CHAPTER I

INTRODUCTION
CHAPTER I

INTRODUCTION

1.1. Introduction

Data mining means extracting knowledge from large amounts of data. Databases are rich with hidden information that can be used for making intelligent business decisions. Classification and prediction are two forms of data analysis that can be used to extract models describing important data classes or to predict data trends. Data mining is defined as "exploration and analysis by automatic and semi-automatic means, of large quantities of data in order to discover meaningful patterns and rules". Data mining for Bioinformatics will bring together researchers those interested in both fields, with the aim of generating new ideas and insights into how to tackle the challenge of floods of data in molecular biology.

In recent years, rapid developments in genomics and proteomics have generated a large amount of data. Often, drawing conclusions from these data requires sophisticated computational analyses. Bioinformatics, or computational biology, is the interdisciplinary science of interpreting biological data using information technology and computer science. The importance of this new field of
inquiry will grow as we continue to generate and integrate large quantities of genomic, proteomic, and other data.

Analysis of algorithms behind the most successful tools such as the local and global sequence alignment packages, and the underlying methods used in fragment assembly packages. Solution of complex biological questions is requiring modification of standard code. Computational methods used for searching, classifying, analyzing, and modeling protein sequences. Tools for analyzing DNA and RNA sequences can be developed with the data mining techniques. More advanced topics, such as genetic algorithms and simulated annealing, which can be used to address folding problems.

1.2. Bioinformatics

Bioinformatics is the application of computational tools and computer technologies to model, analyze, store, retrieve, manage, present, and visualize biological data. Primarily, the data to be processed are huge amounts of molecular biology data, such as DNA and protein sequences. Bioinformatics is a combination of biology and information technology and includes any computational tools and methods for managing, analyzing and manipulating large sets of biology data. Thus, computing technologies are vital for bioinformatics applications. For example, biology problems often require repeating the same task millions of times such as when searching for sequence similarities in existing
databases or comparing groups of sequences to determine evolutionary relationships.

Bioinformatics is an area where truly innovative, interdisciplinary education and collaboration are required in order for the field to progress. Bioinformatics brings together the fields of biology, computer science, and information technology to analyze biological data that have been collected over the past fifteen or more years and continue to be generated today. Many companies have started the process of mining this information and it has been estimated that bioinformatics could be a $2-billion industry within five years. The work that will be done by bioinformaticists in the 21st century will dramatically change health care and the process of biological discovery. Today's bioinformatics students must become expert users of existing tools and develop new tools to meet the challenges of this dynamic and rapidly changing field. Computer Science has become an important supporting area for many other disciplines. As such, the definition of computer science is broadening to include areas of emphases directly related to other disciplines [1]. Bioinformatics brings the fields of biology and Computer Science together for the analysis of biological data.

It is undeniable that, among the sciences, biology played a key role in the twentieth century. That role is likely to acquire further importance in the years to come. In the wake of the work of Watson and Crick, and the sequencing of the
human genome, far-reaching discoveries are constantly being made. One of the
central factors promoting the importance of biology is its relationship with
medicine. Fundamental progress in medicine depends on elucidating some of
perspective. This is not surprising since both biologists and computer scientists
have gone through years of education in their respective fields. A related plausible
reason for such differences is as follows: In computer science, we favor generality
and abstractions, our approach is often top-down, as if we were developing a
program or writing a manual.

In contrast, biologists often favor a bottom-up approach. This is
understandable because the minutiae are so important and biologists are often
involved with time-consuming experiments that may yield ambiguous results,
which in turn have to be resolved by further tests. The work of synthesis
eventually has to take place but, since in biology most of the rules have
exceptions, biologists are wary of generalizations.

1.3. Analogy with Computer Science Programs

It is possible to open a parenthesis to recall the relationship that exists
between computer programs and data; that relationship has analogies that are
applicable to understanding cell behavior. Not all biologists will agree with a
metaphor equating DNA to a computer program. Nevertheless, it is observed that
metaphor useful in explaining DNA to computer scientists.
In the universal Turing Machine (TM) model of computing, one does not distinguish between program and data—they coexist in the machine’s tape and it is the TM interpreter that is commanded to start computations at a given state examining a given element of the tape. Let us introduce the notion of interpretation in our simplified description of a single biological cell. Both DNA and proteins are components of our model, but the interactions that take place between DNA and other components (existing proteins) result in producing new proteins each of which has a specific function needed for cell survival (growth, metabolism, replication, and others).

