A similarity matrix must satisfy Mercer's Theorem to be used in a machine learning algorithm as a kernel matrix, which means that it has to be symmetric and positive semi-definite, i.e. all eigenvalues must be non-negative [27]. In order to ensure that compound-compound and protein-protein similarity matrices satisfy these criteria, we firstly calculated symmetric and regularized compound and protein similarity matrices  $K_c$  and  $K_p$  by

$$K_p = \frac{S_p + S_p^T}{2} + |\lambda_{\min}(S_p)|I \quad \text{and} \quad K_c = \frac{S_c + S_c^T}{2} + |\lambda_{\min}(S_c)|I \quad (3.16)$$

where the diagonal entries of the symmetric similarity matrices are augmented by the minimum eigenvalue of the corresponding compound or protein similarity matrices. We then used the pairwise kernel method [29] to calculate the joint similarity between compound-protein pairs  $(c, p)$  and  $(c', p')$ . Jacob [29] considered compound-compound and protein-protein similarity as dot product of two vectors in an infinite Hilbert space.

$$K_c(c, c') = \varphi_c(c)^T \varphi_c(c') \quad \text{and} \quad K_p(p, p') = \varphi_p(p)^T \varphi_p(p') \quad (3.17)$$

Based on the assumption that a compound-protein pair can be represented in a Hilbert space through Kronecker product of the compound and the protein maps by

$$\varphi(c, p) = \varphi_c(c) \otimes \varphi_p(p) \quad (3.18)$$

we can express the Pairwise Kernel Function as

$$\mathbf{K}((c, p), (c', p')) = \varphi(c, p)^T \varphi(c', p') \quad (3.19)$$

$$= (\varphi_c(c) \otimes \varphi_p(p))^T ((\varphi_c(c') \otimes \varphi_p(p'))) \quad (3.20)$$

$$= \varphi_c(c)^T \varphi_c(c') \times \varphi_p(p)^T \varphi_p(p') \quad (3.21)$$

$$= \mathbf{K}_c(c, c') \times \mathbf{K}_p(p, p') \quad (3.22)$$

which is tantamount to constructing a similarity matrix  $\mathbf{K}$  between compound-protein pairs by the Kronecker product of  $\mathbf{K}_c$  and  $\mathbf{K}_p$  as (Jacob et al., 2008)

$$\mathbf{K} = \mathbf{K}_c \otimes \mathbf{K}_p. \quad (3.23)$$

### 3.3. Quasi-Supervised Learning Algorithm

The Quasi-Supervised Learning Algorithm (QSL) was developed by Karacali (2010) to address one of the major problems of biomedical data analysis, the possible lack of ground-truth labeled data for a class of interest [15]. In this learning strategy, given a two-class recognition scenario with labeled samples of only one of the classes, the data at hand are divided into two datasets: One dataset, say  $C_1$ , consists of the labelled samples of the known class, while the other one,  $C_0$ , consists of all samples without any label. A numerical algorithm then allows nonparametric estimation of posterior probability of each sample belonging to  $C_0$  and  $C_1$  using the asymptotic properties of the nearest neighbor classification rule. Using the estimated posterior probabilities, we can evaluate

the overlap between  $C_0$  and  $C_1$  for automatic labelling of the samples in the unlabeled dataset  $C_0$  that appear among  $C_1$  samples.

The QSL algorithm can be derived leveraging the asymptotic properties of nearest neighbor classification rule as follows: Given  $M$  reference sets  $\{R_1, R_2, \dots, R_M\}$  for nearest neighbor classification constructed with  $n$  samples from  $C_0$  and  $C_1$  each, the average rate of assigning a sample  $x$  to  $C_0$  and  $C_1$  using nearest neighborhood classification with reference to  $R_1, R_2, \dots, R_M$ ,

$$f_1(x) = \frac{1}{M} \sum_{m=1}^M \mathbf{1}(x \text{ is assigned to } C_1 \text{ with reference to } R_m) \quad (3.24)$$

$$f_0(x) = \frac{1}{M} \sum_{m=1}^M \mathbf{1}(x \text{ is assigned to } C_0 \text{ with reference to } R_m) \quad (3.25)$$

where the assignment of a sample  $x$  to  $C_0$  and  $C_1$  is made using a nearest neighbor classifier using the indicated reference set, can be expressed as the likelihood ratio

$$\frac{f_1(x)}{f_0(x)} \simeq \frac{P(x|x \in C_1)}{P(x|x \in C_0)} \quad (3.26)$$

derived from

$$f_1(x) \simeq \frac{P(x|x \in C_1)}{P(x|x \in C_1) + P(x|x \in C_0)} \quad \text{and} \quad f_0(x) \simeq \frac{P(x|x \in C_0)}{P(x|x \in C_1) + P(x|x \in C_0)} \quad (3.27)$$

as long as  $M$  is sufficiently large.

Since the reference set includes equal numbers of samples from  $C_0$  and  $C_1$ , the prior probabilities of datasets will be equal ( $P(C_0) = P(C_1)$ ) in the reference set. Therefore, by the Bayes Rule, the probability of assigning a sample  $x$  to  $C_0$  and  $C_1$  using nearest neighborhood classification with a randomly selected reference set among  $R_1, R_2, \dots, R_M$  will approximate the posterior probabilities under the assumption for equal priors. Mathematically, this can be expressed as

$$P(C_1|x) \simeq f_1(x) = \frac{1}{M} \sum_{m=1}^M \mathbf{1}(x \text{ is assigned to } C_1 \text{ with reference to } R_m) \quad (3.28)$$

$$P(C_0|x) \simeq f_0(x) = \frac{1}{M} \sum_{m=1}^M \mathbf{1}(x \text{ is assigned to } C_0 \text{ with reference to } R_m) \quad (3.29)$$

Since carrying out great numbers of nearest neighbor classification is not feasible due to the associated computational expense, the Quasi-Supervised Learning Algorithm provides a fast and efficient numerical calculation of the rates  $f_1(x)$  and  $f_0(x)$  for each sample  $x_i$  in the collection as described below. Finally, the optimal value for the parameter  $n$  is found by minimizing the cost function

$$E(n) = 4 \sum_i f_1(x_i) f_0(x_i) + 2n \quad (3.30)$$

where the first term penalizes large overlaps between  $C_0$  and  $C_1$ , and the second term penalizes large  $n$  for better generalization.

We applied the QSL strategy to predict potential interactions between the compound-protein pairs that do not have any known interaction. To this end, we constructed two datasets: The unknown dataset  $C_0$  which includes the untested compound-protein pairs that do not have a documented interaction, and the true positive dataset  $C_1$  which includes the compound-protein pairs whose interactions are experimentally validated. The samples in  $C_0$  are assigned with a label of 0 ( $y = 0$ ) and the samples in  $C_1$  are assigned with a label of 1 ( $y = 1$ ). Then, the QSL algorithm calculates the posterior probability of the true positive dataset  $P(C_1|x)$  at each compound-protein pair  $x = (c, p)$  using the similarity matrix,  $\mathbf{K}$  of all compound-protein pairs.

