A WEB REPOSITORY SYSTEM FOR DATA MINING IN DRUG DISCOVERY

Jiali Tang, Jack Wang and Ahmad Reza Hadaegh

Department of Computer Science and Information System, California State University San Marcos, San Marcos, USA

ABSTRACT

This project is to produce a repository database system of drugs, drug features (properties), and drug targets where data can be mined and analyzed. Drug targets are different proteins that drugs try to bind to stop the activities of the protein. Users can utilize the database to mine useful data to predict the specific chemical properties that will have the relative efficacy of a specific target and the coefficient for each chemical property. This database system can be equipped with different data mining approaches/algorithms such as linear, non-linear, and classification types of data modelling. The data models have enhanced with the Genetic Evolution (GE) algorithms. This paper discusses implementation with the linear data models such as Multiple Linear Regression (MLR), Partial Least Square Regression (PLSR), and Support Vector Machine (SVM).

KEYWORDS

Data Mining, Drug Discovery, Drug Description, Chemoinformatics, and Web Application

1. INTRODUCTION

Data mining is the process of extracting data, analyzing it from many dimensions and perspectives, and then producing a summary of the information in a useful form that identifies relationships within the data. There are two types of data mining: descriptive, and predictive data mining. Descriptive data mining gives information about existing data. Predictive data mining makes forecasts based on the data [15]. This project performs the predictive approach by training and testing a series of predictive models on a provided matrix of descriptor values, which describe the chemical properties of a list of drug compounds. The rows in the matrix represent the data associated with each specific compound and the columns represent the descriptor values associated with each common property of all the compounds. The prediction criteria are pIC50 values, the negative logarithms of the compounds' IC50 values, which represent the compound/substance concentration required for 50% inhibition of the compounds' intended targets. Mathematically, we can view this as follows:

$$Y = \beta X + c \text{ which is equal to } Y = \beta 1X1 + \beta 2X2 + \beta 3X3 + \beta 4X4 + \ldots\ldots + \beta nXn + c$$

Where Y refers to prediction criteria, β refers to the model's coefficients, X refers to the values of select descriptors, and "c" refers to prediction error between βX and Y. PIC50 is the negative log(IC50). Thus, the larger the value of the pIC50, and by extension the lower value of the IC50, the more potent the compound is. In this project, the predictive models were generated using genetic evolution algorithms: Genetic Algorithm (GA), Differential Evolution (DE), Binary Particle Swarm Optimization (BPSO) and hybrid form of DE with BPSO (DE-BPSO) [1-14].

The model coefficients β are calculated based on correlating the compounds' descriptor values X with their pIC50 and ends up with a property set. To train the models, we utilized linear machine learning algorithms such as Partial Least-Squares Regression [14], Support Vector Machines Regression, and Multiple Linear Regression [13], along with Multi-Layer Perceptron neural networks.

This step will identify the models that most accurately predict both high- and low-efficacy compounds and display the properties of a compound that are most useful in prediction. Using Data Mining with Genetic Evolutionary algorithms will allow users to build high-quality predictive models for use in drug discovery.

2. RELATED WORKS

The work of Zhong et al. [16] concentrates on the role of Artificial Intelligence (AI) on Drug Discovery. Drug discovery processes have successfully applied Computer-Aided Drug Design (CADD) techniques at certain stages to reduce development costs and risks for preclinical and clinical trials. However, according to Zhong, the decision logic of AI-based models is still difficult to explain. Our modelling techniques do not have any AI flavour, it is simply based on Quantitative Structures Analysis Regression (QSAR) modelling.

Another project by Varsou et al [17] used several representative case studies from drug discovery and computational toxicology to develop a chemo informatics platform, Enalos Suite, using open source software. Enalos Suite (http://enalossuite.novamechanics.com/) was designed and developed as a tool to address a variety of cheminformatics problems; expedite tasks performed in predictive modelling; and allow access, data mining and manipulation for multiple chemical databases.

Enalos Suite allows for user extension and customization to better tailor its functionality for the user's particular field of interest: Nano informatics, biomedical applications, etc. One of the major differences in our work with Varsou's work is that we have also used Genetic Evolutionary techniques to enhance the training of the models.

Other works include "Data Mining and Computational Modelling of High-Throughput Screening Datasets" that is done by Ekins [18], "Web-based Drug Repurposing Tools" that is written be Sam E and Athri P [19], "DRUG Discovery Using Data Mining" that is provided by Charanpreet Kaur and Shweta Bhardwaj [20], and "Using Genetic Algorithms for Data Mining Optimization in an Educational Web-Based System" that is written by Behrouz Minaei-Bidgoli et al. [21]. Some of these papers may provide part of the Data Mining functionalities we present in our work, but none are equipped with a strong backend database as we offer in our work.

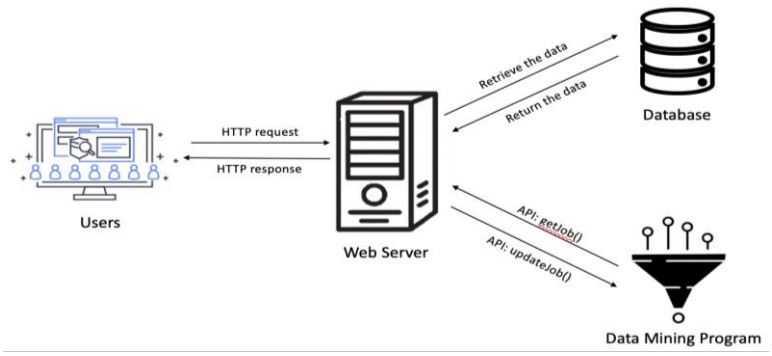

Figure 1: Interaction of the user with data in the database and the programs

3. SYSTEM REQUIREMENTS

Users can utilize this database system by simply uploading their data files and creating data mining tasks. The users can be university professors, students, scientists or researchers.

Users who create accounts can access more extensive data management applications such as editing and deleting their data from the database. They can also modify and cancel their data mining tests.

- **Manage Account Information:** This includes "creating an account", "finding password", and "editing account information (e.g. Name, Email, Password)
- **Manage Datasets:** This includes **"**uploading/editing/deleting datasets", listing all the datasets uploaded by the user", "setting configuration about type, disease, target of the data", "checking the information about datasets" and "searching/downloading datasets.
- **Manage Data Mining Tasks:** This includes **"**creating/editing/deleting data mining tasks", "setting configuration about disease", "targeting, modeling, and algorithm of the tasks, "listing all the data mining tasks owned by a user", "checking the data mining tasks progress information", "searching/sorting tasks by date", "name, disease, target, model, algorithm, dataset" and "downloading the results of tasks".

