A Hadoop-based Molecular Docking System

Yueli Dong, Quan Guo and Bin Sun
Department of Software Engineering, Dalian Neusoft Institute of Information, Dalian, China
Email: dongyueli@neusoft.edu.cn

Abstract. Molecular docking always faces the challenge of managing tens of TB datasets. It is necessary to improve the efficiency of the storage and docking. We proposed the molecular docking platform based on Hadoop for virtual screening, it provides the preprocessing of ligand datasets and the analysis function of the docking results. A molecular cloud database that supports mass data management is constructed. Through this platform, the docking time is reduced, the data storage is efficient, and the management of the ligand datasets is convenient.

1. Introduction
Traditionally, lab chemical experiments are needed in the field of drug design, which cost much money and manpower. Computer-aided drug design can be used to focus on the docking objects set, and the period time of making drug can be shortened. With the completion of human genome project and the rapid development of structure biology and protein purification technologies, the suitable targets of receptor molecules have increased dramatically, at the same time the commercial small molecule database has been updated continuously. It is necessary to use computing technology to optimize the process. Computer aided drug molecular design[1] focuses on the molecular docking object sets. However, the search space of molecule docking is tremendous. A rough estimate of docking covered in the search space includes at least $10^{30}$ solutions which need a large amount of computation time.

Virtual screening procedures search collections of small molecules, seeking those members that contain a set of features that matches a defined search goal. With growth of structural and non-structural data of the ligands, it is an important issue to store and manage a large amount of data. Cloud computing technology completes storing and computing of massive data by distributing data to each computing node of cluster through the network. Tsai[2] constructed a cloud-computing system for traditional Chinese medicine(TCM) intelligent screening system (iScreen). Capuccini, M.[3] developed a method to run existing docking-based screening software on distributed cloud resources, Hongjian Li[4] have developed a publicly-accessible web platform called istar.

In this paper, we proposed a method to construct a molecular database for ligand preprocessing. We implemented a Hadoop-based virtual screening platform using AutoDock Vina[5]. Hadoop[6] is an open-source framework, originally introduced by Google for parallel processing of many small data items, is widely used in massive data processing. Hive is an open source data warehouse tool which is based on the Hadoop. Hive can map the file into a data table, and provide HQL statements. AutoDock Vina is a well-known tool for protein-ligand docking built in the same research lab as the popular tool AutoDock[7]. Our system can reduce a timeline and cost of drug discovery.
2. Data Model
The non-drug-like molecules (according to user-defined filters) can be selected before docking. Some ligands datasets which cannot be the targets should be eliminated from the datasets for docking process, so preprocessing of data is needed to improve the efficiency of docking. During the process of ligand preprocessing, users search for the datasets according to the properties of the compound, and the selected ligands are submitted for docking.

Two tables are necessary for handling properties and the mol2 file of ligand. Table 1 shows the properties of ligand. Users can retrieve data fields for virtual screening and the retrieved data automatically compose a file in mol2 format. Table2 stores the data of the ligands. The results of docking are stored in table 3.

| Attribute        | Description                        | Type    | comment         |
|------------------|------------------------------------|---------|-----------------|
| Ligand_ID        | Ligand ID                          | string  | Primary key     |
| Ligand_name      | Ligand name                        | string  | Unique          |
| XlogP            | Log of the fat water distribution coefficient | double |                |
| Rotatable_bond   | number of rotatable bonds          | int     |                |
| H_bond_donors    | number of H bond donors            | int     |                |
| H_bond_acceptors | number of H bond acceptors         | int     |                |
| Molecular_weight | sum of the weights of the molecule | double  |                |
| Apolar_desolvation | Apolar desolvation             | double  |                |
| atom_count       | Atoms of which it is made          | int     |                |
| bond_count       | bonds count                        | int     |                |

Table 2 The file table of ligand

| Attribute        | Description                        | Type    | comment |
|------------------|------------------------------------|---------|---------|
| Ligand_ID        | Ligand ID                          | string  | Primary key |
| Ligand_name      | Ligand name                        | string  | Unique  |
| Content          | Whole contents of Mol2 file        | string  | -       |
| Ligand_name      | Ligand name                        | string  | Unique  |

Table 3 The result table of docking

| Attribute        | Description                        | Type    | comment |
|------------------|------------------------------------|---------|---------|
| Ligand_ID        | Ligand ID                          | string  | Primary key |
| Ligand_name      | Ligand name                        | string  | Unique  |
| score            | the minimum value of the scores    | string  | -       |

Our platform provides the service for query and the service for preprocessing which inquires information of these ligand data tables.

3. Architecture of Hadoop-Based Molecular Docking Platform
The architecture of molecular docking platform based on Hadoop is shown in Figure 1.

The molecular docking platform includes four layers, they are User Layer, Hive Layer, MapReduce layer and HDFS(Hadoop Distributed File System) layer. User Layer provides services for users, include preprocessing of ligands and operating of docking results. After preprocessing, the drug-like datasets of ligand will be selected as the input file of AutoDock Vina. Hive Layer provides data support. MapReduce is a high performance computing model of Hadoop. The application is executed in many small fragments of works by different nodes in the clusters. Map/Reduce framework divides the work in two phases: the map phase and the reduce phase separated by data transfer between nodes in the cluster. The Reduce stage produces another sets of key-value pair, as final output based on user defined reduce function. HDFS is a Hadoop distributed file system running on commodity hardware.
In HDFS, data is organized into files and directories. Files are divided into uniform sized blocks and distributed across cluster nodes. After docking, the minimum value of the score in the output file will be stored in the result table. Users can retrieve and analysis the information after docking by using the services of this platform.