### 3.3.1. Efficient Numerical Computation of the Posterior Probability Estimations

Let us assume that we want to calculate the posterior probability of the true positive dataset at a compound-protein pair sample  $x$ . Let also a reference set  $R$  be

constructed using  $n$  samples from the unknown dataset  $C_0$  and  $n$  samples from the true positive dataset  $C_1$ . Firstly, we sort all other samples  $x_i$  with their labels  $y_i$  by their similarities to  $x$  in a descending order using joint similarity matrix  $k_i = \mathbf{K}(x, x_i)$  where  $k_i$  denotes the similarity between the sample  $x$  and each other sample  $x_i$  for  $i = 1, 2, \dots, l$ . In this way, we obtain an ordered sequence of compound-protein pair similarities  $\{k_i\}$  with  $k_{(1)} \geq k_{(2)} \geq \dots \geq k_{(l)}$  providing  $x_{(1)}$  and  $y_{(1)}$  as the nearest neighbor and its label, respectively. For analytical computation, we then decompose the posterior probability of the true positive dataset  $P(C_1|x)$  for a compound-protein sample  $x$  respect to  $R$  on whether or not the nearest neighbor  $x_{(1)}$  is in  $R$  and coming from  $C_1$  as

$$f_1(x) = P(x_{(1)} \in R)\mathbf{1}(y_{(1)} = 1) + P(x_{(1)} \notin R)P(y = 1|x_{(1)} \notin R) \quad (3.31)$$

Furthermore, we carry out same decomposition over  $P(y = 1|x_1 \notin R)$  on whether or not the second nearest neighbor  $x_2$  is in  $R$  and belongs to  $C_1$  as

$$P(y = 1|x_{(1)} \notin R) = P(x_{(2)} \in R|x_{(1)} \notin R)\mathbf{1}(y_{(2)} = 1) + P(x_{(2)} \notin R|E_1)P(y = 1|E_2) \quad (3.32)$$

where  $E_2$  denotes the joint event  $x_{(1)}, x_{(2)}, \notin R$ . This decomposition is generalized as

$$P(y = 1|E_{k-1}) = P(x_{(k)} \in R|E_{k-1})\mathbf{1}(y_{(k)} = 1) + P(x_{(k)} \notin R|E_{k-1})P(y = 1|E_k) \quad (3.33)$$

where  $E_k$  denotes the joint event  $x_{(1)}, x_{(2)}, \dots, x_{(k)} \notin R$ . We carry out this decomposition until to reach the point  $k^*$  given by

$$k^* = \max \left\{ k \mid \sum_{k'=k}^l \mathbf{1}(y_{(k')} = 0) \geq n \text{ and } \sum_{k'=k}^l \mathbf{1}(y_{(k')} = 1) \geq n \right\} \quad (3.34)$$

at which the probability of the reference set containing this point equals 1,  $\Pr(x_{(k^*)} \in R|E_{k^*-1}) = 1$  due to the obligation that the reference set must include  $n$  points

from each dataset. The probability of the reference set containing  $x_{(k)}$  under the condition that the reference set does not contain any points before  $x_{(k)}$  can be calculated as

$$P(x_{(k)} \in R | E_{k-1}) = \begin{cases} \frac{n}{l_0^k} & \text{if } y = 0 \\ \frac{n}{l_1^k} & \text{if } y = 1 \end{cases} \quad (3.35)$$

and

$$P(x_{(k)} \notin R | E_{k-1}) = 1 - P(x_{(k)} \in R | E_{k-1}) \quad (3.36)$$

where  $l_0^k$  and  $l_1^k$  indicate the number of samples that belong to  $C_0$  and  $C_1$  beyond the  $k$  nearest points, respectively. Consequently, we used the following algorithm to compute the posterior probability of the true positive dataset  $P(C_1|x)$  for a given compound-protein pair  $x$  based on dataset  $\{x_i, y_i\}, i = 1, 2, \dots, l$  and a  $n$ .

- Compute  $k_i = \mathbf{K}(x, x_i)$
- Sort  $k_i$  so that  $k_{(1)} \geq k_{(2)} \geq \dots \geq k_{(l)}$  and determine corresponding  $x_{(i)}$  and  $y_{(i)}$  by sorted similarities
- Identify  $k^*$  and set  $P(y = L | E_{k^*-1}) = \mathbf{1}(y_{(k^*)} = L)$
- for  $k = k^* - 1, k^* - 2, \dots, 1$ ,
  - compute  $P(y = L | E_k) = P(x_{(k+1)} \in R | E_k) \mathbf{1}(y_{(k+1)} = L) + P(x_{(k+1)} \notin R | E_k) P(y = 1 | E_{k+1})$
- Finally calculate posterior probability
  - $f_0 = P(x_{(1)} \in R) \mathbf{1}(y_{(1)} = 1) + P(x_{(1)} \notin R) P(y = 1 | E_1)$

Another problem with compound-protein interaction data is class imbalance: Since only a small portion of samples are marked as true positive, the number of samples in  $C_0$  is much greater than  $C_1$ . Therefore, we modified the cost function in the QSL algorithm (Eq. 5.7) to find the optimum  $n$  parameter as

$$E(n) = 4 \frac{|C_1|}{|C_0|} \sum_{x_i \in C_0} f_1(x_i) f_0(x_i) + 4 \sum_{x_i \in C_1} f_1(x_i) f_0(x_i) + 2n \quad (3.37)$$



where  $|C_0|$  and  $|C_1|$  denote the number of compound-protein pairs in  $C_0$  and  $C_1$ , respectively.

We adapted the numerical algorithm developed by Karacali (2010) to CPI prediction task in such a way that similarities between pairs are used instead of distances between feature vectors. Note that the most similar pair to a query pair corresponds to its nearest neighbor, in other words, the pair that has the minimum distance in a Euclidean space. This duality allows formulating the nearest neighbor classification and by extension the QSL algorithm in terms of a similarity measure between pairs, which eliminates the need to construct feature vectors for the unstructured chemical and genomics data on which a distance metric is to be defined and calculated for compounds and proteins. The labels of compound-protein pairs in the unknown dataset  $C_0$  are then predicted based on a set of known interactions between a small portion of all possible compound-protein pairs and pairwise similarities between compound-protein pairs. By virtue of the QSL paradigm, only positive interaction information and well-defined similarity measures between chemical and protein data are enough to carry out our proposed method. Nevertheless, one can still embed all compound-protein pairs into a Euclidean space as feature vectors, and distances between vectors can be used: Sorting distances between a query pair and all pairs in an ascending order will be equivalent to sorting similarities in a descending order.

### 3.4. Kolmogorov-Smirnov method

Once estimates of the posterior probability of the true positive dataset  $P(C_1|x_i)$  for samples  $x_i$  in  $C_0$  and  $C_1$  are obtained, the samples in  $C_0$  that would have been labeled as positives had they been tested are expected to exhibit greater posterior probability of belonging to  $C_1$  compared to the actual negatives in  $C_0$ . Such hidden positive samples in  $C_0$  can then be identified as  $x_i \in C_0$  for which  $P(C_1|x_i) > T$  using a suitable threshold  $T$ . Note that in this formulation, the threshold  $T$  draws the boundary of the overlap between the true positive dataset  $C_1$  and the unknown dataset  $C_0$ . We used the Kolmogorov-Smirnov method to determine the optimal posterior probability threshold,  $T$ . To this end, observed posterior probability values of samples from  $C_1$  and  $C_0$   $\{P(C_1|x_1), P(C_1|x_2), P(C_1|x_3) \dots, P(C_1|x_{|C_1|+|C_0|})\}$  are combined in a list and sorted in

an ascending order. Empirical distribution functions of the true positive dataset  $F_{C_1}(t)$  and the unknown dataset  $F_{C_0}(t)$  are calculated separately as

$$F_{C_1}(t) = \frac{1}{|C_1|} \sum_i \mathbf{1}(P(C_1|x_i) < t \text{ for } x_i \in C_1) \quad (3.38)$$

and

$$F_{C_0}(t) = \frac{1}{|C_0|} \sum_i \mathbf{1}(P(C_1|x_i) < t \text{ for } x_i \in C_0) \quad (3.39)$$

for all  $t \in [0,1]$ . Finally, the maximum difference between the empirical cumulative distribution functions of the two sample sets was identified as

$$D_{max} = \max_t |F_{C_1}(t) - F_{C_0}(t)| \quad (3.40)$$

by a line search.