The following statement is crucial to understanding the process of interpretation occurring within a cell. Let a gene $G$ in the DNA component be responsible for producing a protein $P$. Interpreters $I$ capable of processing any gene may well utilize $P$ as one of its components. This implies that if $P$ has not been assembled into the machinery of $I$ no interpretation takes place. Another instance in which $P$ cannot be produced is the already mentioned fact that another protein $P'$ may position itself at the beginning of gene $G$ and (temporarily) prevent the transcription.

The interpreter in the biological case is either one that already exists in a given cell (prior to cell replication) or else it can be assembled from proteins and RNA generated by specific genes (e.g., ribosomal genes). In biology the
interpreter can be viewed as a mechanical gadget that is made of moving parts that produce new components based on given templates (DNA or RNA). The construction of new components is made by subcomponents that happen to be in the vicinity. If they are not, interpretation cannot proceed. One can imagine a similar situation when interpreting computer programs (although it is unlikely to occur in actual interpreters). Assume that the components of \( I \) are first generated on the fly and once \( I \) is assembled (as data), control is transferred to the execution of \( I \) (as a program).

The above situation can be simulated in actual computers by utilizing concurrent processes that comprise a multitude of interrupts to control program execution. This could be implemented using interpreters that first test that all the components have been assembled: an execution proceeds only if that is the case; otherwise an interrupt takes place until the assembly is completed. Alternatively one can execute program parts as soon as they are produced and interrupt execution if a sequel has not yet been fully generated.

### 1.4. Molecular Components

The intent here is to show the importance three-dimensional structures have in understanding the behavior of a living cell. Cells in different organisms or within the same organism vary significantly in shape, size, and behavior. However, they all share common characteristics that are essential for life.
The cell is made up of molecular components, which can be viewed as 3D-structures of various shapes. These molecules can be quite large (like DNA molecules) or relatively small (like the proteins that make up the cell membrane). The membrane acts as a filter that controls the access of exterior elements and also allows certain molecules to exit the cell. Biological molecules in isolation usually maintain their structure; however, they may also contain articulations that allow movements of their subparts (thus, the interest of nanotechnology researchers in those molecules). The intracellular components are made of various types of molecules. Some of them navigate randomly within the media inside the membrane. Other molecules are attracted to each other.

In a living cell, the molecules interact with each other. By interaction it is meant that two or more molecules are combined to form one or more new molecules, that is, new 3D-structures with new shapes. Alternatively, as a result of an interaction, a molecule may be disassembled into smaller fragments. An interaction may also reflect mutual influence among molecules. These interactions are due to attractions and repulsions that take place at the atomic level. An important type of interaction involves catalysis, that is, the presence of a molecule that facilitates the interaction. These facilitators are called enzymes. Interactions amount to chemical reactions that change the energy level of the cell. A living cell
has to maintain its orderly state and this takes energy, which is supplied by surrounding light and nutrients.

It can be said that biological interactions frequently occur because of the shape and location of the cell's constituent molecules. In other words, the proximity of components and the shape of components trigger interactions. Life exists only when the interactions can take place. A cell grows because of the availability of external molecules (nutrients) that can penetrate the cell's membrane and participate in interactions with existing intracellular molecules. Some of those may also exit through the membrane. Thus, a cell is able to "digest" surrounding nutrients and produce other molecules that are able to exit through the cell's membrane. A metabolic pathway is a chain of molecular interactions involving enzymes. Signaling pathways are molecular interactions that enable communication through the cell's membrane. The notions of metabolic and signaling pathways will be useful in understanding gene regulation.

Cells, then, are capable of growing by absorbing outside nutrients. Copies of existing components are made by interactions among exterior and interior molecules. A living cell is thus capable of reproduction: this occurs when there are enough components in the original cell to produce a duplicate of the original cell, capable of acting independently.
So far, we have intuitively explained the concepts of growth, metabolism, and reproduction. These are some of the basic characteristics of living organisms. Other important characteristics of living cells are: motility, the capability of searching for nutrients, and eventually death. Notice that we assumed the initial existence of a cell, and it is fair to ask the question: how could one possibly have engineered such a contraption? The answer lies in evolution. When the cell duplicates it may introduce slight (random) changes in the structure of its components. If those changes extend the life span of the cell they tend to be incorporated in future generations. It is still unclear what ingredients made up the primordial living cell that eventually generated all other cells by the process of evolution.

