CHAPTER II

BACKGROUND KNOWLEDGE

An introduction of some molecular biology that are useful as a background for the reader to understand the context of this work will be described in Section 1. A brief introduction to promoter sequences and features of promoter are included in Section 2 and 3. In the last section, I review the basic of an Artificial Neural Networks (ANNs).

2.1 Molecular biology and Genetics

The fact of biological sequences DNA, RNA and Proteins being involved in the most important cell processes has led to a growing interest in their analysis, with different approaches arising from various scientific fields. These molecules have a fundamental role, defining almost all cells activities.

2.1.1 DNA and RNA

Deoxyribonucleic acid (DNA) is the basic information macromolecule of cells. It is constituted by two chains of *nucleotides*, which are composed of *deoxyribose*, a pentose or five-carbon sugar molecule, linked to a *phosphate group* and to a nitrogen organic base of one of four types: adenine (A), guanine (G), cytosine (C) and thymine (T). Ribonucleic acid (RNA) is a single nucleotide strand exhibiting a similar composition, but with a different constituent sugar – the ribose – and uracil (U) instead of the thymine base.

Nucleotides in each DNA chain are connected by a chemical bond between the sugar of one nucleotide and the phosphate group of the adjacent one. When two DNA strands establish hydrogen bonds between their bases, with standard Watson-Crick pairing A–T and C–G, the classic double-helix is formed in a stable 3-dimensional structure as depicted in Figure 2.1.

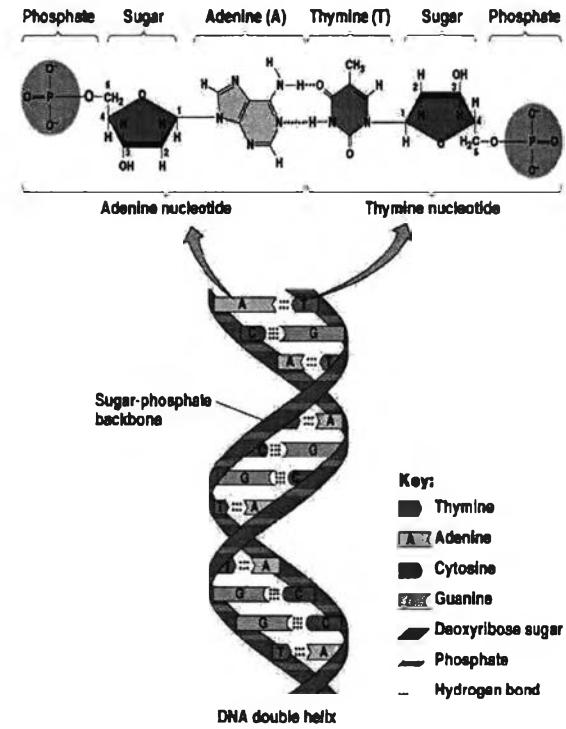

Figure 2.1: DNA structure and composition, formed by two complementary antiparallel chains of nucleotides.

2.1.2 Gene and Chromosome

Each DNA molecule is packaged in a separate *chromosome*. The total genetic information stored in the chromosomes of an organism is said to constitute its *genome*. With few exceptions, every cell of a Eukaryotic multi-cellular organism contains a complete

set of the genome, while the difference in functionality of cells from different tissues is due the variable expression of the corresponding genes. The human genome contains about 3×10^9 base pairs (abbreviated *bp*), organized as 46 chromosomes - 22 different autosomal chromosome pairs, and two sex chromosomes: either XX or XY. The 24 different chromosomes range from 50×10^6 to 250×10^6 bp. The amount of DNA varies between different organisms. The organism *Amoeba dubia* (a single cell organism), for example, has more than 200 times DNA as human. The l iving organisms divide into two major groups: *Prokaryotes*, which are single-celled organisms with no cell nucleus, and *Eukaryotes*, which are higher level organisms, and their cells have nuclei. A gene is a region of DNA that controls a discrete hereditary characteristic, usually corresponding to a single mRNA carrying the information for constructing a protein. In 1977 molecular biologists discovered that most Eukariotic genes have their coding sequences, called *exons*, interrupted by non-coding sequences called *introns*. Human genes constitute exon approximately 2-3% of the DNA, leaving 97-98% of non-genic *junk* DNA. The role of the latter is yet unknown, however, experiments involving removal of these parts. Several theories have been suggested, such as physically fixing the DNA in its compressed position, preserving old genetic data, etc.