Conventionally, the  $D_{max}$  statistic is used to decide whether samples in two different sets come from the same distribution or not with respect to a statistical significance level [49]. In this study, the posterior probability value at which  $D_{max}$  is observed is used as the optimal threshold that separates the hidden positive samples in  $C_0$  from the rest. We also used the value of  $D_{max}$  as a measure pertaining to the ability of the proposed approach and the associated similarity metrics to separate the hidden positive and the actual negative samples in  $C_0$  for performance comparison purposes between different similarity measures.

## CHAPTER 4

### RESULTS

In this chapter, we first provide an analysis of the proposed methodology regarding its ability to separate the hidden positives from the actual negatives in the unknown dataset using different combinations of compound and protein similarity measures. Then, we present the most likely interactions that are predicted by the proposed method and the current records about these interactions in up-to-date compound-protein interaction databases.

We calculated a total of thirteen compound similarity matrix alternatives and one protein similarity matrix for the compounds and proteins in the Nuclear Receptor dataset and the GPCR dataset separately using the methods described earlier. By applying the quasi-supervised learning algorithm on the resulting thirteen combined compound-protein similarity matrices, we calculated the posterior probability of the true positive dataset for all compound-protein pairs. The quality of the separation between the hidden positives and the actual negatives in the unknown dataset for the thirteen different similarity matrix choices was calculated in terms of the  $D_{max}$  values obtained by Kolmogorov-Smirnov analysis. These values indicate the ability of the proposed framework to contrast the true positive dataset against the unknown dataset, and by extension, how good the interacting and non-interacting compound-protein pair classes are distinguished from each other.

Table 9 presents the  $D_{max}$  values of all techniques with which we calculated the similarity between compounds, as there is only one similarity measure for proteins, for the Nuclear Receptor dataset and the GPCR dataset. The results obtained from the Nuclear Receptor dataset indicate that KCF-S Fingerprints achieves the greatest separation between the compound-protein pairs in the true positive dataset  $C_1$  and the unknown dataset  $C_0$ , followed by Maximum Common Substructure (RDkit) and TF-IDF cosine similarity. The top three techniques that achieved the greatest separation in the GPCR dataset are TF-IDF cosine similarity, LINGO similarity with  $q = 3$ , and LINGO similarity with  $q = 4$ , respectively. The resulting separation between the hidden positive compound-protein pairs and the actual negatives in the unknown dataset is also apparent

in the histograms of the posterior probability of the true positive dataset obtained using these compound similarity methods as shown in Figure 12 and Figure 14. Figure 13 and Figure 15 illustrate the cost functions with which we obtained optimum  $n$  for these compound similarity methods. Figure 15 presents the grid search algorithm for faster convergence while Figure 13 show the one that scans all possible  $n$  values in a predetermined range. The compound-protein pairs for which the posterior probability of the true positive dataset was greater than the indicated threshold were identified as hidden positives representing predicted interactions.

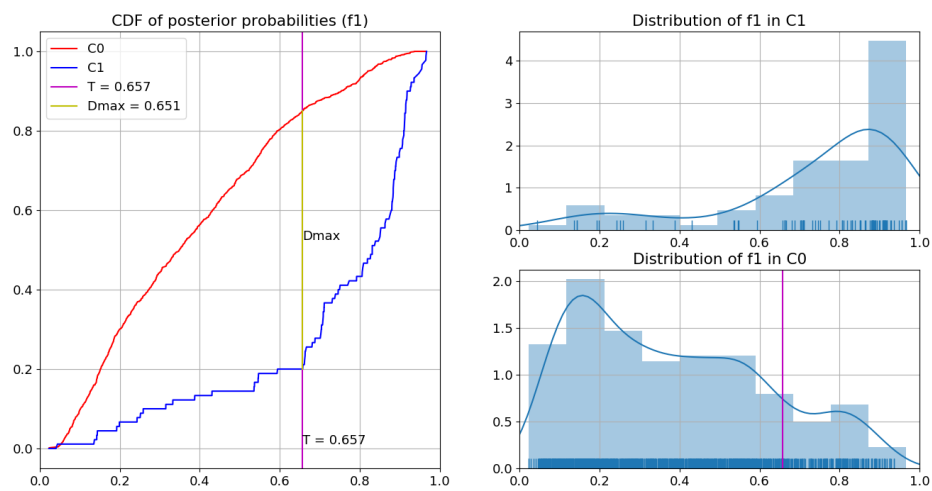
Table 9. Performance comparison of compound similarity measure methods

Compound Similarity Measure Methods	Nuclear Receptors	GPCR	Strategy
	$D_{max}$	$D_{max}$	
Extended-Connectivity Fingerprints-4 (ECFP4)	0.587	0.604	Fingerprint
Functional-Class Fingerprints-4 (FCFP4)	0.512	0.601	Fingerprint
KCF-S Fingerprints (KCF-S)	<b>0.651</b>	0.642	Fingerprint
Molecular ACCess System fingerprints (MACCS)	0.475	0.518	Fingerprint
SIMCOMP	0.622	0.584	Graph
Maximum Common Substructure (MCS) – RDkit	<b>0.647</b>	0.635	Graph
Normalized Longest Common Subsequence (NLCS)	0.543	0.582	SMILES
Combination of LCS Models (CLCS)	0.592	0.584	SMILES
LINGOsim ( $q=3$ )	0.624	<b>0.660</b>	SMILES
LINGOsim ( $q=4$ )	0.630	<b>0.652</b>	SMILES
LINGOsim ( $q=5$ )	0.607	0.629	SMILES
LINGO based TF	0.604	0.646	SMILES
LINGO based TF-IDF	<b>0.645</b>	<b>0.669</b>	SMILES