The above description is very general and abstract. To make it more detailed one has to introduce the differences between the various components of a cell. Let us differentiate between two types of cell molecules: DNA and proteins. DNA can be viewed as a template for producing additional (duplicate) DNA and also for producing proteins.

Protein production is carried out using cascading transformations. In bacterial cells (called *prokaryotes*), RNA is first generated from DNA and proteins are produced from RNA. In a more developed type of cells (*eukaryotes*), there is an additional intermediate transformation: pre-RNA is generated from DNA, RNA
from pre-RNA, and proteins from RNA. Indeed, the present paragraph expresses what is known as the *central dogma* in molecular biology. Note that the above transformations are actually molecular interactions such as we had previously described. A transformation $A \rightarrow B$ means that the resulting molecules of $B$ are constructed anew using subcomponents that are "copies" of the existing molecules of $A$.

The last two paragraphs implicitly assume the existence of *processors* capable of effecting the transformations. Indeed that is the case with RNA-polymerases, spliceosomes, and ribosomes. These are marvelous machineries that are made themselves of proteins and RNA, *which in turn are produced from DNA!* They demonstrate the omnipresence of loops in cell behavior. One can summarize the molecular transformations that interest us using the notation:

![Diagram](image)

**Figure 1.1**

DNA Transformation
The arrows denote transformations and the entities below them indicate the processors responsible for carrying out the corresponding transformations. Some important remarks are in order:

1. All the constituents in the above transformation are three-dimensional structures.

2. It is more appropriate to consider a gene (a subpart of DNA) as the original material processed by RNA polymerase.

3. An arsenal of processors in the vicinity of the DNA molecule works on multiple genes simultaneously.

4. The proteins generated by various genes are used as constituents making up the various processors.

5. A generated protein may prevent (or accelerate) the production of other proteins. For example, a protein $P_i$ may place itself at the origin of gene $G_k$ and prevent $P_k$ from being produced. It is said that $P_i$ represses $P_k$. In other cases, the opposite occurs: one protein activates the production of another.

6. It is known that a splice some is capable of generating different RNAs (alternate splicing) and therefore the old notion that a given gene produces one given protein no longer holds true. As a matter of fact, a gene may produce several different proteins, though the mechanism of this is still a subject of research.

7. It is never repetitious to point out that in biology, most rules have exceptions
The term, gene expression, refers to the production of RNA by a given gene. Presumably, the amount of RNA generated by the various genes of an organism establishes an estimate of the corresponding protein levels. An important datum that can be obtained by laboratory experiments is an estimate of the simultaneous RNA production of thousands of genes. Gene expressions vary depending on a given state of the cell (e.g., starvation or lack of light, abnormal behavior, etc.).

1.5. Goals of Bioinformatics

The present role of bioinformatics is to aid biologists in gathering and processing genomic data to study protein function. Another important role is to aid researchers at pharmaceutical companies in making detailed studies of protein structures to facilitate drug design. Typical tasks done in bioinformatics include:

* Inferring a protein’s shape and function from a given a sequence of amino acids,
* Finding all the genes and proteins in a given genome,
* Determining sites in the protein structure where drug molecules can be attached.

To perform these tasks, one usually has to investigate homologous sequences or proteins for which genes have been determined and structures are available. Homology between two sequences (or structures) suggests that they have a common ancestor. Since those ancestors may well be extinct, one hopes that similarity at the sequence or structural level is a good indicator of homology.
It is important to keep in mind that sequence similarity does not always imply similarity in structure, and vice-versa. As a matter of fact, it is known that two fairly dissimilar sequences of amino acids may fold into similar 3D structures.

Nevertheless, the search for similarity is central to bioinformatics. When given a sequence (nucleotides or amino acids) one usually performs a search of similarity with databases that comprise all available genomes and known proteins. Usually, the search yields many sequences with varying degrees of similarities. It is up to the user to select those that may well turn out to be homologous.

1.6. Biological Data Available

It is observed that all the components of a living cell are 3D structures and that shape is crucial in understanding molecular interactions. A fundamental abstraction often done in biology is to replace the spatial 3D information specifying chemical bindings with a much simpler sequence of symbols: nucleotides or amino acids. In the case of DNA, we know that the helix is the underlying 3D structure.