2.1.3 Synthesis of macromolecules and the central dogma

The main molecules involved in cell mechanisms have been described previously; this section is devoted to the explanation of how the information is passed from genes to genes and from genes to proteins. Protein synthesis is a fundamental process in which the information encoded in DNA is expressed and effectively passed to influence the cell structure and metabolism. It involves several steps, also mediated by enzymes, and where different types of RNA perform an important role. Figure 2.2 depicts some of the

mechanisms that occur in an eukaryotic cell and that will be briefly reviewed.

The *central dogma* states that information flow is from DNA → mRNA → protiens, according to the processes described below. This process is different for prokaryotic and for eukaryotic organisms and goes through the following phases (note that prokaryotes do not have the second phase):

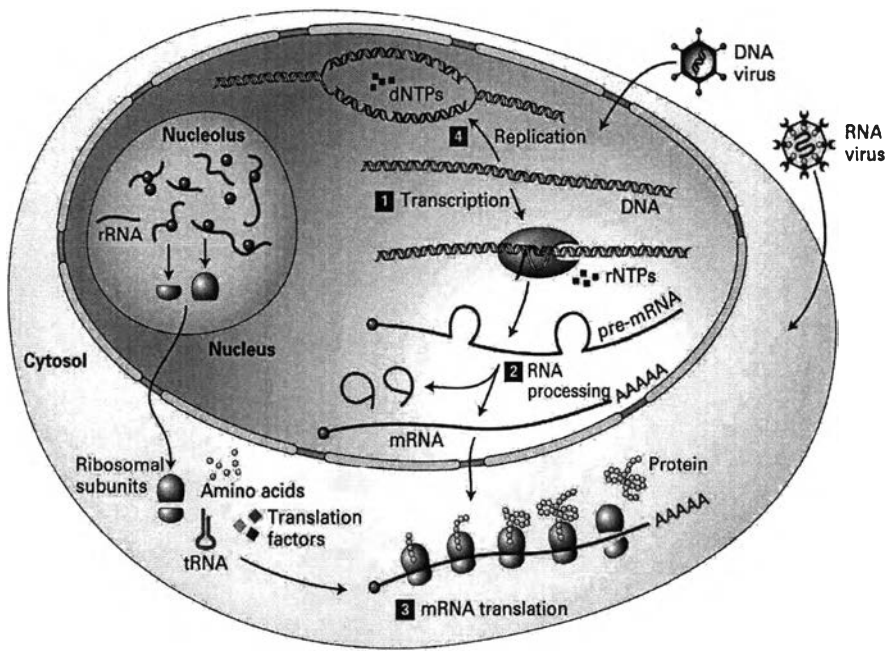

Figure 2.2: Macromolecule synthesis and other eukaryote cell mechanisms. Protein synthesis and DNA replication.

- Initially we have a phase of transcription, which is the process of synthesizing RNA under the control of the DNA template. The complementary section of the DNA sense strand is not transcribed. The resultant RNA is the direct complement of the DNA template, i.e., it is identical to the sense strand that corresponds to the transcribed section of the DNA template, with the exception that T is replaced by U. In prokaryotic organisms, the result of transcription is called messenger RNA (mRNA), while in the eukaryotic organisms the result of transcription is called

pre-mRNA.

- The second phase consists of the so-called RNA processing, which is characterized only for eukaryotic organisms. Its main purpose is to eliminate sections of the pre-mRNA molecules that correspond to the intron regions in the DNA template, splicing the sections that correspond to the consecutive exon regions in the DNA template, and consequently producing mRNA.

- The final phase is called translation. This is the process of the actual synthesis of a sequence of amino-acids based on information in mRNA.

