Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 5, May 2013, pg.275 – 278

RESEARCH ARTICLE

FACTS INCLUSION BIOINFORMATICS: MODERN EFFORTS AND DISPUTE

Dr. Tryambak A. Hiwarkar[1], R. Sridhar Iyer[2]

[1]Associate Professor, Computer Science and Engineering, MBITM, Dongargarh (C.G.), India
[2]Research Scholar, Computer Science, CMJ University, Shillong, India

Abstract— A flood of data means that many of the challenges in biology are now challenges in computing. Bioinformatics, the application of computational techniques to analysis the information associated with bimolecular on a large-scale, has now firmly established itself as a discipline in molecular biology, and encompasses a wide range of subject areas from structural biology, genomics to gene expression studies.

The underlying motivation for many of the bioinformatics and DNA sequencing approaches is the evolution of organisms and the complexity of working with erroneous data. This article also describes the kind of software programs which were developed by the research community in order to (1) search, classify and mine different available biological databases; simulate biological experiments with and without errors.

I. INTRODUCTION

1.1 A Flood of DNA Sequence Data:

The initiate on of large-scale genomic research projects roughly a decade ago engendered an intensive effort to create related information management and analysis tools, largely driven by academic computer scientists associated with the institutions involved. One of the first and most important problems encountered was how to acquire, store and analyze massive amounts of DNA sequence information. Reliable, high-throughput sequencing methods perfected in the past few years are now churning out vast quantities of information from complete genomes of several bacteria and archaic(bacteria -like organisms that live in extreme conditions: a third kingdom of life) up to a mostly complete sequence of human chromosome 22, completed in late 1999.

1.2 DNA sequence reconstruction:

1.2.1 Sequencing scheme:

A SBH chip consists of a fixed numbers of features. Each feature can accommodate one probe. A probe is a string of symbols from the alphabet S={A,C,G,T,-}, where—denotes the 'blank' symbol. SBH provides information about k-mers present in the DNA string, but does not provide information about the positions of the k-mers. Moreover, SP is said to be the spectrum of sequence SEQ if SP is a multi-set of allk-long substrings of SEQ, assuming that the number of occurrences of each k-mer is also known. For example, SEQZATGCAGGTCC and SPZ{ATG,AGC,CAG,GCA,CGT,GTC,TCC,TGC}. A sequencing algorithm is an algorithm that, given a multi-set of k-mers SPZ {SP1, SPnKkC1}, decides if the spectrum defines a unique DNA sequence SEQ, and, if yes, reconstruct the sequence SEQ from its spectrum.[1]

1.2.2. Traditional solutions for the SBH problem:

The fundamental computational problem in SBH is the reconstruction of a sequence from its spectrum, the list of all k-mers that are included in the sequence along with their multiplicities. The traditional solutions for the SBH problem, which are briefly discussed in this paper, are the Hamiltonian path, also known as the Traveling Salesman Problem (TSP), the Eulerian path problem (EPP), and the positional SBH (PSBH).[1]

1.2.3. Hamiltonian path/TSP:

The SBH problem can be approached as a TSP (Blasewicz,Formanowicz, Kasprzak, Markiewicz, & Welglarz, 1999)by defining a directed graphG1 such as every occurrence of a k-mer in the spectrum is represented by a vertex in the graph, i.e. a k-mer that appears more than once is represented by multiple vertices. And every pair of vertices x,y €V are connected by a directed edge from x toy if and only if the k K 1 suffix of x is identical to the kK1 prefix of y. For example: {GTC,TCC} are 3-mers that are connected by a directed edge, since the 2-mer suffix of GTC equals the 2-mer prefix of TCC. Joining the two k-mers into a sequence is connected with a cost. The cost of joining two k-mers is equal to k-minus a number of nucleotides that overlap in these k-mers. For example, two k-mers CCATC and TCTAG may overlap on two nucleotides and create a longer sequence CCATCTAG. Consequently, a cost of joining them is equal to 3. The goal is to visit every vertex inG1exactly only once and return to the starting point in such a way that a sum of costs of traversed edges included in the G1 cycle is at its minimum. Hence, this problem is the same as finding a DNA sequence SEQ with the spectrum SP. This problem is known to be NP-hard, thus unlikely to admit a polynomial–time algorithm.[1]

1.2.4 EPP:

Pevzner (Pevzner, 1989) proposed a different approach, which reduces the SBH problem to the EPP, leading to a simple linear-time algorithm for sequence reconstruction. The idea is to construct a graph G2 (Pevzner's graph), whose edges correspond to k-mers and to find a path in the graph that visits every edge only once. Here the vertices are the full set of (kK1)-mer appearing in the spectrum. Based on the defined graph G2, the problem is translated in finding a path that visits all edges onG2. The solution is not necessarily unique because it is possible to detect a Eulerian cycle, which creates multiple ambiguous solutions. This ambiguity of a SBH solution occurs if it is impossible to reconstruct the original sequence SEQ from Pevzner's graph. Multiple (alternative) solutions which are manifested as branches in the graph, and unless the number of branches is very small, there is no good way to determine the correct sequence (Ben-dor, Peer, Shamir, & Sharan, 2001).

