

CmpE 492 Final Report

PLIS: Web Based Application for Drug Target Interaction Extraction

Student ID: 2013400090

Student Name : Irmak Kavasoglu

Advisor: Arzucan Özgür

January 8, 2019

Contents

1	Introduction	3
2	Existing Platforms	3
2.1	Glass	3
2.2	UniProt	3
2.3	PubChem	4
2.4	ChEMBL	4
3	Project Aim	5
3.1	User Requirements	5
3.2	Affinity Information Sources	6
4	User Interface Design	7
5	Implementation Details	9
5.1	Used Technologies	9
5.2	Data Collection	9
5.2.1	Finding a Data Set	9
5.2.2	Fetching Detailed Information	9
5.3	Server Side	10
5.3.1	Request Format	10
5.3.2	Querying with Id or Name	10
5.3.3	Preparing a Result	10
5.3.4	Preparing the Interaction List	11
5.3.5	Affinities of Interactions	11

5.4	Client Side	11
5.4.1	Main Screen	11
5.4.2	Results Screen	12
6	Next Steps	15
7	Conclusion	15

1 Introduction

Today's medicine is constantly being improved. Many diseases can be cured with drugs, and these drugs are results of long research and laboratory processes.

The process of creating a new drug is a big challenge. One of the key characteristics of the interaction between a potential drug and its related target is what we call *binding affinity*. Binding affinity is a value that shows us if the drug is interacting with the target or not, and the strength of this interaction. For a drug to be effective, it should have a strong interaction with its target.

The difficulty with coming up with a new drug is that the aforementioned binding affinity is not an easy value to predict. The aim of this project is to create a web based application which provides easy access to drug target interactions, or in a more general sense, protein ligand interactions.

2 Existing Platforms

There are several works about drugs and targets, or proteins and ligands. Our project aims to cover all the information these platforms provide and add more to it. Let's look at existing platforms now.

2.1 Glass

Glass is a widely known website which has a great database containing information about protein and ligand associations. This platform is the most similar one to what we want to achieve. In their own words:

GLASS (GPCR-Ligand Association) database is a manually curated repository for experimentally-validated GPCR-ligand interactions. Along with relevant GPCR and chemical information, GPCR-ligand association data are extracted and integrated into GLASS from literature and public databases. [?]

2.2 UniProt

UniProt is also an important database for us. This platform's database is mostly on proteins and the drug side of our drug target interaction is using this database. We use uniprot's database as our main source of detailed information about proteins. They offer an xml file for any given protein in their database, which contains the information we need. In their own words:

The Universal Protein Resource (UniProt) is a comprehensive resource for protein sequence and annotation data. [1]

2.3 PubChem

PubChem is a key information source for us and many others. Its database contains details of different chemicals, we use PubChem as a source for information on compounds. PubChem has their own ids for ligands, but also accepts other ids such as ChEMBL ids. This flexibility is the main reason why we chose to use their website as our source of information on ligands. They offer ligand information in many forms, including JSON. In their own words:

PubChem is an open chemistry database at the National Institutes of Health (NIH) and it has information on chemical structures, identifiers, chemical and physical properties, biological activities, patents, health, safety, toxicity data, and many others. PubChem contains the largest collection of publicly available chemical information. [2]

2.4 ChEMBL

ChEMBL and PubChem are very similar sources in terms of coverage. They both focus on ligands. PubChem and ChEMBL ids can be converted to each other, but in this project it is easier to fetch data from PubChem and convert ChEMBL ids to PubChem ids, therefore we only use ChEMBL for identifying ligands. In their own words:

ChEMBL is a database of bioactive drug-like small molecules, it contains 2-D structures, calculated properties (e.g. logP, Molecular Weight, Lipinski Parameters, etc.) and abstracted bioactivities (e.g. binding constants, pharmacology and ADMET data). [3]

3 Project Aim

From the previous section, we know that there are multiple databases and platforms working on the same topic. Our web based application will be able to provide the existing information, and even more. The features planned are as follows.

3.1 User Requirements

- User will be able to enter a query to make a search. This query can be either a protein, or a ligand. Aim is to support both names and ids of the proteins and ligand in the search engine.

