Utilities for Efficient Usage of Large Biological Databases

Maulika S Patel & Himanshu S Mazumdar, Senior Member, IEEE

Abstract— **We have been witnessing a meticulous expansion in the amount of biological databases as an outcome of the human genome-sequencing project. These biological databases are created and updated by the inventions of new molecules by the biologists. The nature of most of these databases is either non-structured or semi-structured. The data are stored in a flat file, which makes it difficult to retrieve a particular record in reasonable time. Computational biology tasks such as multiple sequence alignment, sequence similarity, motif finding, and structure prediction have yanked many researchers. We feel that computational biologists are many a time not interested in all the fields present in the database. Rather, they are concerned about particular fields depending upon the issue being addressed. We have developed utilities to extract and index UniRef100 database for fast sequential and indexed random access, to normalize occurrences of pairs, trios and quads substrings of amino acids in the database, a programmatically mutated database to test the sequence similarity algorithms. This work shall aid the upcoming researchers in the field of computational biology to customize existing database for the algorithmic needs to accelerate the operations.**

Index Terms—**UniRef100 protein database, customized database, substring frequency, sequence similarity, structure prediction**

I. Introduction

Computational biology [9] tasks such as multiple sequence alignment [13], sequence similarity [9], motif finding, and structure prediction [10] have yanked many researchers. Biology in the 21st century is being transformed from a purely lab-based science to an information science as well [8]. The biological databases are very rich in nature in terms of content and size. The primary public protein databases are Protein Data Bank (PDB) [6, 7] and SWISS-PROT [2]. The genomic databases are GenBank [8] in USA, DNA Data Bank in Japan (DDBJ) [3] and European Molecular Biology Laboratory DNA database (EMBL) [4] in Europe.

Manuscript received May 9, 2009.

Maulika S. Patel is working as Assistant Professor and Head, Department of Computer Engineering, G H Patel College of Engineering & Technology, Vallabh Vidyanagar, India and a PhD Student at Research and Development Center, DDU, Nadiad, India. E-mail: maulika.sandip@gmail.com.
H. S. Mazumdar is working as Professor and Head, Research and Development Center, Dharmsinh Desai University, Nadiad, India. E-mail: hsmazumdar@hotmail.com.

The PDB contains details about experimentally determined structures of proteins, nucleic acids, and complex assemblies.

The UniProt Knowledgebase [1] is another popular database amongst the researchers. It consists of protein sequences, description of the function of a protein, its domains structure, post-translational modifications, variants, etc., with a minimal level of redundancy and high level of integration with other databases.

GenBank is the database of publicly collected genetic sequences. There are approximately 106,533,156,756 bases in 108,431,692 sequence records in the traditional GenBank divisions and 148,165,117,763 bases in 48,443,067 sequence records in the WGS division as of August 2009[8**]**.

The EMBL Nucleotide Sequence Database (EMBL-Bank) comprises of Europe's primary nucleotide sequence resource. Individual researchers' submissions, genome sequencing projects and patent applications contribute to the formation of this database [4, 5].

DNA Data Bank of Japan (DDBJ) is the nucleotide sequence data bank in Asia. It collects nucleotide sequences from researchers and issues the internationally recognized accession number to data submitters [3].

This availability of free and colossal data is a boon and a curse both for computational biologists. The growth rate of protein and genomic databases can be visualized in [7, 8]. The downloadable data from the primary databases are sometimes over loaded with information; this very feature makes the use of databases difficult. On the other hand, availability of enormous data ensures that machine learning tools that heavily depend on the training data may be used effectively. As more and more whole genomes are sequenced, the need for a central, publicly available and easily accessible archive for deposition, searching and analysis of sequence data continues to grow [5]. Researchers involved in the development of tools for sequence similarity and structure prediction might face difficulties in handling these databases. We feel that these semi-structured databases of genomic and proteomic data, needs to be used along with customization for each problem being addressed. Most developers using the protein database would like to view different cross-sections of the database based on the clustering algorithms. These clusters could be k-neighborhood member subset, a cluster of sequences containing desired substring, or containing a combination of substrings. This work is expected to be of use to particularly upcoming researchers in the field of computational biology to use the existing public domain databases in an efficient manner. This will not only ensure that the data are in required form, but also save time. Computational biologist would like

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

to test their packages so that the run time of the program and access time of the database is minimized.

Many reported findings use limited set of data to save time and energy, which is misleading sometimes, as this set may not be representative of the complete set. A local copy of customized set of complete data is desired for perfection. We demonstrate a set of utilities to customize the entire data downloaded from the primary databases so as to comply with the needs of the problem being addressed. Most analysis requires normalized values of the data elements repeatedly for training purpose. Computational overheads on post-processing of the data after data-fetch could be considerably reduced by customizing post-processed random accessed dataset. Neural network algorithms need normalized occurrences of pairs, trios and quads substrings of amino acid in input protein sequences. This needs global average of such substrings as a reference of normalizing the local occurrences. These substrings are particularly interesting to locate similar sequences.