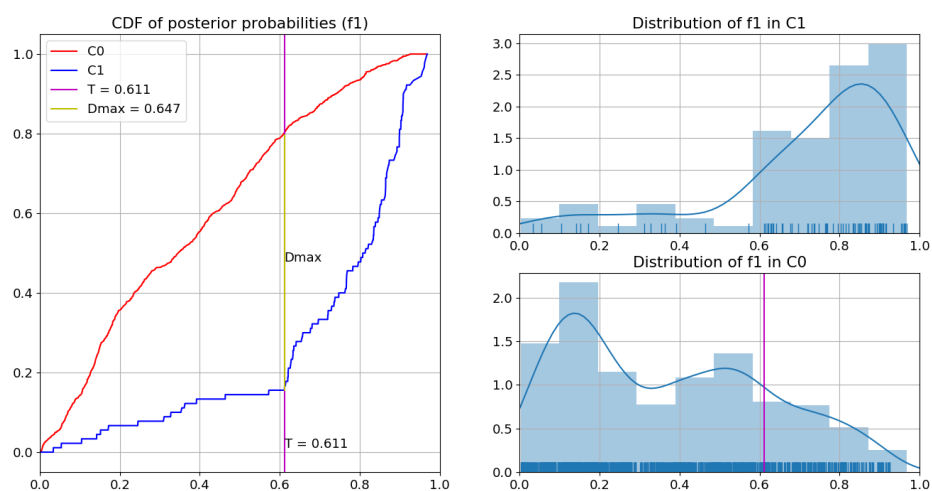
Finally, we constructed two lists of predicted interactions, one for the Nuclear Receptor dataset by taking the intersection of the compound-protein pairs predicted separately by KCF-S, SIMCOMP, MCS, LINGOsim( $q = 3$ ), LINGOsim( $q = 4$ ), LINGOsim( $q = 5$ ), LINGO based TF and LINGO based TF-IDF, and another for the GPCR dataset by taking the intersection of the compound-protein pairs predicted separately by ECFP4, FCFP4, KCF-S, MCS, LINGOsim( $q = 3$ ), LINGOsim( $q = 4$ ), LINGOsim( $q = 5$ ), LINGO based TF and LINGO based TF-IDF for which Kolmogorov-Smirnov Analysis resulted in  $D_{max}$  values greater than 0.6. For each

predicted interaction, we calculated the geometric mean of the posterior probabilities by each method to obtain a unique posterior probability.

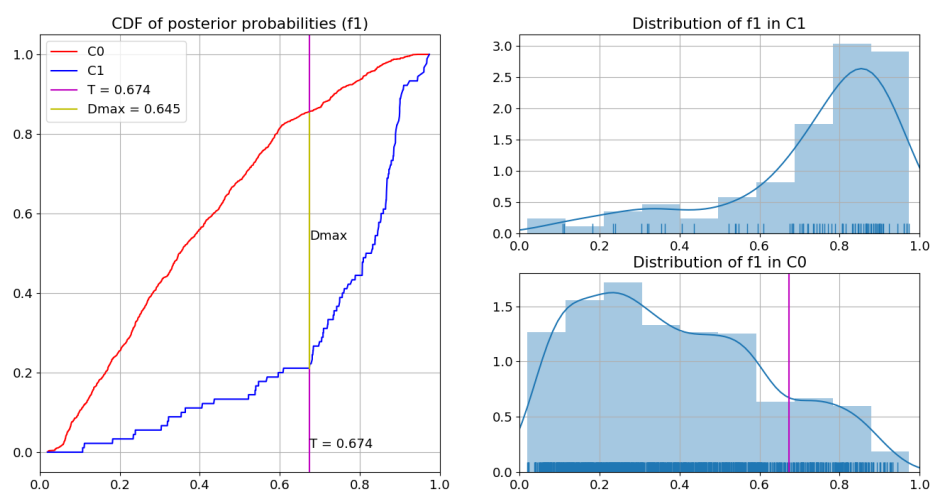
The top forty predicted interactions in both lists are provided in Table 10 and Table 11, respectively, in the descending order of posterior probability of belonging to the set of true interactions along with the current knowledge regarding the predicted interactions in DrugBank (DB) [34], KEGG (KG) [35], SuperTarget (ST) [37], ChEMBL (CH) [50]. Note that since the publication of the Yamanashi dataset in 2008, interactions of some unlabeled pairs in  $C_0$  have been experimentally validated and incorporated in the interaction databases listed above. In Table 10 and Table 11, the pairs that have interaction information in least one dataset were color-coded by green, while the potential interactions suggested by ChEMBL [50] were highlighted by yellow. In the tables, identified positive interactions are indicated by the letter Y and potential interactions are indicated by the letter P, respectively. Note also that a considerable number of predicted interactions are now categorized as positive interaction indicating the success of the proposed approach in identifying unknown true interactions among all possible compound-protein combinations.



**a.**

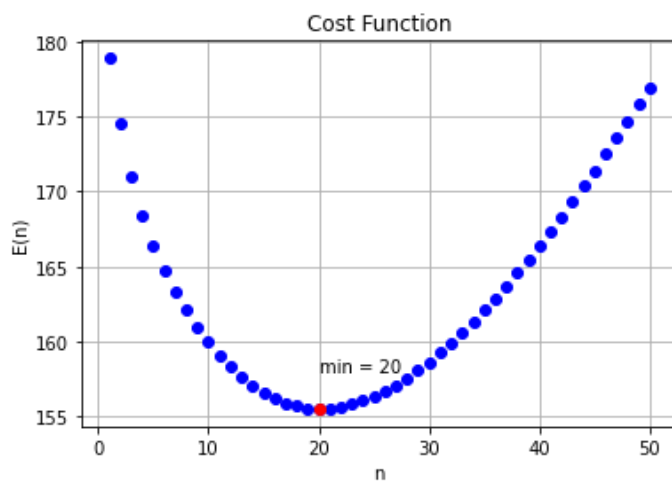


**b.**

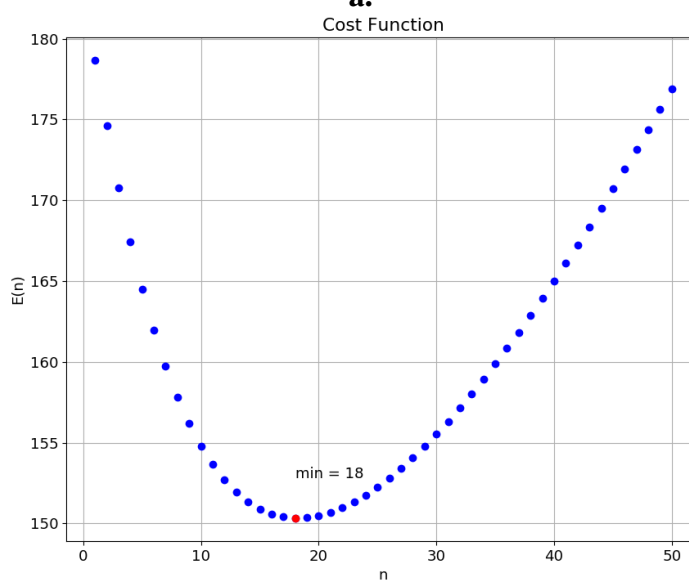


**c.**

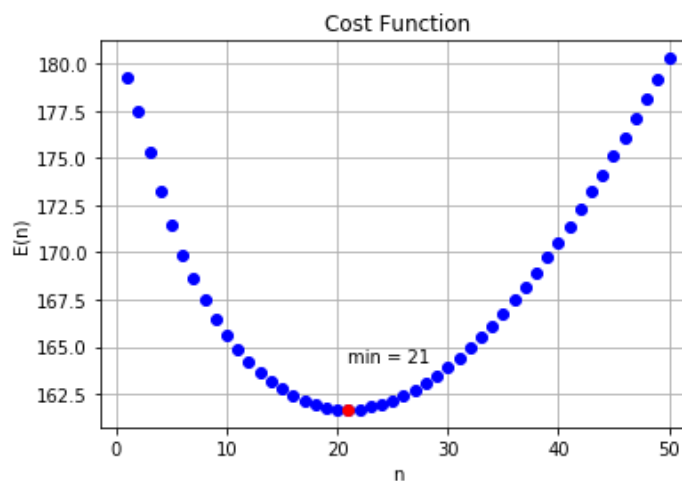
Figure 12. Posterior probability distributions of compound-protein pairs in the Nuclear Receptor dataset to belong to the true positive dataset and their Kolmogorov-Smirnov Analysis for (a) KCF-S Fingerprints, (b) Maximum Common Substructure (RDkit), and (c) LINGO based TF-IDF Cosine Similarity