- User will be able to see the information on the protein or ligand when they query it. This information will contain;

- Name of the ligand or protein
- Id of the ligand or protein
- Smiles string of the ligand
- Protein sequence of the protein
- Gene name of the protein
- Protein sequence length of the protein
- Organism of the protein
- Molecular formula of the ligand
- InChI key of the ligand

- The search result of the query will have links to the related PubMed page.

- The application will display a list of interactions. These interactions will be proteins if the query was a ligand and ligands if the query was a protein.

- Every interaction will have the following information;

- Name of the ligand or protein
- Id of the ligand or protein
- Link to related PubMed page of ligand or protein
- Protein sequence of the protein
- Affinity values list

- Every affinity element in the affinity values list of an interaction will be made of the following elements;

- Affinity value
- Affinity unit
- Affinity type
- Affinity information source
- Link to the information source

3.2 Affinity Information Sources

The previous subsection mentions that the search results will have a list of interaction elements and we will have multiple affinity values for each interaction.

This is because there is not an absolute value for an affinity between a protein and a ligand; there are different values in different databases and there are extra values that are created by deep learning. What matters is the relative values of affinities between different interactions, this would help choose which protein ligand couple is better for a potential drug.

So, there will usually be more than one affinity value defined for a protein ligand couple and our application will present all of these values to the user, along with the source of the value. The affinity values in our application can have one of three sources;

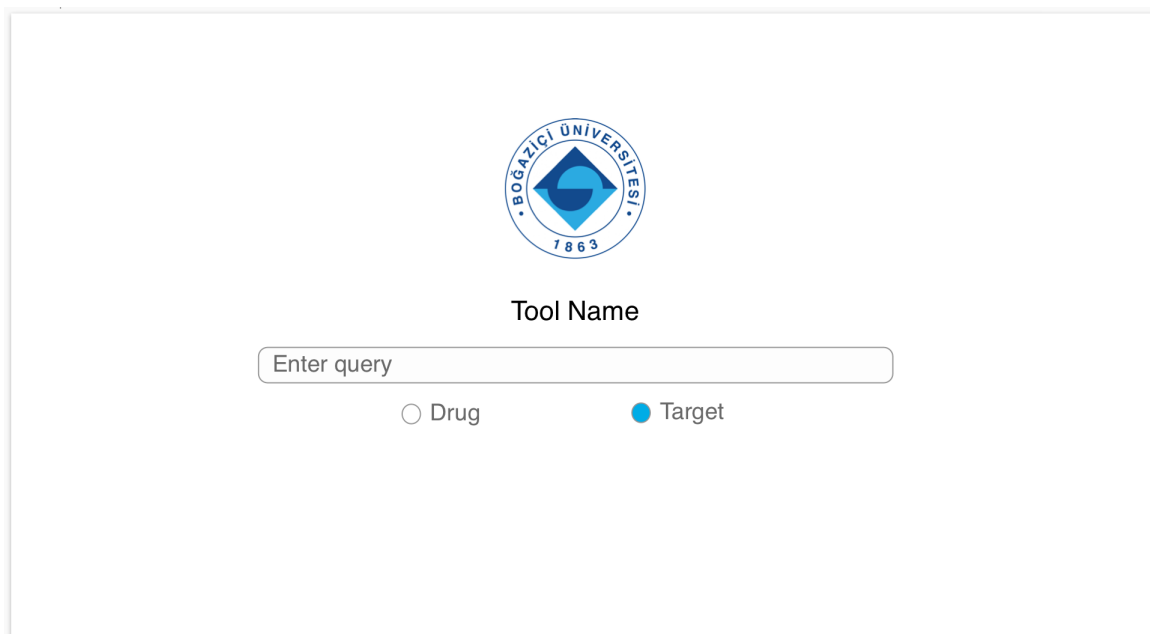
Retrieval First option is the most straightforward way. If we retrieved an existing affinity value from a source like PubMed, we will give a link to the page this information was gathered.

Extraction The second option is through extraction. This method involves text mining the articles containing the protein ligand couple and extracting the affinity value without any database queries to an existing data source.