Although it is much more convenient to deal with sequences of symbols than with complex 3D entities, the problem of shape determination remains a critical one in the case of RNA and proteins. The vast majority of the existing information has been obtained through sequencing, and it is expressible by
strings—that is, sequences of symbols. These sequences specify mostly nucleotides (genomic data) but there is also substantial information on sequences of amino acids.

Next in volume of available information are the results of micro array experiments. These can be viewed as very large usually dense matrices of real numbers. These matrices may have thousands of rows and columns. And that is also the case of the sparse Boolean matrices describing protein interactions. The information about 3D structures of proteins pales in comparison to that available in sequence form. The protein database is the repository for all known three-dimensional protein structures.

In a recent search, it is observed that there are now about 26 billion base pairs representing the various genomes available in the server of the National Center for Biotechnology Information. Besides the human genome with about 3 billion bp, many other species have their complete genome available there. These include several bacteria (e.g., *E. Coli*) and higher organisms including yeast, worm, fruit fly, mouse, and plants (e.g., Arabidopsis).

The largest known gene in the NCBI server has about 20 million base pairs and the largest protein consists of about 34,000 amino acids. These figures give an idea of the lengths of the entities we have to deal with. In contrast, the PDB has a
catalogue of only 45,000 proteins specified by their 3D structure. These proteins
originate from various organisms. The relatively meager protein data shows the
enormous need of inferring protein shape from data available in the form of
sequences. This is one of the major tasks facing biologists. But many others lie
ahead.

There is information available about metabolic pathways in simple
organisms, and parts of those pathways are known for human cells. The
formidable task is to put all the available information together so that it can be
used to understand better the functioning of the human cell. That pursuit is called
*functional* genomics. The term, genomics, is used to denote the study of various
genomes as entities having similar contents. In the past few years other terms
ending with the suffixes-ome or -omics have been popularized. That explains
proteomics (the study of all the proteins of a genome), transcriptome, metabolome,
and so forth.

**1.7. Algorithms for Gene Function**

We recall that a major role of bioinformatics is to help infer gene function
from existing data. Since that data is varied, incomplete, noisy, and covers a
variety of organisms, one has to constantly resort to the biological principles of
evolution to filter out useful information. Based on the availability of the data and
goals, we now present the various algorithms that lead to a better understanding of
gene function. They can be summarized as follows:

(1) *Comparing Sequences.* Given the huge number of sequences available, there is
an urgent need to develop algorithms capable of comparing large numbers of long
sequences. These algorithms should allow the deletion, insertion, and
replacements of symbols representing nucleotides or amino acids, for such
transmutations occur in nature.

(2) *Constructing Evolutionary (Phylogenetic) Trees.*

These trees are often constructed after comparing sequences belonging to
different organisms. Trees group the sequences according to their degree of
similarity. They serve as a guide to reasoning about how these sequences have
been transformed through evolution. For example, they infer homology from
similarity, and may rule out erroneous assumptions that contradict known
evolutionary processes.

(3) Detecting Patterns in Sequences.

There are certain parts of DNA and amino acid sequences that need to be
detected. Two prime examples are the search for genes in DNA and the
determining of subcomponents of a sequence of amino acids (secondary structure).
There are several ways to perform these tasks. Many of them are based on
machine learning and include probabilistic grammars, or neural networks.

(4) Determining 3D Structures from Sequences.
The problems in bioinformatics that relate sequences to 3D structures are computationally difficult. The determination of RNA shape from sequences requires algorithms of cubic complexity. The inference of shapes of proteins from amino acid sequences remains an unsolved problem. (5) Inferring Cell Regulation. The function of a gene or protein is best described by its role in a metabolic or signaling pathway. Genes interact with each other; proteins can also prevent or assist in the production of other proteins. The available approximate models of cell regulation can be either discrete or continuous. One usually distinguishes between cell simulation and modeling. The latter amounts to inferring the former from experimental data (say micro arrays). This process is usually called reverse engineering.

(6) Determining Protein Function and Metabolic Pathways.

This is one of the most challenging areas of bioinformatics and for which there is not considerable data readily available. The objective here is to interpret human annotations for protein function and also to develop databases representing graphs that can be queried for the existence of nodes (specifying reactions) and paths (specifying sequences of reactions).

(7) Assembling DNA Fragments.

Fragments provided by sequencing machines are assembled using computers. The tricky part of that assemblage is that DNA has many repetitive regions and the same fragment may belong to different regions. The algorithms for assembling DNA are mostly used by large companies (like the former Celera).
(8) Using Script Languages.