4. SYSTEM OVERVIEW AND DESIGN

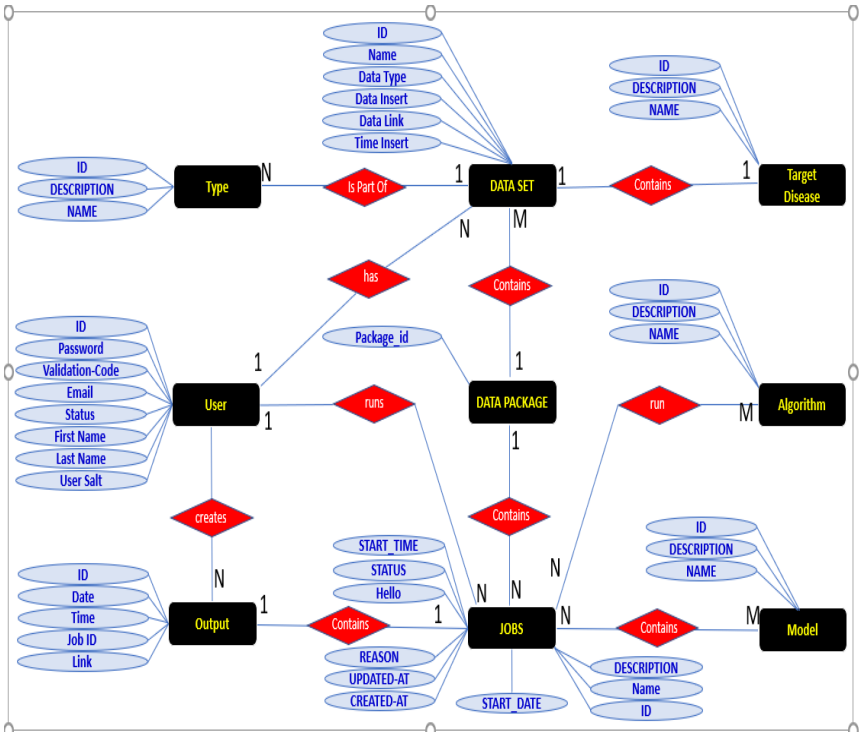

Figure 2: Entity Relationship Diagram

4.1 OVERVIEW

The "CSUSM Chemoinfo Drug Discovery Database System" was developed in PHP and is hosted by an AWS LightSail server that mounts Apache Web Service, MySQL and PHP. User information is stored in a MySQL database. All the data files uploaded by users and the results

from data mining are stored in the AWS LightSail [23] server for now. This database system communicates via custom API with an external data mining program to execute drug discovery. This data mining program was developed in Python by CSUSM's Computer Science Department and is currently hosted in a high-performance AWS EC2 server [24] with two CPUs, 4 GiB memory, and up to 10 Gigabit network performance.

4.2 DESIGN

4.2.1 DATABASE DESIGN

USERS: (user_id, user_password, user_firstname, user_lastname, user_email, user_salt, user_validation_code, user_status)

user_password is encrypted with user_salt. When a user registers an account, the system will send a link to the user to activate their account. This link is created with the user_validation_code and user_status represents the account is active or not.

DATA: (data_id, data_date_insert, data_time_insert, data_upload_by_user, data_user_id, data_name_by_user, data_link, data_target_id, data_disease_id, data_type)

The data file uploaded by the user will be saved in the server, and the address of the file will be saved as data_link. Data_type will record what kind of data this file is, there are three types: descriptors, target, and labels.

JOBS: (job_id, job_updated_at, job_created_at ,job_start_date, job_start_time, job_status, job_user_id, job_name_by_user, job_model_id, job_algorithm_id, job_reason, job_des, data_link, job_attempts, job_queue, job_payload)

Job_status represents the process of the job, and job_reason is to explain the reason or comment if the job is failed to execute or validate, data_link is to the path of the result.

DATAPACKAGE (package_id, job_id, data_id)

This table shows the relationship between data and jobs. A job has multiple data files and a data file can be used for multiple jobs.

OUTPUT: (output_id, output_date, output_time, output_job_id, output_user_id, output_link)

Output_link will save the address where the results are saved in the server, this record will be updated after a job finishes executing.

TYPE: (type_id, type_name, type_description)

This is the data type. There are three data types: descriptor value, target value, and label.

ALGORITHM: (algorithm_id, algorithm_name, algorithm_description)

An algorithm refers to a data mining algorithm that a user chooses for the execution.

MODEL: (model_id, model_name, model_Decription)

A model refers to a data mining model that a user chooses for the execution. For example, MLR, refers to Multiple Linear Regression.

DISEASE: (disease_id, disease_name, disease_description)

An example of the disease is Alzheimer's.

TARGET: (target_id, target_name, target_description, target_disease_id)

A target refers to the value of the PCI50s that we explained above.

4.2.2.PROCESS FLOWCHART

The project's process flow is shown in Figure 3. It shows the relationship between the data, user and the system.

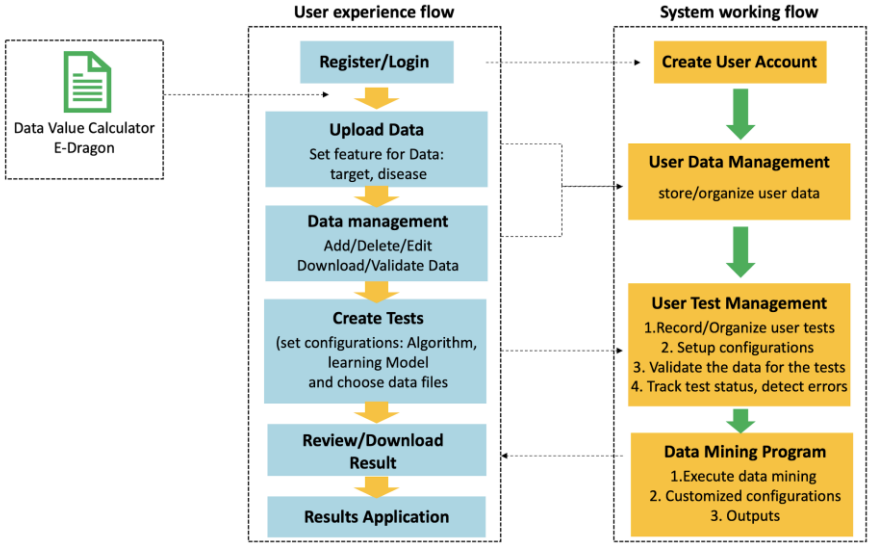

Figure 3: The workflow of the program

5. IMPLEMENTATION AND VALIDATION

5.1 REGISTER AN ACCOUNT/SIGN IN

In order to use this database system, a user needs to register for an account for the first time or sign in after that. To register a new account, a user provides his/her email and sets a password. If the user forgets the password, the system allows the user to receive a new password through his/her email.

5.1.1 UPLOAD DATA FILES

Users can upload their data files and manage them in the database system. This system only accepts CSV (Comma-Separated Value) files. After uploading a file, the user can then designate that file for a data mining test. Each test requires three separate file inputs: one containing a descriptor value matrix, another containing the prediction target values, and the last and optional one containing the names of each property for which a descriptor value was calculated.

There are multiple applications such as E-Dragon[22] that can accept raw data from a user, filter that data, calculate the descriptor values, and output the three required three files. When a user uploads a file to the database, the user can give the file a name and associate it with a particular disease or target compound.

5.1.2 VALIDATION OF THE REGISTRATION - CREATING A TEST

After uploading the three necessary files, a user can create a data mining test. This is done in the "My Test" page by clicking the "Create a new Test" button. The user can give a name and a description to this test. Users are required to designate the evolutionary algorithm and machine learning model that the data mining process will execute. They can additionally specify population sizes, cutoff conditions, number of populations generated, etc. The algorithms and machine learning models will be further detailed in later sections.

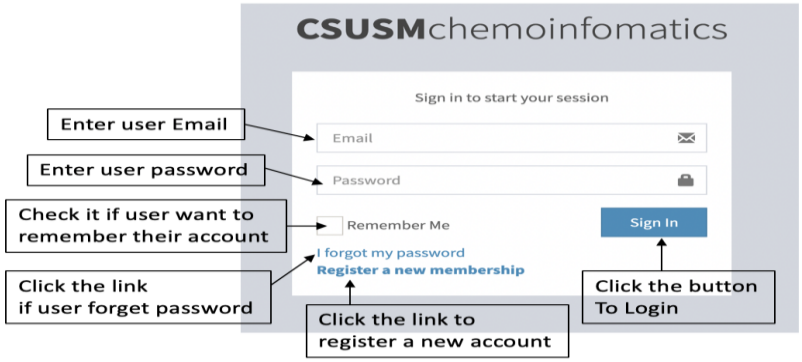

Figure 4: Sign in page

5.2 USER INSTRUCTION AND EXAMPLES

5.2.1 LOGIN AND ACCOUNT REGISTRATION SYSTEM

The login page is the first page of the application. If a user has an account already, then the user can sign in with their email address and password (Figure 4) . If a user enters incorrect login information, the system displays an error message (Figure 5 and Figure 6).