Prediction The third option is predicting the non-existent affinity value using deep learning techniques. The related paper on topic can be found in references. [4]

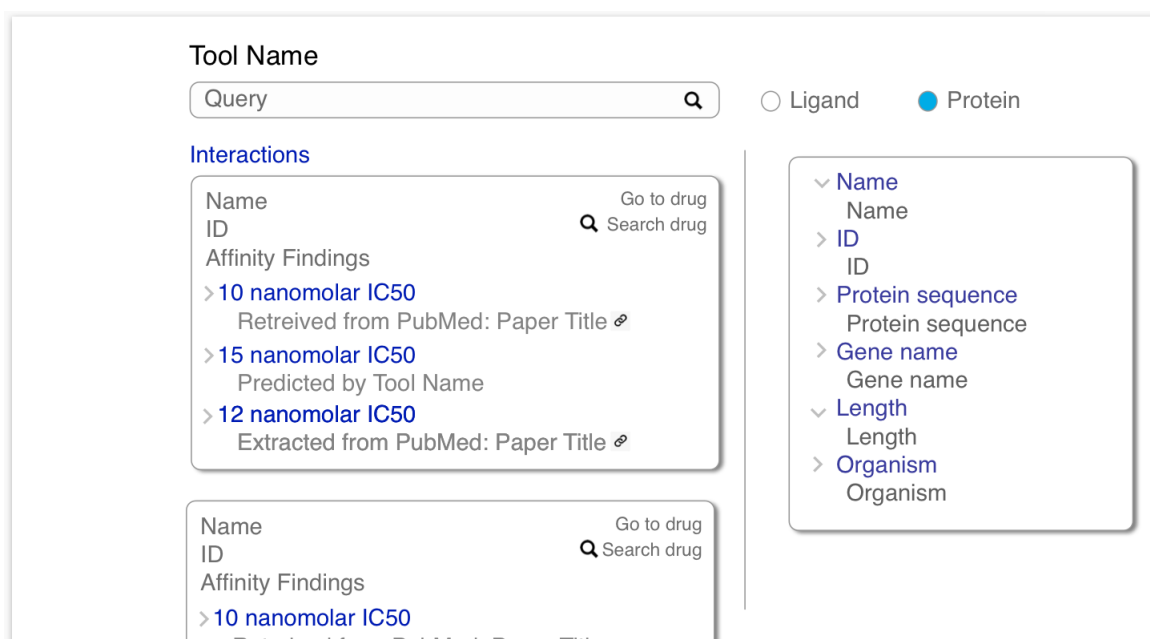
4 User Interface Design

User will interact with our system via a web based application. There will be a simple search box for the main screen where user can select the type of the query, then hit enter.



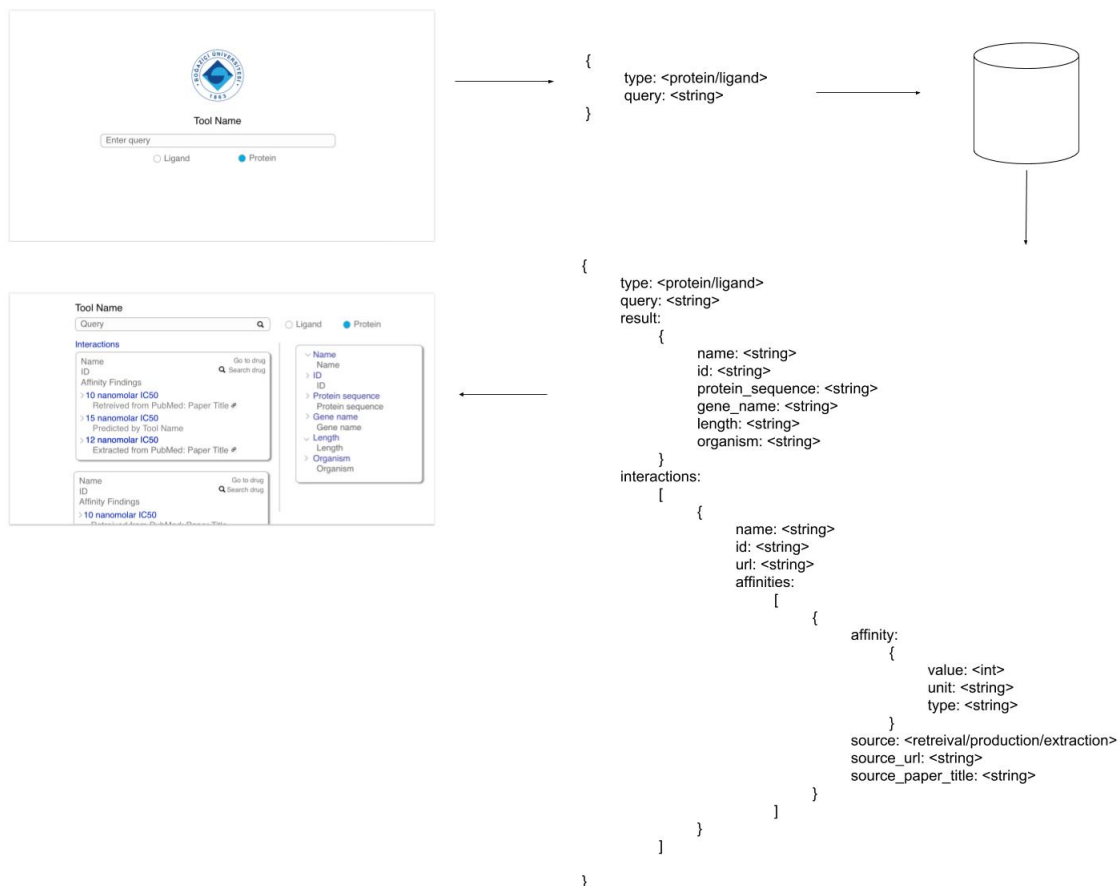
The screenshot shows a web-based application interface. At the top center is the logo of Bogazici University, featuring a blue circular emblem with a stylized 'B' and the text 'BOĞAZICI ÜNİVERSİTESİ' and '1863'. Below the logo, the text 'Tool Name' is displayed. Underneath is a search bar with the placeholder text 'Enter query'. Below the search bar are two radio buttons: 'Drug' (unselected) and 'Target' (selected, indicated by a blue dot).