II. From Genes to Structure and Function

However, this is only the beginning. We need to know the structures of all the proteins (to create an 'atlas' to match the dictionary) and, most importantly, we need to progress to function. Genome-wide efforts to determine 3-D protein structures, or at least one representative 3-D structure for all protein families, are still in their infancy. There is no doubt that many aspects of structure determination could be scaled up – as was done in the sequence field – although the problems are larger and more expensive. Some projects have already started and there is little.

Doubt that more will follow. There-fore, we can expect, within ten years at the most, to have representative structures for most water-soluble protein domains (membrane proteins still defy routine crystallization).These structures will allow modeling of related sequences to provide structures for all genes. However, the relationship between sequence, structure and function is not straightforward. Proteins that are homologous (i.e. descended from a common ancestor) almost always adopt the same basic fold, but their functions, although usually the same or related might have changed during evolution. By contrast, un-related proteins can perform the same or similar functions .Elucidating the function of all genes in vivo will occupy biologists for many years to come, especially as the function could well be context sensitive. Thus, a major challenge for bioinformatics is to provide the tools to help in function identification prior to experimental verification. Structural data might be help full, but only by recognition of a homolog.

2.1 Intelligent systems in bioinformatics:

In the post-genome era, research in bioinformatics has been overwhelmed by the experimental data (Tan & Gilbert, 2003).The complexity of biological data ranges from simple strings(nucleotides and amino acids sequences) to complex graphs(biochemical networks; from 1D (sequence data) to 3D(protein and RNA structures). Considering the amount and complexity of the data, it is becoming impossible for an expert to compute and compare the entries with the current databases. Thus AI and machine learning techniques have been used to analyze biological data sets in order to discover and mine the patterns and similarities existing in various databases. Tan and Gilbert (2003) performed an empirical comparison of rule-based learning systems (decision trees, one rule, decision rules), statistical learning systems (naıve Bayes, instance based, SVM and artificial neural networks) and ensemble methods (stacking, bagging and boosting) on some available data of E. coli, Yeast, Promoters, and HIV. They have reported a comparison of different supervised machine learning techniques in classifying biological data. They also confirmed that none of the single methods could consistently perform well over all the data sets. Their work also showed that combined methods perform better than the individual ones. Kasturi and Acharya (2004) proposed an unsupervised machine-learning algorithm that identifies clusters of genes using combined data (promoter sequences of genes/DNA binding motifs, gene ontologism, and location data). The outcome of their experiments showed that the combined learning approach identified correlated genes effectively.

2.2 Data management in bioinformatics:

Since the advent of modern molecular biology, scientists have been building databases analyzing and documenting every conceivable aspect of the data (Conte et al., 2000; Laskowski, 2001). Most of these databases are at least partially maintained; human intelligence still plays a very important role in analysis. Recent developments in molecular biology have resulted in automated methods, which are capable of generating vast volumes of raw experimental data. Perhaps the best-known example of this phenomenon is DNA sequencing; many bioinformatics projects are dedicated to annotation and analysis of sequence data (Kulikova, Aldebert, Althorpe, Baker, Bates and Browne, 2004).Kumar, Palakal, Mukhopashyay, Stephens, and Li (2004) developed a knowledge base called BioMap using MEDLINE collection, which contains over 12 million citations and author abstracts from over 4600 biomedical journals. They have also presented an organization of a distributed database system to maintain the knowledge base of BioMap. Brecciaing, Fontana, and Busetta (2002) introduced a novel paradigm for providing knowledge based flexible query interfaces to object-oriented biological databases. The prototype they developed has the

Advantage in adopting a semantic approach with a capability of reasoning on the semantics of the queries. These features are aimed at improving the interaction between non-expert users and complex databases. Wang, Kuo, Chen, Hsiao, and Tsai (2005) described the framework of KSPF (Knowledge Sharing for Protein Families), which is applicable to all types of protein Families. Palakal, Mukhopashyay, and Mostafa (2002) designed and implemented an information management system prototype, called BioSifter, applied in the bioinformatics. Their tool was able to automatically retrieve relevant text documents from biological literature based on their interest profile. BLAST (Basic Alignment Search Tool) family of applications allows biologists to find homologous of an input sequence in DNA and protein sequence libraries (Altschul, Gish, Miller, Meyers, & Lipman, 1990). BLAST is an example of an application that has been enhanced as a web source, which provides dynamic access to large data sets.