2.1.4 The Genetic Code

The rules by which the nucleotide sequence of a gene is translated into the amino acid sequence of the corresponding protein, the *genetic code*, were described in the early 1960s. The sequence of nucleotides in the mRNA molecule, that acts as an intermediate was found to be read in serial order in groups of three. Each triplet of nucleotides, called a *codon*, specifies one *amino acid* (the basic unit of a protein, analogous to nucleotides in DNA). Since RNA is a linear polymer of four different nucleotides, there are $4^3 = 64$ possible codon triplets (See Figure 2.3). However, only 20 different amino acids are commonly found in proteins, so most amino acids are specified by several codons. In addition, 3 codons (of the 64) specify the end of translation, and are called *stop codons*. The codon specifying beginning of translation is AUG, and is also the codon for the amino acid Methionine. The code has been highly conserved during evolution: with a few minor exceptions, it is the same in organisms as diverse as bacteria, plants, and humans.

1st	2nd U	2nd C	2nd A	2nd G	3rd
U	Phe F	Ser S	Tyr Y	Cys C	U
					C
	Leu L		*stop*	*stop*	A
			stop	trp W	G
C	Leu L	Pro P	His H	Arg R	U
					C
			Gln G		A
					G
A	Ile I	Thr T	Asn N	Ser S	U
					C
	Met M		Lys K	Arg R	A
					G
G	Val V	Ala A	Asp D	Gly G	U
					C
			Glu E		A
					G

Figure 2.3: The genetic code, written by convention in the form in which the codons appear in mRNAs.

2.2 Basic introduction to promoter

There are two main classes of functional information encoded in the genomic DNA of every living organism. One class is the coding region, which specifies the structure and function of each gene product; another class is the regulatory region (occasionally, but very rarely, overlapping with a coding region), which controls and regulates when, where, and how the genes are expressed. A promoter is the most important regulatory region that controls and regulates the very first step of gene expression: mRNA transcription. The promoter is commonly referred to as the DNA region that is required to control and regulate the transcriptional initiation of the immediately downstream gene. In the process of the initiation of transcription, specific proteins, enzymes called RNA polymerases, attempt to bind to promoter regions. It is important to note that molecules of

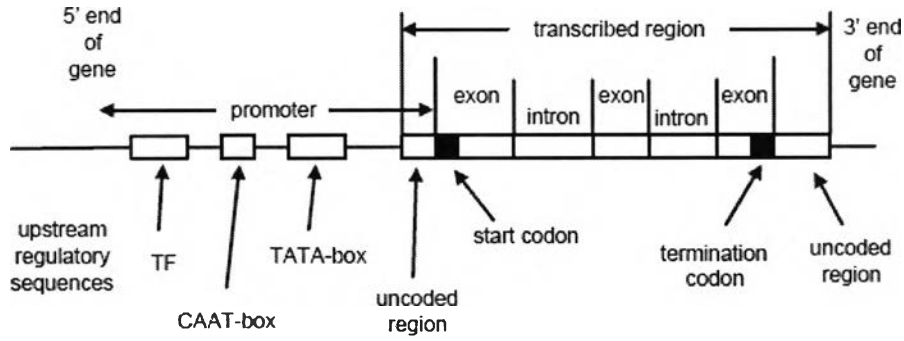

Figure 2.4: A possible promoter gene relation structure of mRNA eukaryotic gene.

RNA polymerase cannot recognize and bind to the promoter region directly. Before they can do this, it is necessary that some other proteins, called transcription factors (TFs), bind to specific subregions of promoters, called the transcription factor binding sites. Only then will RNA polymerase molecules be able to recognize the complex between the transcription factors and DNA, and to bind to the promoter. Promoters generally indicate and contain the starting point of transcription, the TSS, and regulate the rate of initiation of transcription. A promoter in eukaryotes is defined somewhat loosely as