**a.**

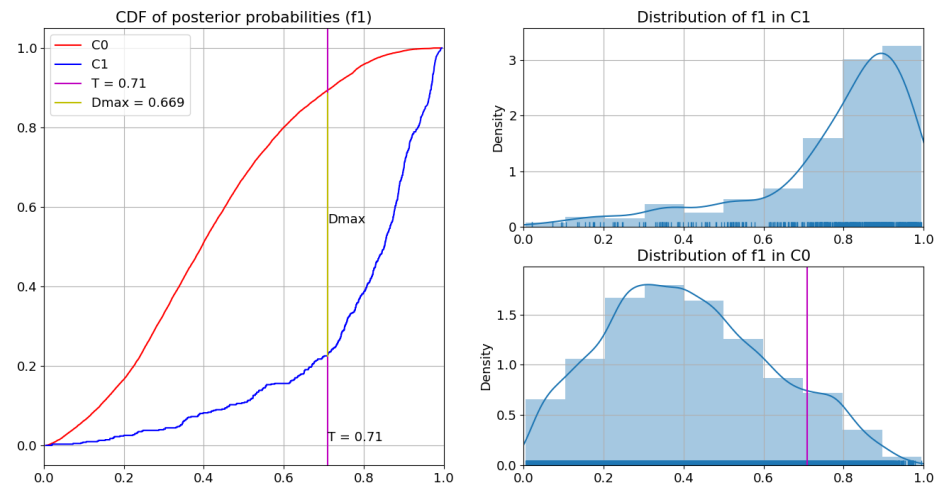


**b.**

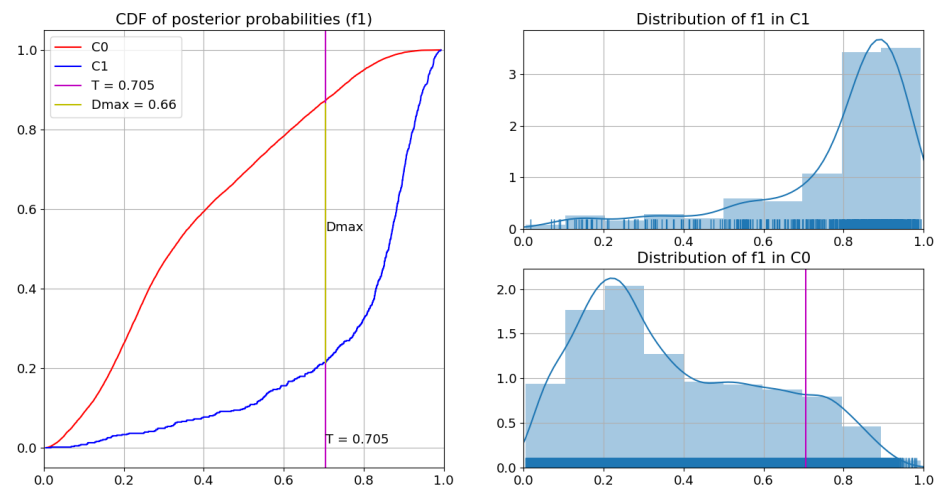


**c.**

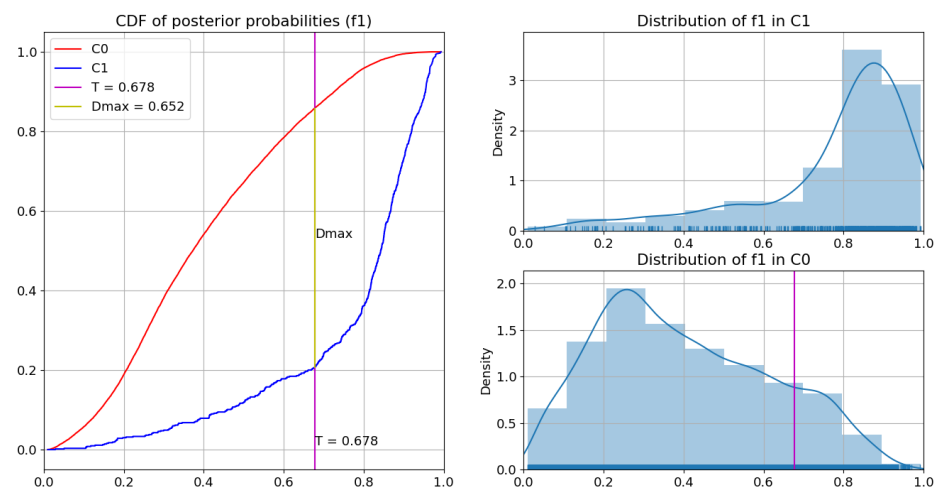
Figure 13. The plot of the cost functions,  $E(n)$  for (a) KCF-S Fingerprints, (b) Maximum Common Substructure (RDkit), and (c) LINGO based TF-IDF Cosine Similarity



**a.**



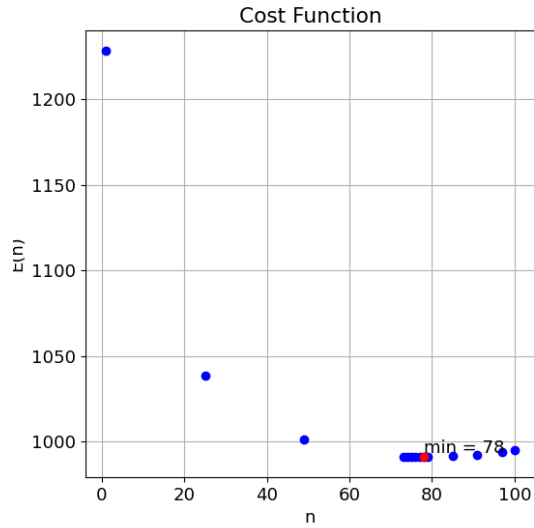
**b.**



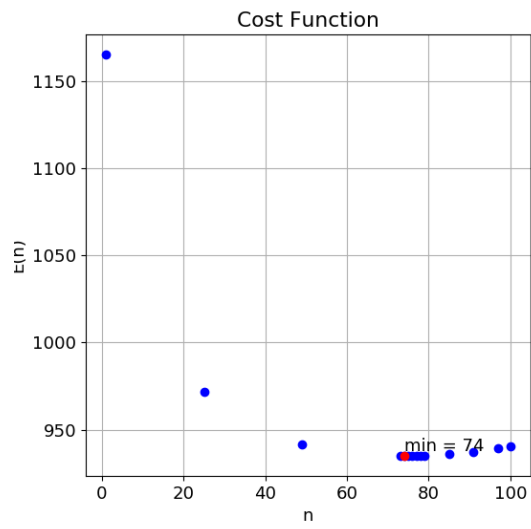
**c.**

Figure 14. Posterior probability distributions of compound-protein pairs in the GPCR dataset to belong to the true positive dataset and their Kolmogorov-Smirnov Analysis for (a) LINGO based TF-IDF Cosine Similarity, (b) LINGOsim ( $q=3$ ), and (c) LINGOsim ( $q=4$ )