The results page will have the detailed information explained in the user requirements section. The result of the query is in the right and the interactions are on the left.



The screenshot shows a results page layout. At the top, there is a search bar labeled 'Query' with a magnifying glass icon, and two radio buttons: 'Ligand' (unselected) and 'Protein' (selected, indicated by a blue dot). Below the search bar, the page is divided into two main sections. On the left, under the heading 'Interactions', there are two identical boxes. Each box contains the following text: 'Name', 'ID', 'Affinity Findings', '> 10 nanomolar IC50', 'Retreived from PubMed: Paper Title', '> 15 nanomolar IC50', 'Predicted by Tool Name', '> 12 nanomolar IC50', and 'Extracted from PubMed: Paper Title'. To the right of each box is a 'Go to drug' link with a magnifying glass icon and the text 'Search drug'. On the right side of the page, there is a vertical list of expandable/collapsible items: 'Name' (expanded, showing 'Name'), 'ID' (expanded, showing 'ID'), 'Protein sequence' (expanded, showing 'Protein sequence'), 'Gene name' (expanded, showing 'Gene name'), 'Length' (expanded, showing 'Length'), and 'Organism' (expanded, showing 'Organism').

The website will talk to a server which will establish the connection between the existing databases and new techniques. The planned skeleton of the communication is as follows.



5 Implementation Details

This section focuses on the implementation details of this project. This project had both server-side and client-side implementation, as well as data collection for database. You can check the code out from GitHub. [9]

5.1 Used Technologies

For the website; Sketch was used for design, Sublime Text and VSCode were used as development environments, and HTML, CSS, JavaScript and ReactJS were used as languages to create the client side. For the server; IntelliJ was the main development environment and Java was used as the server side programming language. Communication between two were constructed with Fetch API in client side and HttpServer in server side, their messages were in JSON format. Database was constructed as JSON objects and the information was fetched from APIs of PubChem and UniProt where available, and just webpage parsing where needed.

5.2 Data Collection

5.2.1 Finding a Data Set

The first ever thing to do was to have a data set that we can work on. The websites in the existing platforms section do have databases but for the beginning, our project needed a smaller data set.

We had two options for such data sets. One of these options is the Kiba [5] data set, and the other is the Davis [6] data set. We first fetched these data sets [7] [8], and examined which one would be a better fit for our needs.

A small experiment showed that Kiba is the better one. Both of these data sets had proteins, ligands and an affinity matrix. The problem with Davis was that the keys for protein dataset were not actually id's of proteins, they were the gene names of the proteins. Therefore, we decided to work on Kiba.