If the user has forgotten the password, then the user can access the "I forgot my password" link and request the system to email a password reset link to the user. If the user doesn't have an account yet, then the user can click the link "Register a new membership" to register a new account.

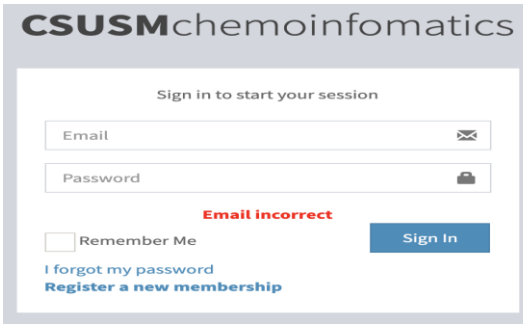

Figure 5: Sign in email error

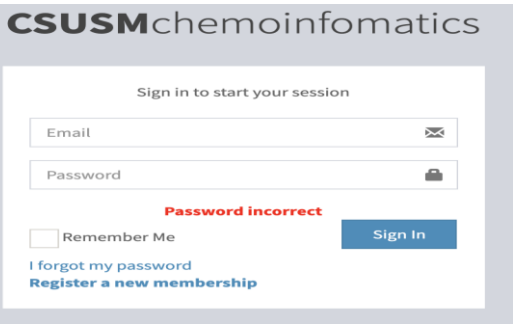

Figure 6:Sign in password error

5.2.2 REGISTER A NEW ACCOUNT

Figure 7 and 8 display the account registration page. The user enters their preferred first name, last name, email address, and password, and then re-enters the password to confirm. Next the user will be shown the terms of service and must agree to them before he or she can confirm his or her

registration. Once the user confirms his or her registration, the system displays a message validating whether the registration succeeded or not (Figure 9).

If the registration succeeds, the user will receive an email from CSUSMChemoinfo with a link (Figure 10). That's the link for the user clicks to activate the account. The link is unique for every user, and this link will automatically login the user account and leads to the dashboard page. If the Email that the user enters already exist, it will show an error message as seen in Figure 11.

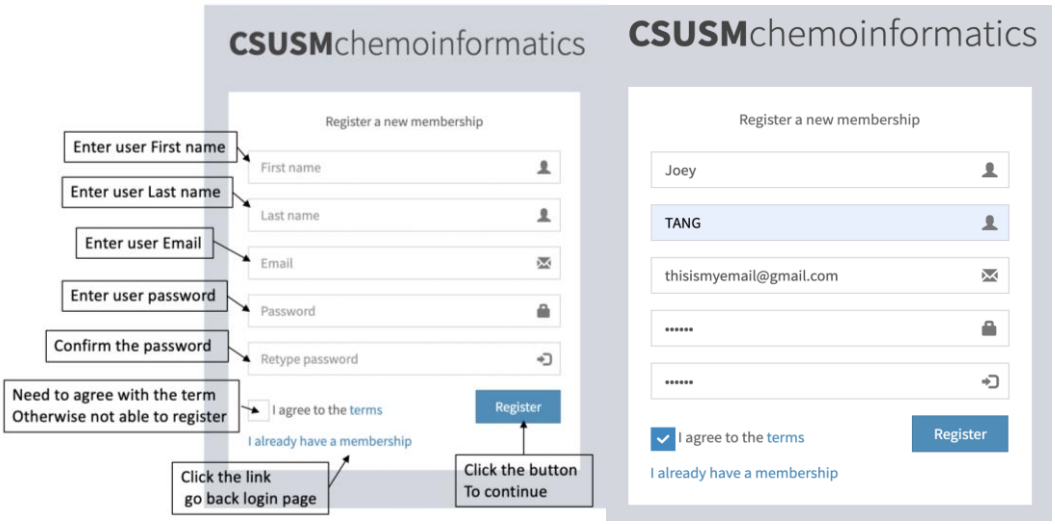

Figure 7: Register new account page Figure 8: Register new account example

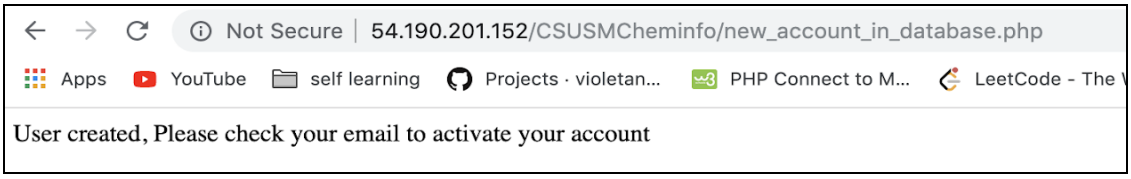

Figure 9:Account created notification

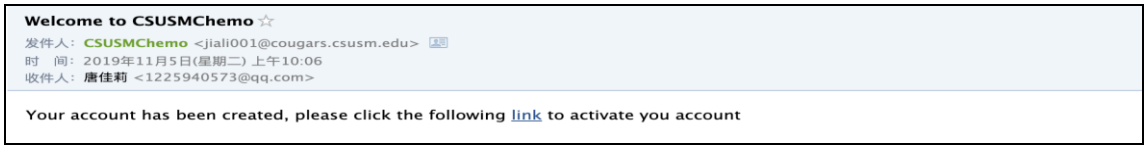

Figure 10: Email send to user to activate their account

5.2.3 FORGET/RESET PASSWORD

If the user clicks "I forgot my password", the user is redirected to the Find Your Password page, shown in Figure 12 and Figure 13. To begin the password recovery process, the user enters his/her email address and clicks on the submit button. Figures 14-17 illustrate the password recovery process.