The rest of the paper is organized as follows. A brief discussion on the methods of handling large databases is presented in Section II. Section III describes and demonstrates the use of our utilities. The conclusion is given in Section IV.

II. Methods of Data Handling

Most biological archives are text files, comprising of 8-9 million records and 2-4 GB size. It is difficult to use office tools or compiler editor for viewing, editing or cut-past operation. One of the possible ways to operate on these data is to port to existing database system like MS SQL. This will enable to view and customize records interactively or programmatically using query. However online batch operations like training, sorting and classifying is very slow with 9 million data records and database overheads.

Here, we have used the following approach. For accessing a random sequence, an indexed binary file is used to extract correct length sequence from desired location of file with minimum overhead. The record pointer and record length is fetched from binary indexed file by positioning the read pointer at (record no * size of record). Index record consists of data record position and data record length as long integer.

III. Utilities

Following utilities are developed using the procedure mentioned in section 2.

a. To extract training set for applications such as similarity search and secondary structure prediction
b. Create a database of protein sequence and secondary structure pair
c. Computing the frequency of substrings of different lengths
d. Create a programmatically mutated database.

A. *Extracting training set:*

We have been using the UniRef100 database [14] for our research. Most frequently, data are extracted from each record to suit the algorithm being tested. For example, a neural network may require training files that consist of protein sequence and secondary structure pair for secondary structure prediction. Similarly, a test file is also needed to test the neural network. These functions are common in most operations and consume most time. A tool, as shown in figure 1, is developed to extract such parameters to form desired training sets and simple retrieving functions. This is particularly required for our ongoing work that uses neural network for secondary structure prediction.

B. Database of relative frequency of sub-strings in the UniRef100:

A neural network may need normalized parameters as input and output, or normalized occurrence matrix of pairs and trios of amino acids. We have developed a utility to calculate the sorted frequencies of potential 20 mono, 400 pairs, 8000 trios, and 160000 quads of amino acids. This information has been further used in our similarity search algorithms. We describe the method used for the above purpose.

-We take an initialized integer array "**freq**" of length equal to the maximum length (of all possible sequences) of desired substring length. Each string from desired database whose frequency is to be estimated is examined as follows.

-A substring of desired length is extracted from left to right in sequence.

-Each substring is converted to index integer by subtracting 'A' from each letter representing amino acid and using the code in figure 2.

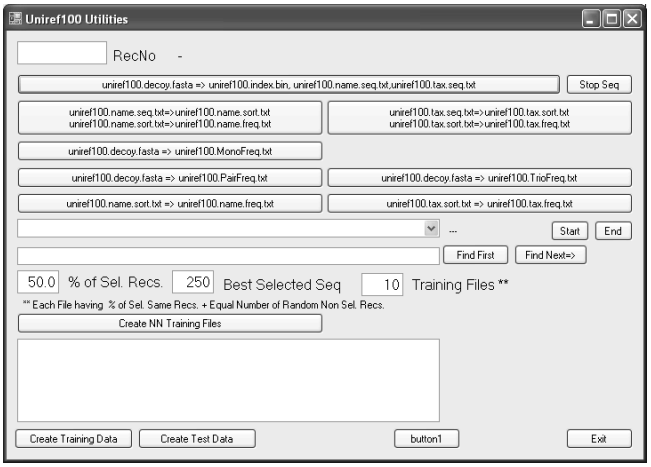

Fig. 1: Snapshot showing Binary file generation, Generating index file to retrieve records, and calculating mono, pair, trio and quad amino acids frequency. The remaining buttons help in providing input to a neural network.

On occurrence of a letter, corresponding array element is incremented using the index value. Value of freq[n] depends on the size of the sample. Hence we have normalized the values using the following equation.

$$Freq[n] = Freq[n]/\Sigma freq[n] * 100 \ldots\ldots\ldots \text{Equation 1.}$$

This data are saved in a binary file for future use. Similarly, frequency is calculated for each sequence individually, normalized with respect to the average frequency of the database.