4. AN Algorithm of Docking Process

The essential enhancement is the data management from the main jobs in virtual screening, such as docking and ranking. Users not only can control and inspect the computing process but also attain consistent and logically organize data while computing is in progress or finished.

4.1. Preprocessing

All structures and properties of the molecules required for docking and ranking were calculated and stored in the database (Table 1 and Table 2). For each molecule in the ZINC library 25 CHARMM26 atom types were assigned. Then preprocessing system was applied to calculate the atomic and chemical properties of each molecule. Even though not all of these properties were used in docking and ranking, they were prepared for different kinds of filters. Besides these properties, the mol2 file of each molecule was also stored in the database.

4.2. Docking

In order to improve the efficiency of docking, molecules from the library ligand table were selected according the drug-like rules. The poses of each molecule in the PDB format, with their interaction energies with the receptor and efficiencies (electrostatics and VDW), were stored in the table Pose of the database. During the docking, the computing clients acquired the 3D structure of the molecules directly from the database and stored poses and energies in the database after each docking process finished.

4.3. Ranking

We used a developed approach based on calculations of quantum mechanics to efficiently rank the poses. Algorithm used of docking process is as follows.

1) Connect to Mol2File database;
2) Lock the data in table Ligand_file according to the properties of the ligands and set ‘unfinished’ to ligands;
3) While there are still unfinished ligands exist do
4) Select a ligand within unfinished molecules in table Ligand_file;
5) Generate the pose records randomly and store them into table pose in Pose database;

Figure 1. The architecture of molecular docking platform based on Hadoop
6) Compute each pose in clients parallel;
7) Store the interaction energy w in Ranking table;
8) Update the tag of ligand to ‘finished’;
9) Endwhile.

5. Experiments and performance evaluation
We implemented a Hadoop-based molecular docking platform in order to validate the performance of the platform based on Hadoop. This section describes the hardware components in the platform and the testing scenarios. Figure 2 shows the deployment of the experimental system.

![Figure 2. The deployment of the experimental system based on Hadoop](image)

This experimental system assigns one node as controller for the management and five nodes as slaves of the Hadoop cluster. The CPU of controller and each computer is Inter(R) Core™ 2Quad, 2.66Hz processor, and the memory is 8GB, the hard drive of 160GB. We construct a database that store existing chemical databases using MySQL.

We performed docking jobs with 2000 ligands on a target receptor. Figure 3 shows the comparison of execution time as the number of docking jobs increases. We compared three different approaches to execute docking jobs. The first approach is to execute docking jobs on only single node which has the best computing performance. The second approach is to execute docking jobs on Grid-based molecular docking system[8] with 5 computation nodes. The third approach is to execute docking jobs using our Hadoop-based molecular docking system. In Figure 3, we can see that the performance of our molecular docking system is better than other approaches.

![Figure 3. The comparison of execution time](image)

When 2000 docking jobs are executed, the execution time of first approach is 455.3s, the execution time of second approach is 1547.9s and third approach is 3368.9s.
When a crash failure happens, in the first case, the docking jobs will be restarted, that means more time will be cost. In the second case, resource manager will migrate the remaining jobs to restart docking in the other computing node. In the third case, resource manager performs rescheduling and job migration when a crash failure happens. Figure 4 shows the comparison of the extra time for three cases to deal with the failure when crash failure happens during the docking process of 2000 ligands.

According to the performance testing on the experiment platform, it shows that the Hadoop-based molecular docking system has a good performance, scalability and fault tolerance. Meanwhile, this platform provides preprocessing function and result processing function for users to access.

6. Conclusion
In this paper, we design and develop a Hadoop-based molecular docking system, which reduces an unmanageable number of compounds to a limited number of compounds for the target of interest. We constructed a ligand database for virtual screening and developed the preprocessing service and ranking service for uses. Through performance testing on the experiment platform, our system can reduce the timeline and cost of drug discovery. We plan to improve the algorithm to improve efficiency, and make various experiments with large datasets to measure the platform in the future.

7. Acknowledgment
These works are mainly supported by National Science Foundation under Grant No.61170168 and No.61170169.

8. References
[1] Chen Kai-xian, Jiang Hua-liang, Ji Ru-yun, Computer aided drug design -- The principle, method and application, Shanghai: Shanghai scientific & Technical Publishers, 2000.
[2] Tsai, Tsung Ying, K. W. Chang, and Y. C. Chen, “iScreen: world’s first cloud-computing web server for virtual screening and de novo drug design based on TCM database@Taiwan,” Journal of computer-aided molecular design, vol.25, 2011, pp.525-531.
[3] Capuccini, M, et al, “Large-scale virtual screening on public cloud resources with Apache Spark,” Journal of Cheminformatics, in press.
[4] Li H, Kwong-Sak L, Ballester P J, et al, “istar: A Web Platform for Large-Scale Protein-Ligand Docking,” Plos One, 2014, vol.9, pp.1-12.
[5] Trott O, Olson A J, “AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading,” Journal of Computational Chemistry, vol.31, 2010, pp.455-461.
[6] Apache Hadoop framework, Accessed: June 2017, http://hadoop.apache.org/.
[7] Ewing T J, Makino S, Skillman A G, et al, “DOCK 4.0: search strategies for automated molecular docking of flexible molecule databases,” J Comput Aided Mol Des, vol. 15, 2001, pp.411-428.
[8] Dong Yueli, Guo Quan, Sun Bin, et al, “Dynamic task migration optimizing for molecule docking,” Journal of Jilin University (Engineering and Technology Edition), vol.45, 2015, pp.1253-1259.