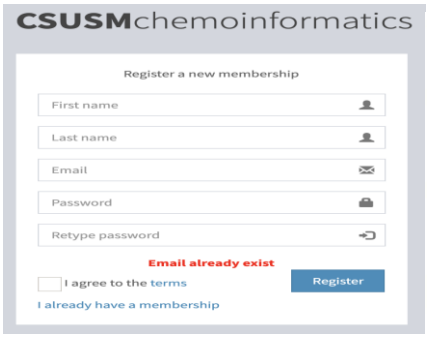

Figure 11: Register error: email already exist

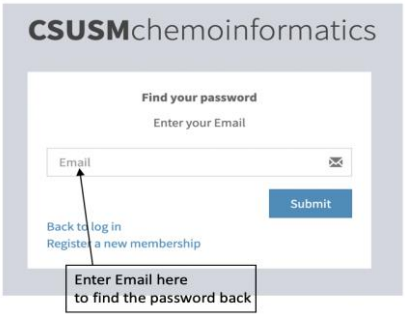

Figure 12: Find password page

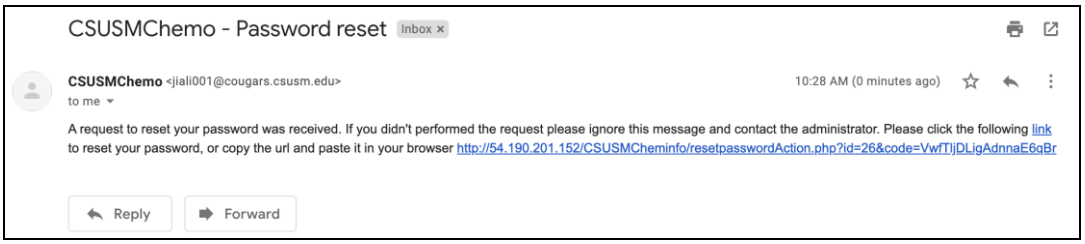

Figure 15: Email sent to user for reset password

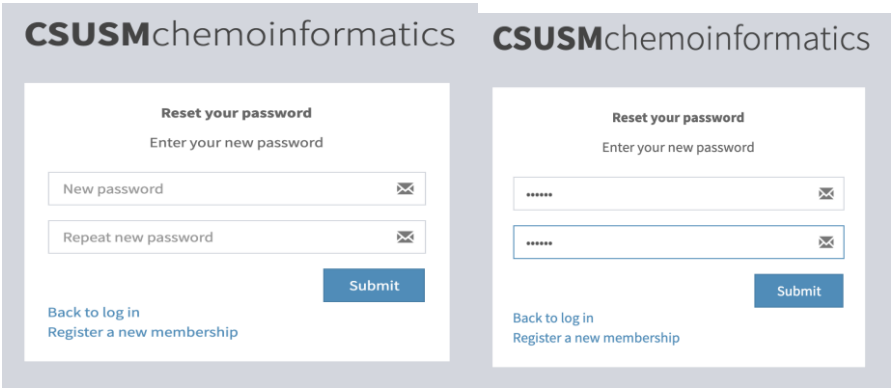

Figure 16: Reset password

Figure 17: Reset password example

5.2.4 DASHBOARD

The dashboard page serves as the main menu for a user. It summarizes the user's test statuses and displays the current progress of tests being run. The main navigation bar is on the left side of the dashboard (Figure 18) and contains the following links: the "My Data" link leads to the uploaded data management page, the "My Tests" link leads to the test management page, the "Profile" link leads to the user's personal information page, and the "Logout" link logs out the current user's account and redirects the user to the main login page. At the top of the navigation bar, there is a button that allows the user to hide the navigation bar. Figure 18 shows the unhidden mode and Figure 19 shows the hidden mode.

Case Status: As shown in Figure 20, the Case status has two parts, a circle graph, and a progress line table. In the circle graph, the colors represent the statuses of all of a user's tests. The progress line shows the number of jobs in certain status.

Figure 18: Dashboard

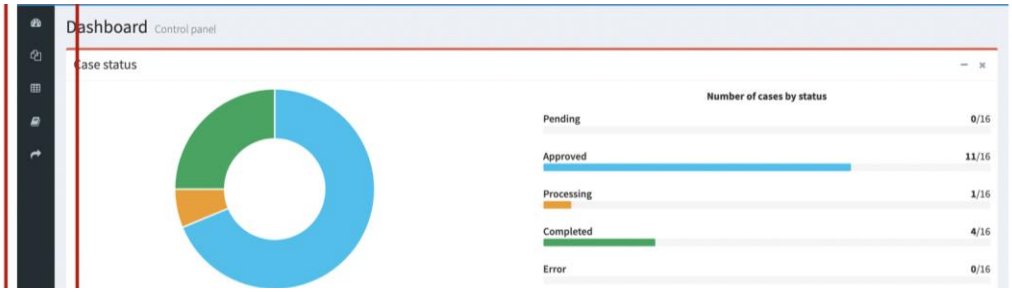

Figure 19: Navigation bar is hidden

Figure 20: Dashboard - Case status

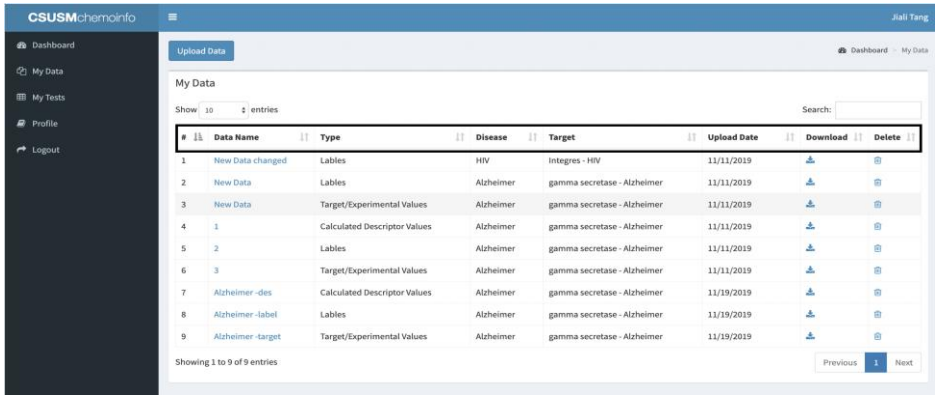

Figure 21: Data List fields

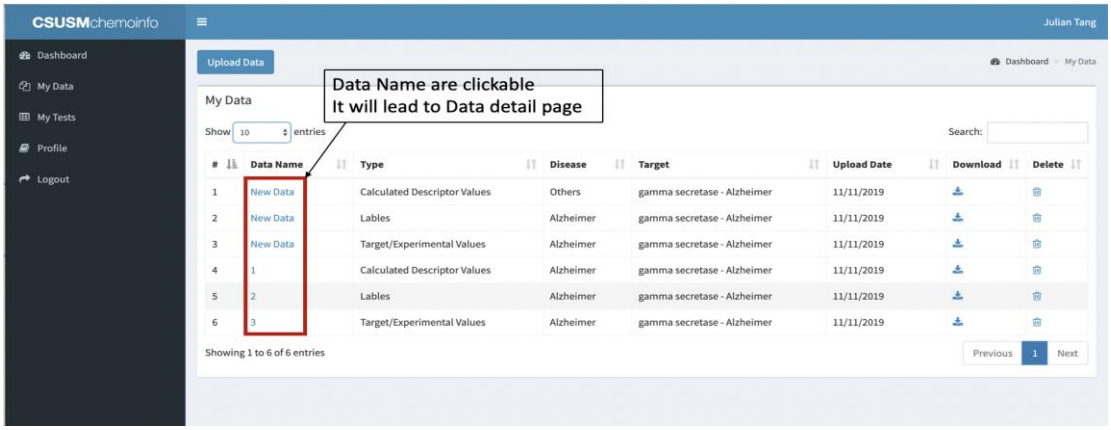

Figure 22: Data List - Data names are clickable

5.2.5 DATA MANAGEMENT

Data List: The Data List page displays all the files uploaded by a user along with each file's associated information, such as data name, data type, disease, target and uploaded date. (Figure 21) A user can download a data file (Figure 23) or delete a data file by clicking the related icons in each row (Figure 24). The user can also click the data name field to view more detailed information about the data (Figure 22).

To upload a new data file, the user can click the "Upload Data" button, which sends the user to the file upload page.

Users can also find data files using the keyword search function. For example, in Figure 25, the keyword is "descriptor". The list shows all the data files with the word "descriptor" in their associated information, in this case data type.