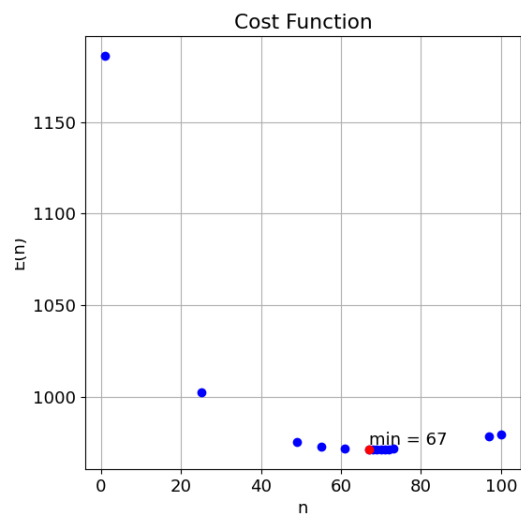




**a.**



**b.**



**c.**

Figure 15. The plot of the cost functions,  $E(n)$  for (a) LINGO based TF-IDF Cosine Similarity, (b) LINGOsim ( $q=3$ ), and (c) LINGOsim ( $q=4$ )

Table 10. The list of top 40 predicted positive interactions in the unknown dataset  $C_0$  for the Nuclear Receptor Dataset

Compound	Protein	Posterior Probability	ST	DB	KG	CH
Nandrolone phenpropionate	estrogen receptor 1	0,92284				
Fluoxymesterone	progesterone receptor	0,92249				
Testosterone	progesterone receptor	0,92149	Y	Y		Y
Hydrocortisone	progesterone receptor	0,91974				P
Norethindrone	estrogen receptor 1	0,91877				
Spironolactone	progesterone receptor	0,91811		Y		Y
Nandrolone phenpropionate	progesterone receptor	0,91511				P
Eplerenone	progesterone receptor	0,91061	Y	Y		
Testosterone	estrogen receptor 1	0,90441	Y	Y		Y
Oxandrolone	progesterone receptor	0,90432				P
Budesonide	progesterone receptor	0,90063				P
Mifepristone	estrogen receptor 1	0,9002	Y			
Loteprednol etabonate	progesterone receptor	0,89722				
Aminonide	progesterone receptor	0,88766				
Isotretinoin	retinoic acid receptor beta	0,88641			Y	Y
Pregnenolone	progesterone receptor	0,88571				P
Isotretinoin	retinoic acid receptor gamma	0,87976	Y	Y	Y	Y
Oxandrolone	estrogen receptor 1	0,87834				
Hydrocortisone	estrogen receptor 1	0,87657				
Dydrogesterone	estrogen receptor 1	0,86944				
Spironolactone	estrogen receptor 1	0,8678				Y
Ethinyl estradiol	progesterone receptor	0,86463				P
Isotretinoin	retinoid X receptor gamma	0,86261	Y			
Chenodiol	progesterone receptor	0,86051				
Tazarotene	estrogen receptor 1	0,85617				
Cholesterol	progesterone receptor	0,85454				P
Eplerenone	estrogen receptor 1	0,85175	Y	Y		
Isotretinoin	retinoid X receptor alpha	0,846				
Etretinate	estrogen receptor 1	0,84411				
Chenodiol	estrogen receptor 1	0,8422				
Medroxyprogesterone acetate	estrogen receptor 1	0,84218		Y		
Pregnenolone	estrogen receptor 1	0,84155				
Mometasone furoate	estrogen receptor 1	0,83937				
Estrone	estrogen receptor 2	0,83929	Y	Y	Y	P
Isotretinoin	retinoid X receptor beta	0,83837	Y			
Estrone	progesterone receptor	0,83666				P
Loteprednol etabonate	estrogen receptor 1	0,83524				
Tazarotene	peroxisome proliferator activated receptor alpha	0,83361				
Tretinoin	RAR related orphan receptor A	0,83319				
Etretinate	peroxisome proliferator activated receptor alpha	0,83081				

Table 11. The list of top 40 predicted positive interactions in the unknown dataset  $C_0$  for the GPCR Dataset

Compound	Protein	Posterior Probability	ST	DB	KG	CH
Isoetharine	adrenoceptor beta 2	0,96806		Y	Y	
Albuterol	adrenoceptor beta 1	0,96794	Y	Y		
Clozapine	dopamine receptor D3	0,96584	Y	Y		Y
Metoprolol	adrenoceptor beta 2	0,9627	Y	Y	Y	
Denopamine	adrenoceptor beta 2	0,95763				P
Levodopa	adrenoceptor beta 2	0,952				
Ritodrine	adrenoceptor beta 1	0,95129		Y		P
Dipivefrin	adrenoceptor beta 2	0,94935		Y	Y	
Epinephrine	adrenoceptor beta 3	0,94265			Y	Y
Methoxamine hydrochloride	adrenoceptor beta 2	0,94259				
Albuterol sulfate	adrenoceptor beta 1	0,9424	Y	Y		P
Levodopa	adrenoceptor beta 1	0,94178				
Methoxamine hydrochloride	adrenoceptor beta 1	0,94074				
Dipivefrin	adrenoceptor beta 1	0,93835			Y	
Bisoprolol	adrenoceptor beta 3	0,93605	Y			
Atenolol	adrenoceptor beta 3	0,93445	Y			
Cicloprolol hydrochloride	adrenoceptor beta 3	0,93416			Y	
Betaxolol hydrochloride	adrenoceptor beta 3	0,93132				
Clozapine	adrenoceptor alpha 2C	0,93129		Y		Y
Chlorpromazine	histamine receptor H1	0,93035		Y	Y	Y
Fenoldopam mesylate	adrenoceptor beta 2	0,92823				
Terbutaline sulfate	adrenoceptor beta 1	0,9279	Y	Y		P
Methixene hydrochloride	histamine receptor H1	0,92739				
Clozapine	cholinergic receptor muscarinic 3	0,9265		Y		
Perphenazine	histamine receptor H1	0,92473				P
Chlorpromazine phenolphthalinate	histamine receptor H1	0,92187		Y	Y	
Chlorpromazine hibenzate	dopamine receptor D2	0,92114		Y	Y	
Oxymetazoline hydrochloride	adrenoceptor beta 2	0,9203				
Albuterol	adrenoceptor beta 3	0,91992	Y			
Mesoridazine	histamine receptor H1	0,91985				
Olanzapine	adrenoceptor alpha 2C	0,91954	Y	Y		Y
Olanzapine	5-hydroxytryptamine receptor 1B	0,91739		Y		
Olanzapine	5-hydroxytryptamine receptor 1D	0,91734		Y		
Clozapine	cholinergic receptor muscarinic 4	0,91727		Y		
Thiethylperazine	dopamine receptor D3	0,91711	Y			Y
Chlorpromazine phenolphthalinate	dopamine receptor D2	0,91527	Y	Y	Y	
Promethazine hydrochloride	cholinergic receptor muscarinic 1	0,91433	Y	Y		
Tamsulosin hydrochloride	adrenoceptor beta 2	0,91393				
Methdilazine	cholinergic receptor muscarinic 1	0,91377				P
Metoclopramide	adrenoceptor alpha 1A	0,91375				

## CHAPTER 5

### CONCLUSION

In this study, we have proposed a quasi-supervised learning approach for compound-protein interaction prediction that addresses the issues associated with the lack of ground-truth negative instances in compound-protein interaction datasets. As mentioned in the literature review, there are very few studies in the literature that address the absence of reliable negatives as well as data imbalance between true positives and unlabeled compound-protein pairs. The present study offers an alternative strategy for an adequate evaluation of unlabeled compound-protein pairs. The results show that the quasi-supervised learning algorithm can make accurate predictions on interaction status of unlabeled compound-protein pairs without requiring an experimentally validated set of true negatives; or compound-protein pairs that have been established not to interact.