5.2.2 Fetching Detailed Information

The Kiba data set only provided us with protein ids, ligand ids and an affinity list between these. Our aim for the website was also providing detailed information about proteins and ligands, therefore we needed to fetch more information on these. The first ever thing that our server does is to fetch this information if it is missing.

Prepare data set This part has four parts. The first three is to import ids of proteins, ids of ligands and the affinity matrix. The fourth part is to create an interactions object. The affinity matrix either has the affinity value between a protein and a ligand, or has 'nan' instead. The interactions object contains only the values from this matrix in a more organized way.

Prepare data set details This part is the part where we actually get some new data. For both ligands and proteins.

Ligands For ligands, we have their ChEMBL ids. PubChem allows us to use a ChEMBL id to navigate to a ligand’s webpage, but it requires PubChem id to fetch the JSON file. So the thing to do was to convert the ChEMBL id to PubChem id.

Initially, we ping the PubChem website and request the webpage with ChEMBL id. The response has a full html file with all the metadata. Among the metadata, we have the "og:url" data. The og tag in metadata has the purpose of serving the summary, image and link when the webpage in question is quoted somewhere. To our luck, the og:url metadata had the same url, but with the PubChem id instead of ChEMBL id. The server extracts this id and uses it to fetch the JSON from PubChem.

Proteins The proteins in our dataset have UniProt ids and the UniProt website allows us to use this id to fetch information about the protein in question. The only problem here was the format: UniProt does not provide us a JSON format, so the server fetches xml format instead and converts it to JSON afterwards.

5.3 Server Side

As mentioned before, the server side is implemented in Java, using IntelliJ IDE. The first thing server does is to import all the data we fetched previously, using methods from the previous section. Then it starts the server and handles requests.

5.3.1 Request Format

A request that arrives in server is thought to have two parameters. The *query* and the *query type*. The former is what user has entered the search box and the latter is either protein, or ligand.

5.3.2 Querying with Id or Name

The request will have a query and this query can be directly the id of the searched protein or ligand, or it can be the name of it. To provide this flexibility, the server prepares an inverted index and for every id, fetches the possible names for that element. Then, these names are linked to the related ids, so when user queries with a name, server first converts it to an id, and then actually handles the query.

5.3.3 Preparing a Result

The response will contain two main things; the result to queried item and an interaction list. To create the result, depending on the query type, a protein is retrieved from the protein database or a ligand is retrieved from the ligand database. Since this data is too large, it is simplified by selecting only necessary things and eliminating the rest before sending it back to the website. This way, both the website will

not be overloaded, and the server will keep the detailed data for further use whenever necessary.

5.3.4 Preparing the Interaction List

The second thing the response needs is the interaction list. Every interaction on this list has 2 parts; the result for the interaction and the affinity values. The result of the interaction item is retrieved the same way as a query would be retrieved and simplified similarly. The affinities of the interaction item is a bit more tricky and explained in next section.

5.3.5 Affinities of Interactions

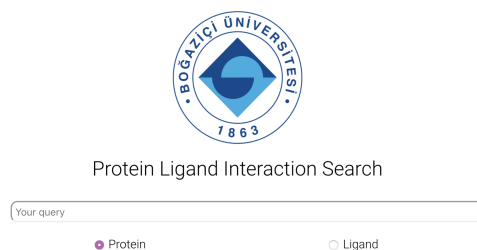
Every interaction triggers three checks. These checks are for the affinity values that can be from retrieval (the Kiba database), from extraction (the text mining part of our project), or from prediction (the machine learning part of our project). The values and sources of the affinities are added to the interaction before adding the interaction to interaction list of the response.

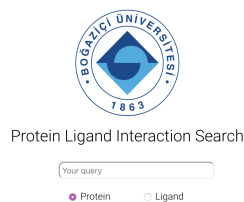
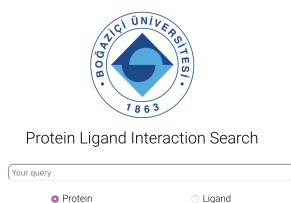
5.4 Client Side

The client side was initially implemented with Vanilla JavaScript but later on migrated to ReactJS for a more flexible and convenient codebase.

5.4.1 Main Screen

The implementation of the website was done as closely to design as possible. Please find several screenshots of the main screen below. The website was made to fit different screen sizes nicely, you can see how the query type buttons and search bar change positions depending on screen size.