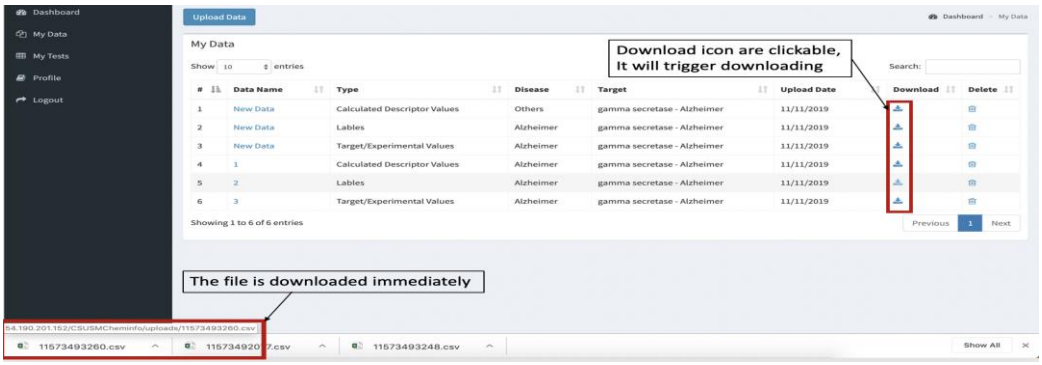

Figure 23: Data List – download icon

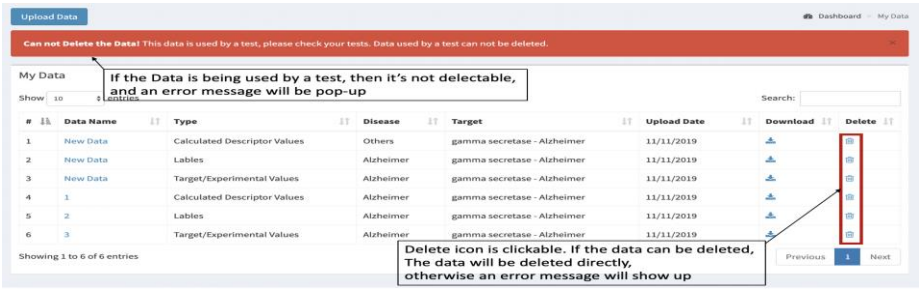

Figure 24: Data List – Delete icon and warning notification

Figure 25: Data List – search example

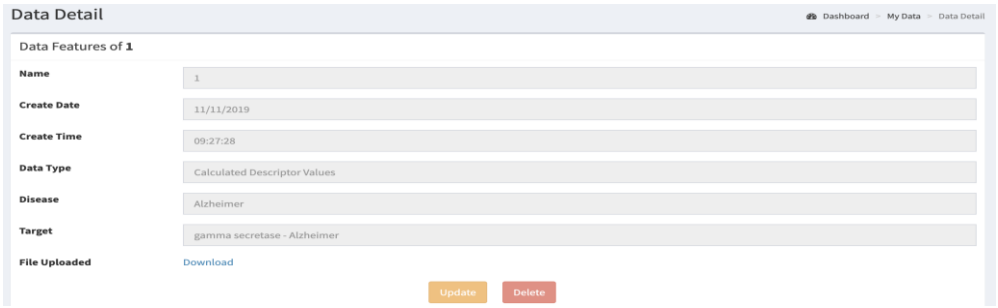

Figure 26: Data Detail – not editable

Show Data Detail / Update Data: Each field in the Data Name column is a link to the associated file's Data Detail page, which allows the user to update the file information or delete the file.

If this data file is being used by a test, then it cannot be updated or deleted. The delete and update buttons are disabled (Figure 26). Otherwise, the data file can be updated or deleted (Figure 27).

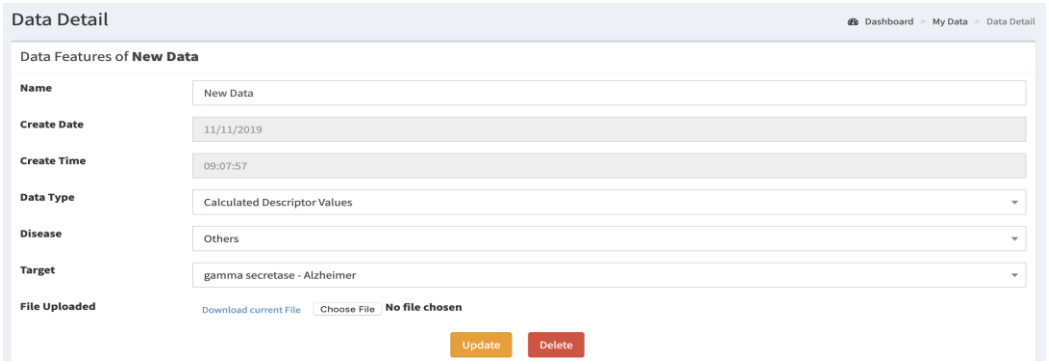

Figure 27: Data Detail – editable

For example, in the "Data Detail Page", the user can replace this file with a new file, give the file a new name or description, and/or associate it with a particular disease or target compounds. Once the changes are confirmed by clicking the "Update" button, the user will be redirected to the "Data List" page (Figure 28), which shows the updated file information as shown in Figure 29.

Upload Data: In the upload data page, users can configure the metadata for the data files that they upload. As seen in Figure 29, the user can name the data file, and choose the related disease and the target. The user can also choose the type of this data file (target, descriptor, label). The user can also associate the data file with particular diseases or target compounds listed in dropdown menus, as shown in Figure 30. Clicking the "Update" button will save this information to the database.

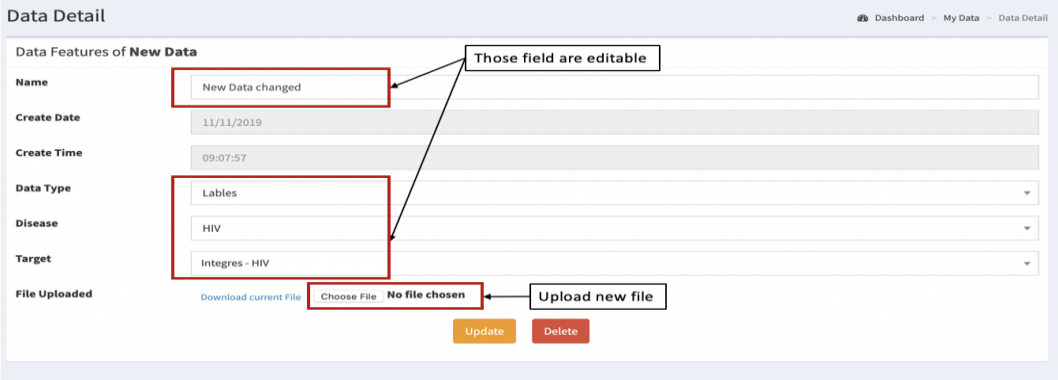

Figure 28: Data Detail – update example

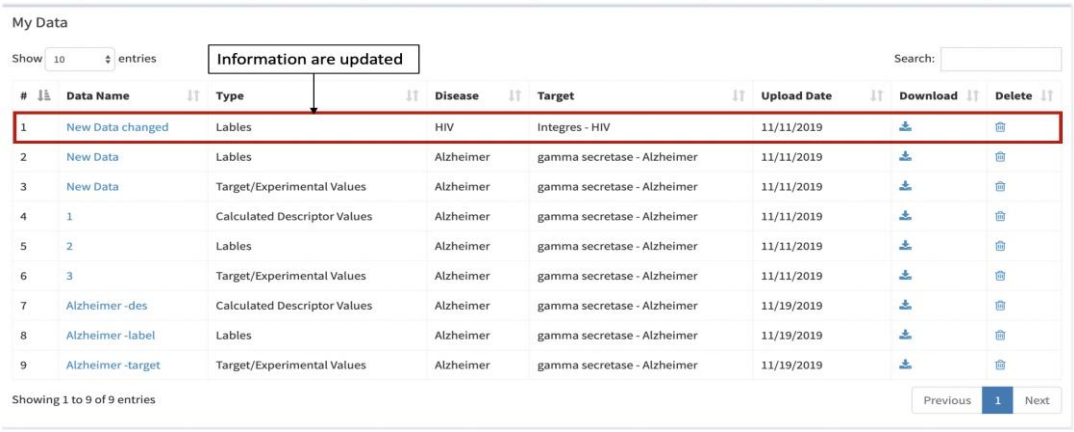

Figure 29: Data List is updated after Data information is changed

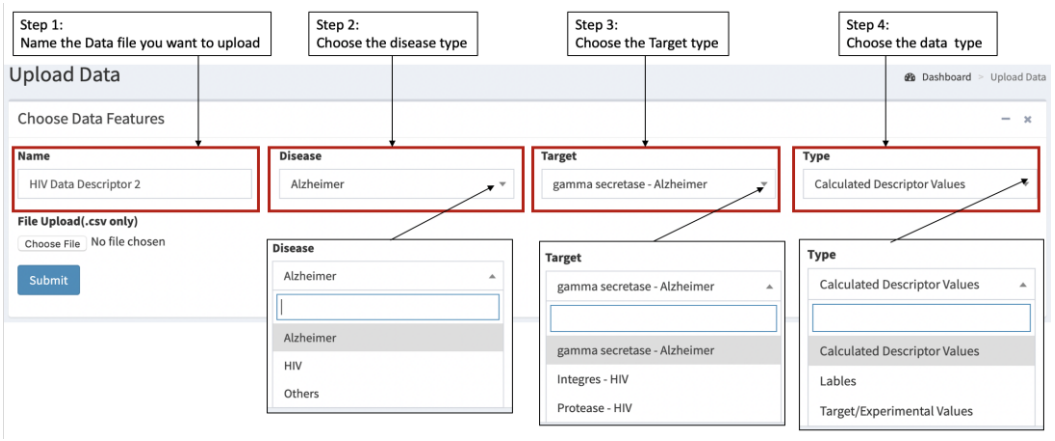

Figure 30: Upload Data

Right now, this system allows data files to be associated with Alzheimer's disease and HIV. We have done some work for HIV-Protease, HIV-Integrase for HIV and gamma-secretases for Alzheimer's. The data mining process requires files for three types of data: calculated descriptor values, target/experimental values, and compound property labels. Users can acquire these data files from E-Dragon [22].