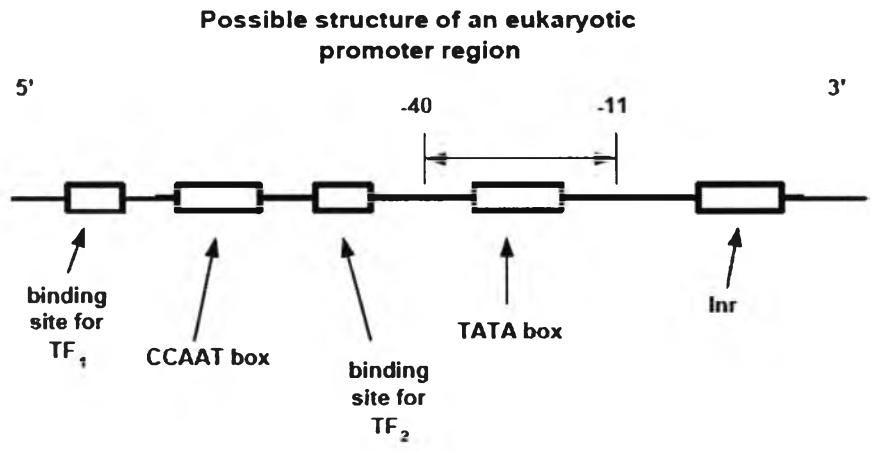

Figure 2.5: A possible structure of a Pol II promoter.

a portion of the DNA sequence around the transcription initiation site. Eukaryotic pro-

moters may contain different subregions such as TATA-box, CAAT-box, Inr, GC-box, together with other different transcription factors binding sites. The problem with these subregions in eukaryotic promoters is that they vary a lot, they may appear in different combinations, their relative locations with respect to the TSS are different for different promoters, and not all of these specific subregions need to exist in a particular promoter. The high complexity of eukaryotic organisms has led to specialization of the genes, so that promoters in eukaryotes are adjusted to their different conditions of expressions, for example in different cell types or tissues. Thus, the variability of internal eukaryotic promoter structure can be huge and the characteristics of the eukaryotic promoter are rather individual for the promoter than common for a larger promoter group. For this reason it is not easy to precisely define a promoter in eukaryotic organisms. And this is also one of the reasons why at this moment there is no adequate computer tool to accurately detect different types of promoters in a large-scale search through DNA databases, whilst at the same time being very selective and not producing a large number of false reporting. A possible model of an eukaryotic mRNA gene in relation to its promoter is given in Figure 2.4.

There are three types of RNA polymerase molecules in eukaryotes that bind to promoter regions. Our specific interest will be for RNA Polymerase II and their corresponding promoters whose associated genes provide codes for proteins. A typical arrangement of a Pol II promoter region may be as in Figure 2.5. The computer tools for the general mapping of DNA are aimed at large-scale scanning and searching of DNA databases so as to recognize and locate as many as possible different promoters, and not to make large numbers of false recognitions. Obviously, such tools cannot be based on highly specific structures of very narrow types of promoters.

2.3 The features of Promoter Sequences

Here I will show some significant features of promoter sequences which have been reported in some literatures. Some of these features are valid only in either prokaryotic or eukaryotic promoter sequences.

2.3.1 TATA-Box and TTG-Box

The *transcriptional elements*, which have high appearance frequency in the promoter region, are guessed to play an important role in transcriptions to find transcriptional elements from the promoter sequences. The transcription elements are the basic concept of finding the features of promoters. Experimentally, two identified transcriptional elements in promoter sequences are the -10 box and the -35 box. The -10 and -35 mean that these elements always appear around -10 and -35 positions (position of TSS is +1). The -10 box is a TATA-box and -35 box is the pattern of TTG called a TTG-box.

2.3.2 CpG Islands

CpG islands is another feature of promoter sequences. CpG islands means that in a sequence, the G nucleotide usually appears following the C nucleotide. The p in CpG denotes the phosphodiester linkage of the DNA sequence. In fact, a DNA sequence is always methylated around the TSS, so CpG islands have high appearance frequency in the promoter in all DNA sequences. This feature is found in eukaryotic promoter sequences. No significant CpG islands have been observed in prokaryote. So this feature cannot help us in the promoter prediction with *E.coli*.

2.4 Artificial Neural Networks (ANNs)

2.4.1 What is a Neural Network?

An Artificial Neural Network (ANN) is a machine-learning technique that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements (neurones) working in unit to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurones.