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

```
for (int k = 0; k < seqn.Length; k++)
    {
            int p = (seqn[k] - 'A');
            FreqMono[p]++;
            sum++;
    }
for (int k = 0; k < seqn.Length - 1; k++)
    {
        int p = (seqn[k] - 'A') * 26 +
                (seqn[k + 1] - 'A');
        FreqPair[p]++;
        sum++;
    }
for (int k = 0; k < seqn.Length - 2; k++)
  {
      int p = (seqn[k] - 'A') * 26 * 26 +
              (seqn[k + 1]-'A') * 26 +
              (seqn[k + 2] - 'A');
      FreqTrio[p]++;
      sum++;
    }
```

Fig. 2: Pseudo code for calculating frequency of substrings of size 1 (mono), 2 (pair) and 3 (trio)

C. Database of protein sequence and secondary structure pair

One of the very popular bioinformatics tasks is the protein secondary structure prediction [3]. Several methods and techniques have been published that addresses this issue [10]. Researchers have been using various databases, and many have their own databases for the purpose. Our next operation extracts those records from UniProt that have secondary structure information. Not surprisingly, these records do have lot of other information of biological relevance and needs filtering to fit the input and output requirements. Computer scientists would be highly interested in the sequence and the structure of the sequence. Many neural network [11] algorithms [12] have been reported for secondary structure prediction. It is known that neural networks learn from data. So, this operation provides the sequence and structure pair database, which can be used by neural networks.

D. Creating a programmatically mutated database

Last, but not the least, we have created a programmatically mutated bio-random database. The input to this operation is the sequences of UniProt. From these sequences, we apply desired amount of random mutation, i.e. insert, delete or replace a character, at every step either successively or from the original sequence. Also, we formed groups of sequences. The user can set the number of groups and the number of sequences in each group. This is particularly useful for extensive testing of sequence similarity search algorithms. Figure 3 shows a snap shot for the same.

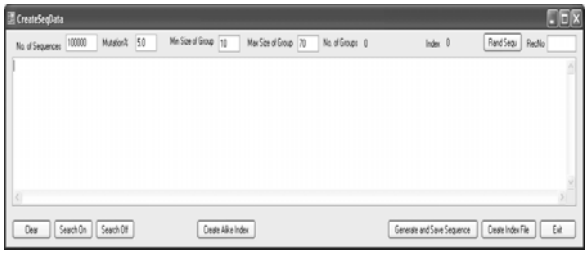

Fig.3: A synthetic database creation snapshot

IV. Conclusions

We have implemented and demonstrated a set of utilities that indexes the primary database (huge in size), extracts training set for applications such as similarity search and secondary structure prediction, computes the frequency of substrings of different lengths, and creates a programmatically mutated database. Due to space constraints, we have here provided only two snapshots of the utilities being used for two different purposes. However, we have discussed all the features of the utilities in detail. We strongly believe that use of such utilities will aid the researchers in using the public domain databases and testing their algorithms efficiently.

References

[1] Baris E. Suzek, Hongzhan Huang, Peter McGarvey, Raja Mazumder, and Cathy H. Wu, UniRef: comprehensive and non-redundant UniProt reference clusters, Bioinformatics Advance Access published on May 15, 2007, DOI 10.1093/bioinformatics/btm098. Bioinformatics 23: 1282-1288.

[2] Amos Bairoch, and Brigitte Boeckmann, The SWISS-PROT protein sequence data bank, Nucleic Acids Research, Advance Access published on May 11, 1992, DOI 10.1093/nar/20.suppl.2019, Nucleic Acids Research. 20: 2019-2022.

[3] DDBJ- Tateno, Y & Miyazaki, S & Ota, M & Sugawara, H & Gojobori, T. (2000). DNA data bank of Japan (DDBJ) in collaboration with mass sequencing teams. Nucleic acids research, 28.

[4] EMBL- The EMBL Nucleotide Sequence Database, Guenter Stoesser, Mary Ann Moseley, Joanne Sleep, Michael McGowran, Maria Garcia-Pastor and Peter Sterk, Nucleic acids research, 26, 1998

[5] Europe's leading laboratory for basic research in molecular biology, www.ebi.ac.uk/embl

[6] Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne, The Protein Data Bank, Nucl. Acids Res. 28: 235-242.

[7] The PDB archive containing information about experimentally-determined structures of proteins, nucleic acids, and complex assemblies, www.pdb.org

[8] The National Center for Biotechnology Information provides access to biomedical and genomic information. http://www.ncbi.nlm.nih.gov/

[9] Setubal and J. Meidanis, "Introduction to Computational Molecular Biology", Cengage Learning, 1997

[10] J Cheng, A Tegge and P Baldi, "Machine learning methods for protein structure prediction", IEEE Reviews in Biomedical Engineering, Vol I, 2008.

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[11] K Mehrotra, C K Mohan, and S Ranka, Elements of Artificial Neural Networks, Penram International, 1997.

[12] S.S. Ray, S. Bandyopadhyay, P. Mitra and S.K. Pal, "Bioinformatics in neurocomputing framework", IEE Proc.-Circuits Devices Syst., Vol. 152, No. 5, October 2005.

[13] Canillo, H., and D. Lipman, The multiple sequence alignment problem in biology, SIAMJ Appl. Math, 1981. 48, 1073-1082.

[14] Database in fasta format- Uniref100.fasta, downloaded on 31st Jan 2010 from www.uniprot.org

ISBN: 978-988-18210-0-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)