A user can upload a new file by clicking the "Choose File" button, as shown in Figure 31. Only .csv (Comma-Separated Values) files are currently accepted. Once the file has been uploaded, the

user can access the file in the system at any time. Should the upload fail, the system will display an error message that may help the user diagnose the cause of the error (Figure 32).

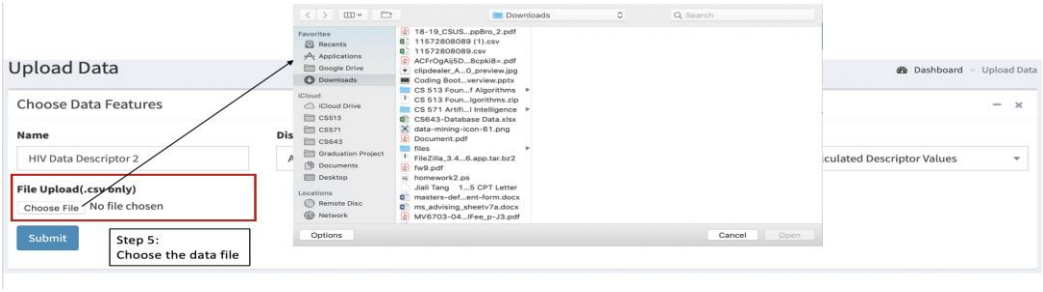

Figure 31: Upload Data – choose file

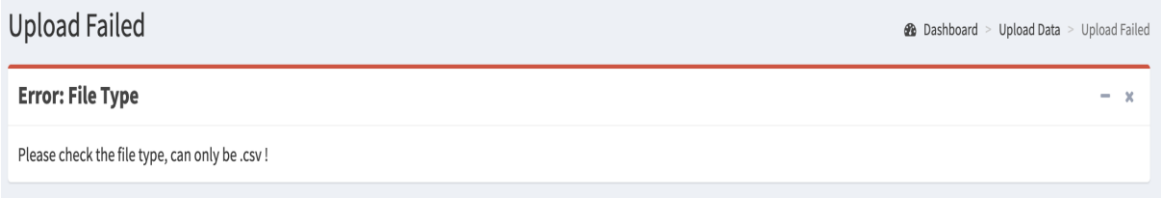

Figure 32: Upload Data – Error message

6. TEST MANAGEMENT

Test List: The Test List page will display all of a user's tests and each test's metadata, such as name, description, algorithm, model, date of creation, and data of completion (Figure 33). Users can also download the data mining results. Embedded in the test's name is a link that can show more detailed information about the test. If the user wants to create a new test, then clicking the "Create new test" button will redirect the user to the "Create Test" page.

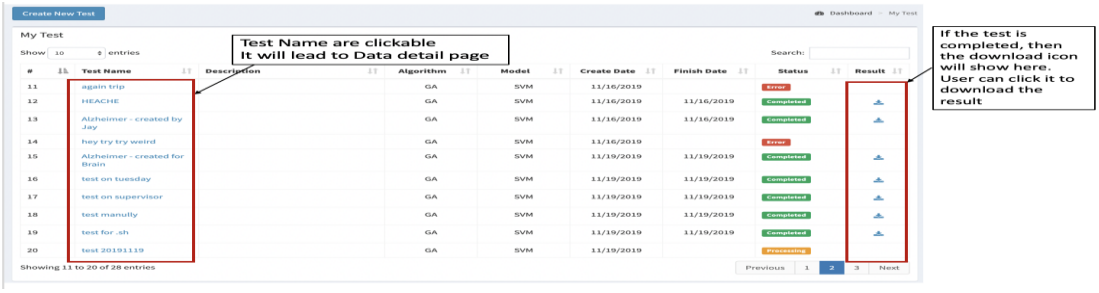

Figure 33: Test List

Users can also retrieve tests by keyword search. For example, in Figure 34, entering the keyword "completed" will return a list of all of a user's completed tests.

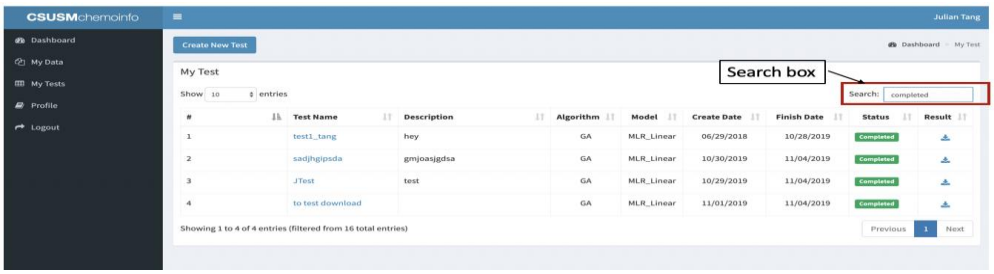

Figure 34: Test List – search function

Create Test: In the Create Test page, users can configure their test and submit the tests to be run. As Figure 35 shows, the user can give a name and description to the test and choose a particular algorithm and machine learning model that will be executed for the test. All available algorithms and machine learning models can be selected from the dropdown menus (Figure 35). There are also dropdown menus that list the user's previously uploaded data. The user must also select three files from these menus: one file with calculated descriptor values, the second with the target/experimental values, and compound property labels.

After setting up those configurations, the user can click the button "Submit" to submit a request to execute the test. When the test completes, its "Status" will change to "Complete".