In the main screen user can select the query type by using the buttons below search bar. Initially, the protein button will come selected by default. Then user can enter their query to the search bar and hit enter.

5.4.2 Results Screen

The main page will send the query to server and receive back a response, and with that response, the results screen is created. Please find a screenshot below.

Protein Ligand Interaction Search

Search: ☐ Protein ☒ Ligand

Interactions

- Name:** KGP2_HUMAN
Id: Q13237
Affinity Findings:
 ✓ 11.1 nanomolar IC50
 The source of this value is retrieval.
 ✓ 19.83600586505512 nanomolar IC50
 The source of this value is prediction.
- Name:** FGFR2_HUMAN
Id: P21802
Affinity Findings:
 > 11.1 nanomolar IC50
 > 19.101137578174075 nanomolar IC50
- Name:** KPCT_HUMAN
Id: Q04759
Affinity Findings:
 > 11.1 nanomolar IC50
 > 14.239734045063729 nanomolar IC50
- Name:** FER_HUMAN
Id: P16591
Affinity Findings:
 > 11.1 nanomolar IC50
 > 11.753476318126492 nanomolar IC50

Ligand Details:

- Name:** > Ilorasertib
- Id:** > CHEMBL1980297
- InChI:** > InChI=1S/C25H21FN6O2S/c26-17-2-1-3-19(10-17)31-25(34)30-18-6-4...
- IUPAC Name:** > 1-[4-[4-amino-7-[1-(2-hydroxyethyl)pyrazol-4-yl]thieno[3,2-c]pyridin-3-yl]phenyl]-3-(3-fluorophenyl)urea
- InChI Key:** > WPHKIQPVVPYJNAX-UHFFFAOYSA-N
- Canonical SMILES:** > C1=CC(=CC(=C1)F)NC(=O)NC2=CC=C(C(=C2)C3=CSC4=C3C(=NC=C4C5=CN(N=C5)CCO)N

The above screenshot was taken from a ligand search. The search was made using the id of the ligand. On the right side of the screen, we can see the details of this ligand, its id, name, and different chemical names. Since the chemical names can be long, these values arrive truncated, and if user wants to see the full version, they can click on the little arrow next to the values to expand and see the full text.

On the left side, we see the interaction list. Every interaction is a box; since chemical terms can get really long and not-user-friendly, the boxes are an easy way to differentiate between interactions. Every interaction has their name and id, along with an affinity findings list.

The affinities are shown with only their values initially, and user can click on the left arrow to expand and get more information about the affinity. When expanded, user will see the source of the affinity such as text mining, retrieval or prediction.

Protein Ligand Interaction Search

Protein kinase C theta type ☒ Protein ☐ Ligand

Interactions

Name: Ilorasertib
Id: CHEMBL1980297
Affinity Findings
> 11.1 nanomolar IC50
> 17.22543168260695 nanomolar IC50

Name: KNARVZUULIBGV-UHFFFAOYSA-N
Id: CHEMBL1982476
Affinity Findings
> 11.1 nanomolar IC50
> 15.044904665323013 nanomolar IC50

Name: 5-(5,6-Dimethoxybenzimidazol-1-yl)-3-[[2-(trifluoromethyl)phenyl]methoxy]thiophene-2-carboxamide
Id: CHEMBL514499
Affinity Findings
> 13.1 nanomolar IC50
> 11.302750411716897 nanomolar IC50

Name: 3-Methyl-4-oxo-4,7-dihydroisothiazolo[5,4-b]pyridine-5-carboxylic acid
Id: CHEMBL2002553
Affinity Findings
> 12.1 nanomolar IC50
> 19.33104709566537 nanomolar IC50