2.3 Implications and challenges:

The applications and commercial ramifications of bioinformatics are considerable. In the past, computer experts have often been regarded as part of the service environment. In the future, the crucial management decisions on drug discovery pro-grams will be made by individuals who not only understand the biology but can also use the bioinformatics tools and the knowledge they release to develop hypotheses and identify quality targets. The data explosion modifies the old challenges in computational biology and presents new exciting prospects. As more sequences are determined, the identification of remote homolog's will become easier, as intermediate sequences will pro-vide the 'missing links' The long-term goal of predicting structure from sequence abilities will become more academic, because it will be possible to model most structures from a relative. The emphasis will therefore shift to understanding the principles and control of biological function and the interactions between molecules. Modeling cellular processes, such as signaling and metabolic pathways, will become increasingly important, especially as more proteomic data become available. Understanding and modeling function is essential to enable the rational design or modification of proteins or legends for new functions. In my view, this is the greatest challenge for bioinformatics in the next millennium.[2] [3].

III. BIOINFORMATICS TECHNOLOGY: AN EXAMPLE

Sequence an Interesting Genomic Region. You might start by finding the DNA sequence of a chromosomal region, speculating that it contains genes from an interesting biological pathway. Your ultimate goal might be to find an undiscovered drug target. To rapidly assemble a contiguous DNA sequence that might have one or more complete genes you might use the "shotgun" technique. This technique relies on piecing together many small, electrophoretically determined stretches of DNA sequence, each say 500 base pairs in length, into a much larger continuous stretch, say 2 million base pairs in length. To do this in a (mostly) automated fashion, you will need special programs like PHRED to read the raw DNA sequence, and PHRAP to assemble the small pieces into a large stretch of sequence. You will probably also need to use a laboratory information management system (LIMS) to track your sequencing project, as the process involves many individual samples and pieces of data that need to be stored and organized.[5]

3.1 Find Genes and Other Interesting Features in Your Genomic Sequence:

DNA sequence that is a string of several million symbols (like ...AAGGCTGAGTGCTAAGCGCGCG…), or a few strings of several hundred thousand symbols if you cannot put it all together (a common problem). You want to find regions that correspond to genes and perhaps regulatory sequences that control when the genes are turned on and off. You might start by using a program called BLAST (Basic Local Alignment Search Tool) to search the public or commercial DNA sequence databases to see if any stretches of your 2 million base pair sequence match previously identified gene sequences. To do this faster you might use special computer hardware known as an "accelerator," such as the DeCypher system from Time Logic.[4] You might use a more sophisticated software package like GENIE, GENSCAN or GRAIL to better identify where in your sequence the gene starts and stops, and where regulatory regions might be. These “gene finding” programs are not completely reliable in most cases, but are useful when used in conjunction with other methods. In this regard, you would also want to search public and private expressed sequence tag (EST) databases. ESTs are short sequences (several hundred base pairs) experimentally determined to correspond to real genes.[6] If an EST matches part of your sequence, it is likely that that part contains a real gene. This is a very powerful technique, as you don’t actually have to know what the gene does, and because available EST databases are now very comprehensive.

IV. CONCLUSION

The topics covered in this article are applications of AI in bioinformatics and DNA sequencing. The outcome of this research demonstrates the need for improving the existing tools, which are already being applied to extract important knowledge and to find out useful patterns from the massive amount of raw biological data.

REFERENCES

[1] Zoheir Ezziane “Applications of artificial intelligence in bioinformatics: A review” www.elsevier.com/locate/eswa

[2] Orengo,C.A.et al.(1997) Structure 5,1093–1108

[3] Martin,A.C.R.et al.(1998) Structure6,875–884

[4] Adleman, L. M. (1998).Location sensitive sequencing of DNA. Technical report, University of Southern California

[5] Altman, R. B. (2001). Challenges for intelligent systems in biology. IEEE Intelligent Systems, 16(6), 14–18.

[6] Altschul, S. F., Gish, W., Miller, W., Meyers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool.Journal of Molecular Biology, 215(3), 403–410