The quasi-supervised learning algorithm is well-suited to the compound-protein interaction prediction problem due to two reasons. Firstly, it uses only ground-truth positive compound-protein pairs without making any unrealistic and potentially erroneous presumptions on the interaction status of the unlabeled pairs. Instead, it successfully contrasts the set of all unlabeled compound-protein pairs with no known interaction with the true-positive dataset, and identifies the pairs most likely to interact with each other automatically. Secondly, it can operate on the similarity structure between protein and compound pairs directly without requiring a feature vector representation for either of them, a common requirement for most other machine learning strategies. In this manner, it avoids the issues and shortcomings associated with feature-extraction processes that constitute major challenges especially for unstructured compound and protein data. This also allows incorporating alternative notions of similarity between protein and compound pairs from a larger, non-numeric class of similarity measures and enhances the breadth of the analysis.

On a final note, the proposed methodology can be extended in several ways. First, we applied quasi-supervised learning algorithm on only Nuclear Receptor and GPCR datasets due to computational limitations. The proposed methodology can also be applied on datasets of other common target protein families such as Enzyme and Ion Channels of

the Yamanashi dataset using more powerful computing resources. The Quasi-Supervised Learning algorithm appears particularly suitable for parallelization allowing for wider-scale applications on parallel computation architectures. Furthermore, the combination of different similarity measurement can be tried rather than treating them separately. Apart from this, further research can explore additional similarity measures that reflect the correlation between chemical and genomic spaces for potentially more efficient prediction. For instance, LINGO-like similarity measures for proteins can be explored in terms of protein motifs and domains that may incorporate the established functional characteristics of the proteins into the similarity structure more adequately.

## REFERENCES

- [1] T. I. Engel and J. Gasteiger, “Drug Discovery: An Overview ,” in *Applied Chemoinformatics: Achievements and Future Opportunities*, Germany: Wiley-VCH, 2018.
- [2] J. A. DiMasi, H. G. Grabowski, and R. W. Hansen, “Innovation in the pharmaceutical industry: New estimates of R&D costs,” *Journal of Health Economics*, vol. 47, May 2016, doi: 10.1016/j.jhealeco.2016.01.012.
- [3] “Center for Drug Evaluation and Research. ‘Novel Drug Approvals for 2019.’ U.S. Food and Drug Administration. FDA. <https://www.fda.gov/drugs/new-drugs-fda-cders-new-molecular-entities-and-new-therapeutic-biological-products/novel-drug-approvals-2019> .”
- [4] J. L. Medina-Franco, M. A. Giulianotti, G. S. Welmaker, and R. A. Houghten, “Shifting from the single to the multitarget paradigm in drug discovery,” *Drug Discovery Today*, vol. 18, no. 9–10, May 2013, doi: 10.1016/j.drudis.2013.01.008.
- [5] E. Lounkine *et al.*, “Large-scale prediction and testing of drug activity on side-effect targets,” *Nature*, vol. 486, no. 7403, Jun. 2012, doi: 10.1038/nature11159.
- [6] N. Novac, “Challenges and opportunities of drug repositioning,” *Trends in Pharmacological Sciences*, vol. 34, no. 5, May 2013, doi: 10.1016/j.tips.2013.03.004.
- [7] E. W. Sayers *et al.*, “Database resources of the National Center for Biotechnology Information,” *Nucleic Acids Research*, vol. 47, no. D1, Jan. 2019, doi: 10.1093/nar/gky1069.
- [8] A. L. Hopkins, “Predicting promiscuity,” 2009. [Online]. Available: <http://pubchem.ncbi.nlm.nih.gov>
- [9] A. C. Cheng *et al.*, “Structure-based maximal affinity model predicts small-molecule druggability,” *Nature Biotechnology*, vol. 25, no. 1, Jan. 2007, doi: 10.1038/nbt1273.
- [10] Y. Yamanishi, “Chemogenomic Approaches to Infer Drug–Target Interaction Networks,” 2013. doi: 10.1007/978-1-62703-107-3\_9.

- [11] M. J. Keiser, B. L. Roth, B. N. Armbruster, P. Ernsberger, J. J. Irwin, and B. K. Shoichet, "Relating protein pharmacology by ligand chemistry," *Nature Biotechnology*, vol. 25, no. 2, Feb. 2007, doi: 10.1038/nbt1284.
- [12] H. Ding, I. Takigawa, H. Mamitsuka, and S. Zhu, "Similarity-based machine learning methods for predicting drug-target interactions: A brief review," *Briefings in Bioinformatics*, vol. 15, no. 5, pp. 734–747, May 2013, doi: 10.1093/bib/bbt056.
- [13] T. Cheng, M. Hao, T. Takeda, S. H. Bryant, and Y. Wang, "Large-Scale Prediction of Drug-Target Interaction: a Data-Centric Review," *AAPS Journal*, vol. 19, no. 5, pp. 1264–1275, Sep. 2017, doi: 10.1208/s12248-017-0092-6.
- [14] A. Ezzat, M. Wu, X. L. Li, and C. K. Kwoh, "Computational prediction of drug-target interactions using chemogenomic approaches: An empirical survey," *Briefings in Bioinformatics*, vol. 20, no. 4, pp. 1337–1357, Mar. 2018, doi: 10.1093/bib/bby002.
- [15] B. Karaçali, "Quasi-supervised learning for biomedical data analysis," *Pattern Recognition*, vol. 43, no. 10, pp. 3674–3682, Oct. 2010, doi: 10.1016/j.patcog.2010.04.024.
- [16] T. Engel, "Basic overview of chemoinformatics," *Journal of Chemical Information and Modeling*, vol. 46, no. 6, pp. 2267–2277, Nov. 2006, doi: 10.1021/ci600234z.
- [17] P. Imming, C. Sinning, and A. Meyer, "Drugs, their targets and the nature and number of drug targets," *Nature Reviews Drug Discovery*, vol. 5, no. 10, Oct. 2006, doi: 10.1038/nrd2132.
- [18] G. Maggiora, M. Vogt, D. Stumpfe, and J. Bajorath, "Molecular Similarity in Medicinal Chemistry," *Journal of Medicinal Chemistry*, vol. 57, no. 8, Apr. 2014, doi: 10.1021/jm401411z.
- [19] "Ctfile Formats, MDL Information Systems Inc.: <http://www.mdli.com/downloads/literature/ctfile.pdf>," CA, 1998.
- [20] H. L. Morgan, "The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service.," *Journal of Chemical Documentation*, vol. 5, no. 2, May 1965, doi: 10.1021/c160017a018.