Name: FXXMEMJTLQATRB-UHFFFAOYSA-N
Id: CHEMBL2001228

Protein

> KPCT_HUMAN

Id
> Q04759

Organism
> Homo sapiens

Gene
> PRKCQ

Length
> 706

Sequence
MSPFLRIGLSNFDGSCQSCQGEAVNPYCAVLVKEYVESENGQMVIQKKPT
MYPPWDSTFDHINKGRVMQIVKGNVDLSETTVELYSLAERCCKNNGK
TEIWLELKPOGRMLMNAFYLEMSDTKDMNEFETEGFFALHQRGAIKQA
KVHHVKCHEFTATFFPQPTFCVCHFEVWGLNKOGYQCRCQNAAIHKKI
DKVIAKCTGSAINRETMFHKERFKIDMPHRFKVYNYKSPTFCEHCGTLLW
GLARQGLKCDACGMNVHHRCTKVANLCGINQKLMAEALAMISTQOAR
CLRDTEQIFREGPVEIGLPCSIKNEARPPCLPTPGKREPOGISWESPLDEVK
MCHLPEPELNKERPSLQIKLIEDFILHMKLGKSGFKVFLAEFKTNQFFAI
KALKKDVVLMDDVVECTMVEKRVLSLAWHEPFLTHMCTFQTKENLFFVM
EYLDGGDLMYHIQSCHKFDLSRATFYAAEILGLOFLHSGKIVYRDLDNILL
DKDGHKIDFGMCKENMLGDAKTNTFCGTPDYIAPEILLGQKYNHSDVW
WSFGVLLYEMIGQSPFHGQDEELFHSIRMDNPFYPRWLEKADLLVCLF
VREPEKRLGVRGDIRQHPLFREINWEELEKEIDPPFRPKVKSPFDCSNFDK
EFLNEKPRLSFADRALINSMQNMFRNFSFMNPGMERLIS

> Protein kinase C theta type

Above is another screenshot from the results page. This time we see a protein search, and instead of id, this time there is a search with the name of the protein. Interactions part is the same, only the query result on the right has different content. As demonstrated, the website will display any number of details about the queried item.

As the last screenshot of this section, we want to show that the results screen is also flexible with different screen sizes and can adapt to narrow screens as well.



Protein Ligand Interaction Search

Protein kinase C theta type

☒ Protein ☐ Ligand

Interactions

Name: Ilorasertib

Id: ChEMBL1980297

Affinity Findings

✓ [11.1 nanomolar IC50](#)

The source of this value is retrieval.

✓ [17.22543168260695 nanomolar IC50](#)

The source of this value is prediction.

Name: KNARVZUUCLIBGV-UHFFFAOYSA-N

Id: ChEMBL1982476

Affinity Findings

> [11.1 nanomolar IC50](#)

> [15.044904665323013 nanomolar IC50](#)

Name: 5-(5,6-Dimethoxybenzimidazol-1-yl)-3-[[2-(trifluoromethyl)phenyl]methoxy]thiophene-2-carboxamide

Id: ChEMBL514499

Affinity Findings

> [13.1 nanomolar IC50](#)

> [11.302750411716897 nanomolar IC50](#)

Name: 3-Methyl-4-oxo-4,7-dihydroisothiazolo[5,4-b]pyridine-5-carboxylic acid

Id: ChEMBL2002553

Affinity Findings

Name

> KPCT_HUMAN

Id

> Q04759

Organism

> Homo sapiens

Gene

> PRKCQ

Length

> 706

Sequence

> MSPFLRIGLSNFDGSCQSCQGEAVN...

Protein

> Protein kinase C theta type

6 Next Steps

This project works with a limited data set currently. As a next step, we can expand our data coverage. The website supports the affinity findings from text mining and prediction; but these are not yet completed studies and are being improved by other members of this project. The importing of the newly found data to the existing database would make this tool a lot more useful.

7 Conclusion

Drug target interaction topic is a very important topic for research, because the results of this area saves many lives. Drug discovery process can take many years and we hope that our project can help speed up this process by providing users with different and new affinity findings.

References

- [1] UniProt: <https://www.uniprot.org/help/about>
- [2] PubChem: <https://pubchemdocs.ncbi.nlm.nih.gov/about>
- [3] ChEMBL: <https://www.ebi.ac.uk/chembl/about>
- [4] DeepDTA: deep drug target binding affinity prediction, Bioinformatics, 34, 2018, i821i829, Hakime Ozturk, Arzucan Ozgur, Elif Ozkirimli.
- [5] Kiba: Tang J.et al. . (2014) Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. J. Chem. Inf. Model. , 54, 735743.
- [6] Davis:Davis M.I.et al. . (2011) Comprehensive analysis of kinase inhibitor selectivity. Nat. Biotechnol. , 29, 10461051.
- [7] : <https://github.com/hkmztrk/DeepDTA/tree/master/data/kiba>
- [8] : <https://github.com/hkmztrk/DeepDTA/tree/master/data/davis>
- [9] : <https://github.com/theTytoAlba/plis>