Many of the above applications usage requires scripting that provides data for an application, receives it back, and then analyzes it. What differentiates bioinformatics problems from others is the huge size of the data and its (sometimes questionable) quality. That explains the need for approximate solutions. It should be remarked that several of the problems in bioinformatics are constrained optimization problems. The solution to those problems is usually computationally expensive. One of the efficient known methods in optimization is dynamic programming. That explains why this technique is often used in bioinformatics. Other approaches like branch and-bound are also used, but they are known to have higher complexity.

**DNA** is helix-shaped molecule whose constituents are two parallel strands of **nucleotides**. There are four types of nucleotides in DNA and they correspond to the letters A (for *adenine*), T (thymine), C (cytosine) and G (guanine). DNA is usually represented by sequences of these four nucleotides. This assumes that only one strand is considered; the second strand is always derivable from the first by pairing A’s with T’s and C’s with G’s and viceversa. That derivation is called finding the reverse complementary pair of a strand.

**Genes** are contiguous subparts of single stranded DNA that are templates for producing **proteins**. Genes can appear in either of the DNAs strands. The set of
all genes in a given organism is called the *genome* for that organism. The function of DNA material between genes is largely unknown. Certain intergenic regions of DNA (called *non coding*) are known to play a major role in cell regulation, the process that controls the production of proteins and their possible interactions with DNA.

Proteins are produced from DNA using three operations or transformations called *transcription, splicing, and translation*. In humans and higher species (*eukaryotes*) the genes are only a minute part of the total DNA that exists in a cell. It is known that chromosomes are compact chains of coiled DNA. In more rudimentary types of cells that do not have a nucleus (*prokaryotes*), the phase of *splicing* does not occur.

DNA is capable of replicating itself. The cell machinery that performs that task is called *DNA-polymerase*. Biologists call the capability of DNA for replication and undergoing the above three (or two) transformations the *central dogma*.

- Genes are *transcribed* into *pre-RNA* by a complex ensemble of molecules called *RNA-polymerase*. During transcription the nucleotide T (*thymine*) is substituted by another one designated by the letter U (for *uracil*). Pre-RNA can be represented by alternations of sequence segments called *exons* and *introns*. The
exons represent the parts of pre-RNA that will be *expressed*, that is, translated into proteins. Next comes the operation called *splicing*; an ensemble of proteins called the *spliceosome* performs it. Splicing consists of concatenating the exons and excising the introns to form what is known as *mRNA*, or simply RNA.

The final phase, called *translation*, is essentially a "table look-up" performed by complex molecules called *ribosomes* (an ensemble of RNA and proteins). Translation repeatedly considers a triplet of consecutive nucleotides in RNA and produces one corresponding amino acid. The triplet is called a *codon*. In RNA, there is one special codon called a *start codon* and a few others called the *stop codons*. An *open reading frame* (ORF) is a sequence of codons starting with a start codon and ending with an end codon. The ORF is thus the sequence of nucleotides that is used by the ribosome to produce the sequence of amino acids that makes up a protein.

There are basically 20 amino acids but, in certain rare situations, others can be added to that list. Since there are 64 different codons and 20 amino acids, the "table look-up" for translating each codon into an amino acid is redundant in the sense that multiple codons can produce the same amino acid. The "table" used by nature to perform translation is called the *genetic code*. Due to the redundancy of the genetic code, certain nucleotide changes in DNA may not alter the resulting protein. Once a protein is produced, it folds (most of the time) into a unique structure in 3D space.
In the 3D representation of a protein, one can distinguish three different types of components: \(\alpha\)-helices, \(\beta\)-sheets and coils. The secondary structure of a protein is its sequence of amino acids, annotated to distinguish the boundaries of each component: helices, sheets, and coils. The tertiary structure of a protein is its 3D representation.

The function of a protein is the way it participates with other proteins and molecules in keeping the cell alive and interacting with its environment. Function is closely related to tertiary structure. In functional genomics, one studies the function of all the proteins of a genome. One of the important goals of bioinformatics is to help biologists in deciphering the function of proteins.

This thesis is conceived and organized on the following lines:

- The Chapter II highlights the review of related literature and the problem definitions.
- The Chapter III explains the methodologies used in our work.
- The Chapter IV consists of result and discussion of our thesis work.
- The Chapter V concludes by summarizing the work undertaken and provides advantages and future scopes.