- [21] D. Weininger, A. Weininger, and J. L. Weininger, "SMILES. 2. Algorithm for generation of unique SMILES notation," *Journal of Chemical Information and Modeling*, vol. 29, no. 2, May 1989, doi: 10.1021/ci00062a008.
- [22] "Daylight.com. 2020. Daylight Theory: SMILES. ."
- [23] J. Schwartz, M. Awale, and J.-L. Reymond, "SMIfp (SMILES fingerprint) Chemical Space for Virtual Screening and Visualization of Large Databases of Organic Molecules," *Journal of Chemical Information and Modeling*, vol. 53, no. 8, Aug. 2013, doi: 10.1021/ci400206h.
- [24] W. H. Brooks, W. C. Guida, and K. G. Daniel, "The Significance of Chirality in Drug Design and Development," *Current Topics in Medicinal Chemistry*, vol. 11, no. 7, Apr. 2011, doi: 10.2174/156802611795165098.
- [25] R. Sawada, M. Kotera, and Y. Yamanishi, "Benchmarking a wide range of chemical descriptors for drug-target interaction prediction using a chemogenomic approach," *Molecular Informatics*, vol. 33, no. 11–12. Wiley-VCH Verlag, pp. 719–731, Nov. 24, 2014. doi: 10.1002/minf.201400066.
- [26] B. Rost, "Protein Structure Prediction in 1D, 2D, and 3D," in *Encyclopedia of Computational Chemistry*, Chichester, UK: John Wiley & Sons, Ltd, 2002. doi: 10.1002/0470845015.cpa033m.
- [27] Y. Chen, E. K. Garcia, M. R. Gupta, A. Rahimi, and L. Cazzanti, "Similarity-Based Classification: Concepts and Algorithms," *J. Mach. Learn. Res.*, vol. 10, pp. 747–776, Jun. 2009.
- [28] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, "Prediction of drug-target interaction networks from the integration of chemical and genomic spaces," *Bioinformatics*, vol. 24, no. 13, Jul. 2008, doi: 10.1093/bioinformatics/btn162.
- [29] L. Jacob and J. P. Vert, "Protein-ligand interaction prediction: An improved chemogenomics approach," *Bioinformatics*, vol. 24, no. 19, pp. 2149–2156, Oct. 2008, doi: 10.1093/bioinformatics/btn409.



- [30] T. van Laarhoven, S. B. Nabuurs, and E. Marchiori, “Gaussian interaction profile kernels for predicting drug-target interaction,” *Bioinformatics*, vol. 27, no. 21, pp. 3036–3043, Nov. 2011, doi: 10.1093/bioinformatics/btr500.
- [31] T. van Laarhoven and E. Marchiori, “Predicting Drug-Target Interactions for New Drug Compounds Using a Weighted Nearest Neighbor Profile,” *PLoS ONE*, vol. 8, no. 6, Jun. 2013, doi: 10.1371/journal.pone.0066952.
- [32] M. Gönen, “Predicting drug-target interactions from chemical and genomic kernels using Bayesian matrix factorization,” *Bioinformatics*, vol. 28, no. 18, pp. 2304–2310, Sep. 2012, doi: 10.1093/bioinformatics/bts360.
- [33] X. Zheng, H. Ding, H. Mamitsuka, and S. Zhu, “Collaborative matrix factorization with multiple similarities for predicting drug-target interactions,” Aug. 2013. doi: 10.1145/2487575.2487670.
- [34] D. S. Wishart *et al.*, “DrugBank: a knowledgebase for drugs, drug actions and drug targets,” *Nucleic Acids Research*, vol. 36, no. suppl\_1, Jan. 2008, doi: 10.1093/nar/gkm958.
- [35] M. Kanehisa, “From genomics to chemical genomics: new developments in KEGG,” *Nucleic Acids Research*, vol. 34, no. 90001, Jan. 2006, doi: 10.1093/nar/gkj102.
- [36] I. Schomburg, “BRENDA, the enzyme database: updates and major new developments,” *Nucleic Acids Research*, vol. 32, no. 90001, Jan. 2004, doi: 10.1093/nar/gkh081.
- [37] S. Gunther *et al.*, “SuperTarget and Matador: resources for exploring drug-target relationships,” *Nucleic Acids Research*, vol. 36, no. Database, Dec. 2007, doi: 10.1093/nar/gkm862.
- [38] M. Hattori, Y. Okuno, S. Goto, and M. Kanehisa, “Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways,” *Journal of the American Chemical Society*, vol. 125, no. 39, pp. 11853–11865, Oct. 2003, doi: 10.1021/ja036030u.
- [39] “RDKit: Open-source cheminformatics; <http://www.rdkit.org>.”

- [40] H. Öztürk, E. Ozkirimli, and A. Özgür, “A comparative study of SMILES-based compound similarity functions for drug-target interaction prediction,” *BMC Bioinformatics*, vol. 17, no. 1, 2016, doi: 10.1186/s12859-016-0977-x.
- [41] D. Vidal, M. Thormann, and M. Pons, “LINGO, an efficient holographic text based method to calculate biophysical properties and intermolecular similarities,” *Journal of Chemical Information and Modeling*, vol. 45, no. 2, pp. 386–393, 2005, doi: 10.1021/ci0496797.
- [42] D. Rogers and M. Hahn, “Extended-connectivity fingerprints,” *Journal of Chemical Information and Modeling*, vol. 50, no. 5, pp. 742–754, May 2010, doi: 10.1021/ci100050t.
- [43] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse, “Reoptimization of MDL Keys for Use in Drug Discovery,” *Journal of Chemical Information and Computer Sciences*, vol. 42, no. 6, Nov. 2002, doi: 10.1021/ci010132r.
- [44] M. Kotera *et al.*, “KCF-S: KEGG Chemical Function and Substructure for improved interpretability and prediction in chemical bioinformatics,” *BMC Systems Biology*, vol. 7, 2013, doi: 10.1186/1752-0509-7-S6-S2.
- [45] M. Sato, H. Suetake, and M. Kotera, “KCF-Convoy: Efficient Python package to convert KEGG Chemical Function and Substructure fingerprints,” *bioRxiv*. bioRxiv, Oct. 24, 2018. doi: 10.1101/452383.
- [46] Phillip Compeau and Pavel Pevzner, *Bioinformatics Algorithms: An Active Learning Approach, Chapter 5: How Do We Compare DNA Sequences?*, 2nd edition. La Jolla, CA: Active Learning Publishers, 2015.
- [47] T. F. Smith and M. S. Waterman, “Identification of common molecular subsequences,” *Journal of Molecular Biology*, vol. 147, no. 1, pp. 195–197, 1981, doi: [https://doi.org/10.1016/0022-2836\(81\)90087-5](https://doi.org/10.1016/0022-2836(81)90087-5).
- [48] S. Henikoff and J. G. Henikoff, “Amino acid substitution matrices from protein blocks,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 89, no. 22, 1992, doi: 10.1073/pnas.89.22.10915.
- [49] B. Bagwell, “A journey through flow cytometric immunofluorescence analyses—Finding accurate and robust algorithms that estimate positive fraction

distributions,” *Clinical Immunology Newsletter*, vol. 16, no. 3, Mar. 1996, doi: 10.1016/S0197-1859(00)80002-3.

- [50] A. Gaulton *et al.*, “ChEMBL: a large-scale bioactivity database for drug discovery,” *Nucleic Acids Research*, vol. 40, no. D1, Jan. 2012, doi: 10.1093/nar/gkr777.