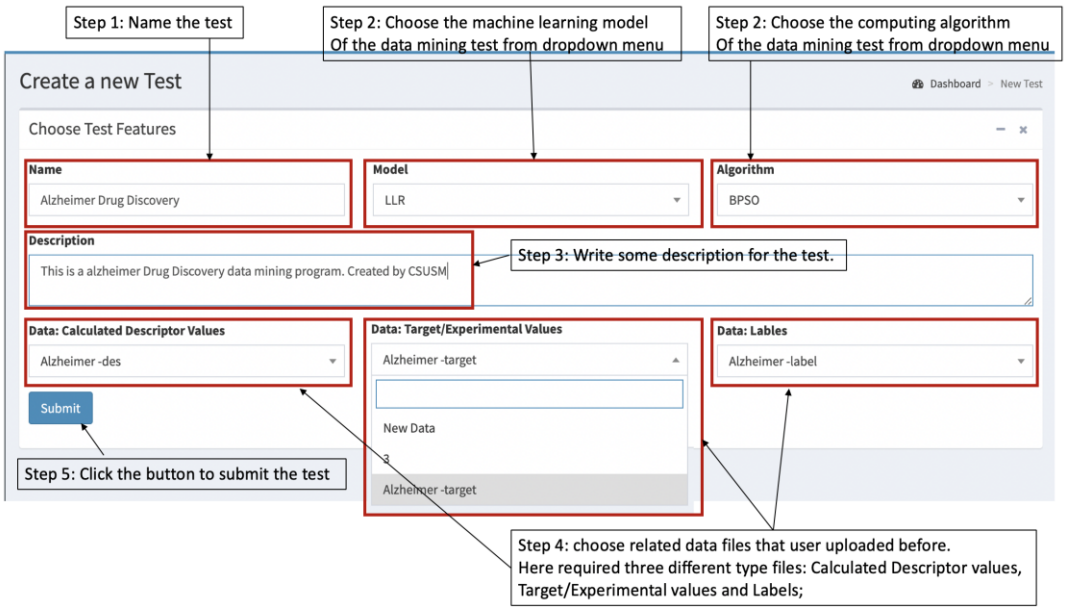

Figure 35: Creating a new Test

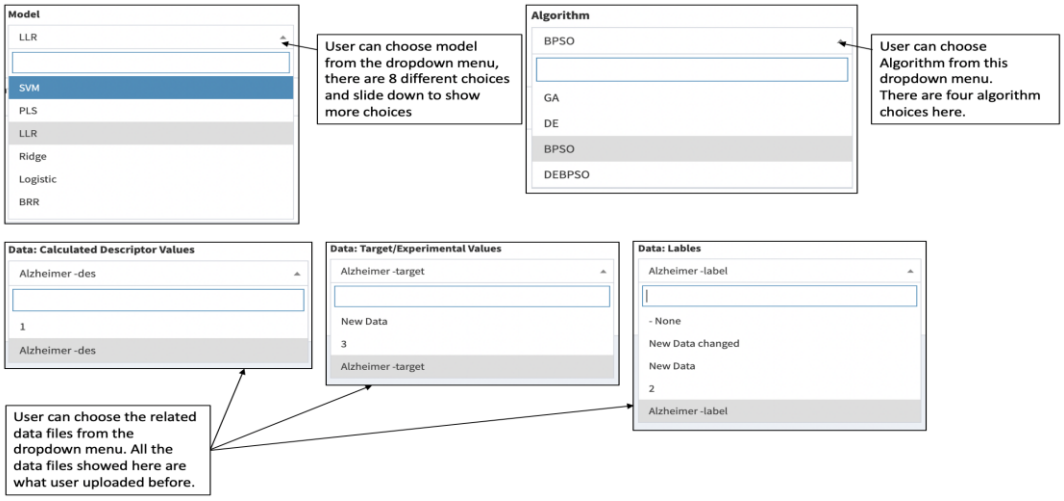

Figure 36: Dropdown menus for test options

Test Details: By clicking the name of a test, a user will be redirected to the Test Details page, which displays more detailed information and additional options for that particular test. If the test is currently being run, then it cannot be updated or deleted. As shown in Figure 37, the delete button is disabled. Once the test is completed, then it can be deleted (Figure 38). If the test ends

prematurely because of an error, which will be displayed in the "Status" column, the test's configuration can be updated or the test can simply be deleted.

Aside from the information about the test, the Test Details page also shows the data files that are being used for the test. The user can access additional information about the data by clicking the "More detail" button, which will lead to the Data Details page. The Data Details page also provides a "Download" button to allow users to directly download a particular file. (Figure 37).

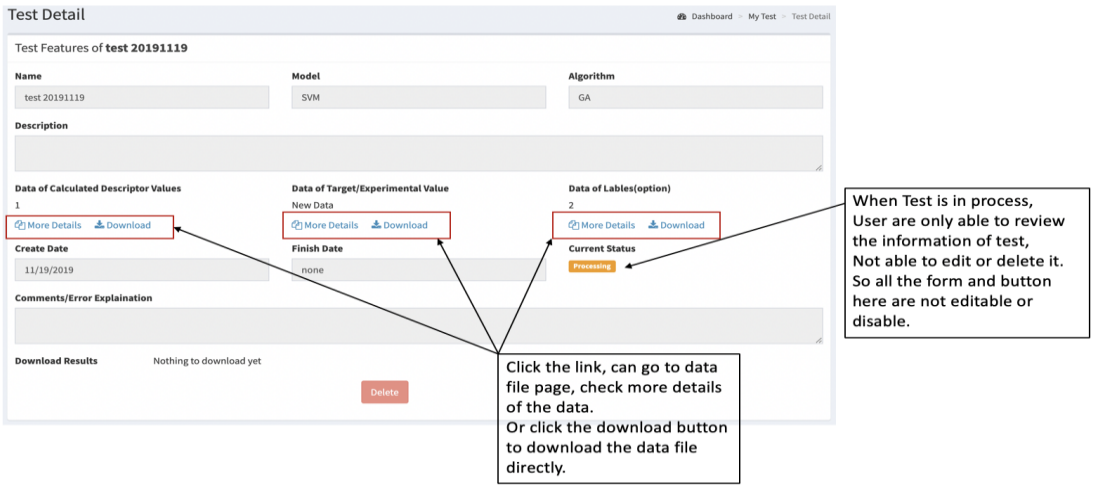

Figure 37: Test Detail – not editable

The Comment/Error column displays any error messages returned by test failures along with any comments about the failed tests. The Download Results column displays download links for the results from successfully completed tests.

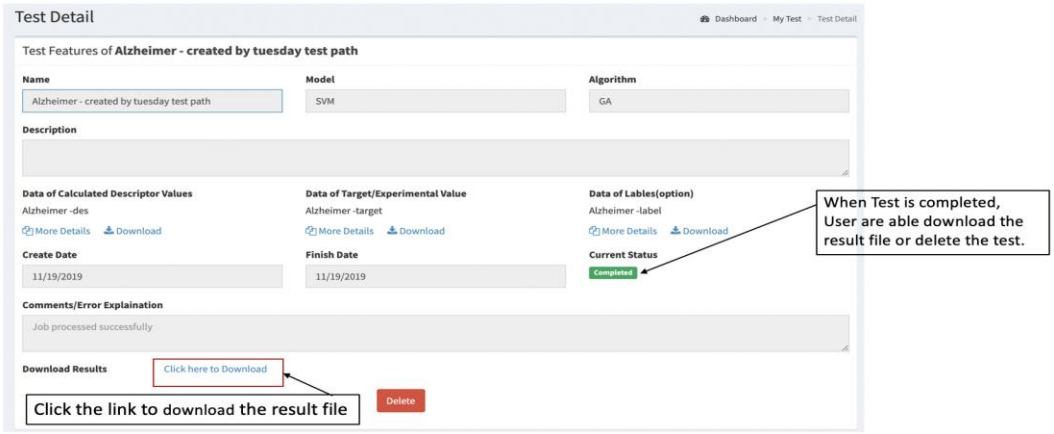

Figure 38: Test Detail – delete

Update Test: A data mining test may fail as a result of incorrect input in the data files or a network connection error. Upon a test failure, users can choose to either delete the test or reset the configuration of the test and execute it again. A user can first go to the Test Details page to check the reasons for the failure. Based on this information the user can then alter the test configuration to remove the failure conditions, such as selecting a different algorithm or machine-learning model or updating to the correct data files. After the user confirms the updated test configuration and by clicking the "Update" button, the data-mining test will be sent to the data mining program for another execution.

7. CONCLUSION AND FUTURE WORK

The paper described a database service and web application to allow any researcher, especially anyone without computer science experience, to utilize a data mining application for drug discovery. This service, hosted on Amazon Web Services, allows users to upload experimental input, run tests, and download test results.

There exist other high-quality data mining services, but this work's specific combination of features, such as machine learning models augmented by evolutionary algorithms and an accessible database to store input and output data, has not been implemented in a public application as far as the authors are aware.

The next phase of implementation can involve adding nonlinear models such as nonlinear SVM, Artificial Neural Network (ANN), and classification models such as Random Forest (RF) to this project's list of machine-learning models.

If the new version of E-Dragon contains an API, another possible expansion could be development of a native API for the project that connects to the hypothetical E-Dragon API. This would allow users to directly submit raw data, such as compound SMILES files, and have E-Dragon automatically filter the raw data and calculate descriptor values without needing a third-party service. This would also improve data security for the application.

The project's cloud infrastructure could also be improved in speed, scalability, and reliability using various tools and systems. The data-mining application is currently hosted on a single instance of AWS Elastic Compute Cloud service; to improve the three characteristics listed before, there are several services that could be added to the stack.

The first step is to create the launch template, which contains configuration data such as network requirements, instance types, and disk images that will be mounted on the instances. With a template, the creation of a scaling group will allow setting dynamic parameters, such as CPU load or memory usage which can trigger the creation and invocation of new instances, as well as the termination of those instances when demand declines, not only optimizing scalability but also reducing costs. Finally, the implementation of an Elastic Load Balancer will allow the incoming traffic and task request to be distributed among the available servers (instances) by allocating them to the highest availability server, which can significantly improve the speed and reliability of our system.

REFERENCES

[1] Ko, Gene, Reddy, Srinivas, Garg, Rajni, Kumar, Sunil, & Hadaegh, Ahmad, (2012) "Computational Modelling Methods for QSAR Studies on HIV-1 Integrase Inhibitors (2005-2010),". Curr Comput Aided Drug Des. Vol. 8, No 4, pp 255-270.

[2] Thakor, Falguni, Hadaegh, Ahmad, & Zhang, Xiaoyu, (2017), " Comparative study of Differential Evolutionary-Binary Particle Swarm Optimization (DE-BPSO) algorithm as a feature selection technique with different linear regression models for analysis of HIV-1 Integrase Inhibition features of Aryl β-Diketo Acids", Proceedings of 9th International Conference on Bioinformatics and Computational Biology, Honolulu, Hawaii, USA, ISBN: 978–1–943436–07–1, pp 179-184.

[3] Kane Ian, & Hadaegh Ahmad, "Non-linear Quantitative Structure-Activity Relationship (QSAR) Models for the Prediction of HIV Drug Performance", (2015), 24th International Conference on Software Engineering and Data Engineering, pp 63-68. Vol 1, ISBN: 9781510812277, San Diego, CA.

[4] Galvan Richard, Kashani, Maninatalsadat, & Hadaegh, Ahmad, "Improving Pharmacological Research of HIV-1 Integrase Inhibition Using Differential Evolution-Binary Particle Swarm Optimization and Non-Linear Adaptive Boosting Random Forest Regression",(2015), IEEE International Workshop on Data Integration and Mining San Francisco, Information Reuse and Integration (IRI), IEEE International Conference, pp 485-490, DOI: 10.1109/IRI.2015.80. INSPEC Accession Number: 15556631. San Francisco, CA.

[5] Kashani, Maninatalsadat, Galvan Richard, & Hadaegh Ahmad, "Improving the Feature Selection for the Development of Linear Model for Discovery of HIV-1 Integrase Inhibitors", (2015) ABDA'15 International Conference on Advances in Big Data Analytics. In Proceeding of the 2015 International Conferences on Advances on Big Data Analyses, pp 150-154. ISBN: 1-60132-411-1, Las Vegas, Nevada.

[6] Ko, Gene, Garg, Rajni, Kumar, Sunil, Kumar, Bailey, Barbara, & Hadaegh Ahmad, "A Hybridized Evolutionary Algorithm for Feature Selection of Chemical Descriptors for Computational QSAR Modeling of HIV-1 Integrase Inhibitors", (2013), Computational Science Curriculum Development Forum and Applied Computational Science and Engineering Student Support for Industry, San Diego State University.

[7] Ko, Gene, Garg, Rajni, Kumar, Sunil, Bailey, Barbara, & Hadaegh Ahmad, "Differential Evolution-Binary Particle Swarm Optimization for the Analysis of Aryl b-Kiketo Acids for HIV-1 Integres Inhibition, (2012), WCCI 2012 IEEE World Congress on Computational Intelligence. Brisbane Australia, pp 1849-1855.

[8] Ko, Gene, Reddy, Srinivas, Kumar, Kumar, Bailey, Barbara, Garg, Rajni, & Hadaegh, Ahmad, "Evolutionary Computational Modelling of β-Diketo Acids for Virtual Screening of HIV-1 Integrase Inhibitors", (2012), IEEE World Congress on Computational Intelligence, Brisbane, Australia.

[9] Ko, Gene, Reddy, Srinivas, Kumar, Kumar, Garg, Rajni, & Hadaegh, Ahmad "Evolutionary Computational Modelling of β-Diketo Acids for Virtual Screening of HIV-1 Integrase Inhibitors", (2012), 243rd National Meeting of the American Chemical Society, San Diego, CA.

[10] Gonzales, Miguel, Turner, Chris, Ko, Gene, & Hadaegh, Ahmad, "Binary Particle Swarm Optimization Model of Dimeric Aryl Diketo Acid Inhibitors for HIV-1 Integrase" (2012), 243rd National Meeting of the American Chemical Society, San Diego, CA.

[11] Ko, Gene, Reddy, Srinivas, Kumar, Sunil, Garg, Rajni, & Hadaegh, Ahmad, "Analysis of HIV-1 Integrase Inhibitors Using Computational QSAR Modelling", (2012), Computational Science Curriculum Development Forum and Applied Computational Science and Engineering Student Support for Industry, San Diego State University.

[12] Garg Rajni, Reddy Srinivas, Zhang Xiaoyu, & Hadaegh Ahmad, "MUT-HIV: Mutation database of HIV proteases", (2007), American Chemical Society (ACS) 234th National Meeting & Exposition, Boston, MA USA CINF 42.

[13] MLR: http://www.stat.yale.edu/Courses/1997-98/101/linmult.htm

[14] PLSR: https://www.mathworks.com/help/stats/plsregress.html

[15] https://techdifferences.com/difference-between-descriptive-and-predictive-data-mining.html

[16] Zhong et al. Artificial intelligence in drug design. Sci China Life Sci. 2018 Jul 18. doi: 10.1007/s11427-018-9342-2. [Epub ahead of print]

[17] Varsou Dimitra-Danai, Nikolakopoulos, Spyridon, Tsoumanis Andreas, Melagraki Georgia, & Afantitis, Antreas, "New Cheminformatics Platform for Drug Discovery and Computational Toxicology", (2018), Methods Mol Biol. 2018; 1800:287-311. doi: 10.1007/978-1-4939-7899-1_14

[18] Ekins, Sean, Clark, Alex, Dole, Krishna, Gregory, Kellan, Mcnutt, Andrew, Spektor, Anna, Weatherall, Charlie, & Litterman, Nadia, "Data Mining and Computational Modeling of High-Throughput Screening Datasets", (2018), Methods Mol Biol, 1755:197-221. doi: 10.1007/978-1-4939-7724-6_14.

[19] Sam Elizabeth, & Athri Prashanth, "Web-based drug repurposing tools: a survey. Brief Bioinform", (2017), Oct 6. doi: 10.1093/bib/bbx125. [Epub ahead of print].

[20] Kaur, Charanpreet, & Bhardwaj, Shweta, "DRUG Discovery Using Data Mining International Journal of Information and Computation Technology", (2014), ISSN 0974-2239 Volume 4, Number 4, pp 335-342 © International Research Publications House http://www. irphouse.com /ijict.htm

[21] Minaei-Bidgoli, Behrouz, & Punch, William, "Using Genetic Algorithms for Data Mining Optimization in an Educational Web-Based System, (2003), Genetic Algorithms Research and Applications Group (GARAGe) Department of Computer Science & Engineering Michigan State University 2340 Engineering Building East Lansing, MI 48824.

[22] https://chm.kode-solutions.net/products_dragon.php

[23] AWS LightSail: https://aws.amazon.com/lightsail/?nc2=h_ql_prod_fs_ls

[24] AWS EC2 Server: https://aws.amazon.com/ec2/