2.4.2 How does Human Brain Learn?

Much is still unknown about how the brain trains itself to process information. In the human brain, a typical neuron collects signals from others through a host of fine structures called *dendrites*. The neuron sends out spikes of electrical activity through a long, thin stand known as an *axon*, which splits into thousands of branches. At the end of each branch, a structure called a *synapse* converts the activity from the axon into electrical effects that inhibit or excite in the connected neurones. When a neuron receives excitatory input that is sufficiently large compared with its inhibitory input, it sends a spike of electrical activity down its axon. Learning occurs by changing the effectiveness of the synapses so that the influence of one neuron on another changes (Figure 2.6).

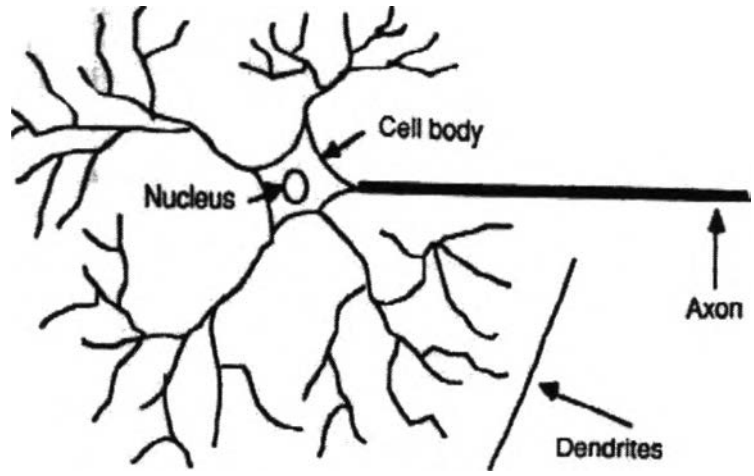

Figure 2.6: Component of Human Neurons

2.4.3 From Human Neurones to Artificial Neurones

It conduct these neural networks by first trying to deduce the essential features of neurones and their interconnections. Then typically program a computer to simulate these features. Due to incomplete knowledge of neurones and limited computing power, ANNs models are necessarily gross idealizations of real networks of neurones as demonstrated Figure 2.7.

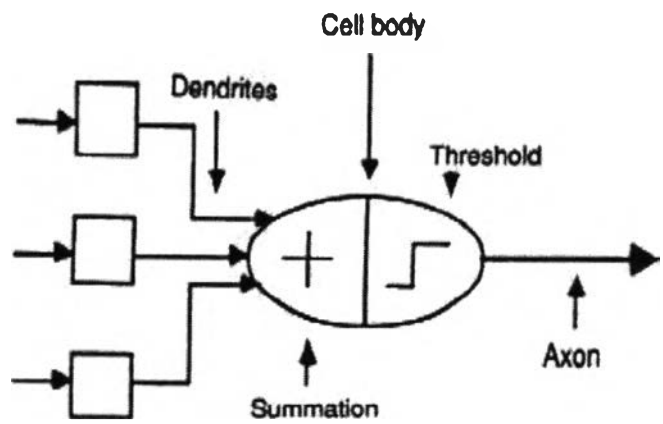

Figure 2.7: Components of Artificial Neural Network

2.4.4 Multi-layer feedforward and Backpropagation Neural Networks

This class of networks consists of multiple layers of computational units, usually interconnected in a feed-forward way. Each neuron in one layer has directed connections to the neurons of the subsequent layer (Figure 2.8). In many applications, the units of these networks apply a sigmoid function as an activation function. Multi-layer networks use a variety of learning techniques, the most popular one being *backpropagation*. Here the output values are compared with the correct answer to compute the value of some predefined error-function. By various techniques the error is then fed back through the network. Using this information, the algorithm adjusts the weights of each connection in order to reduce the value of the error function by some small amount. After repeating this process for a sufficiently large number of training cycles, the network will usually converge to some state where the error of the calculations is small. In this case, one says that the network has learned a certain target function. In principle, backpropagation provides a way to train networks with any number of hidden units arranged in any number of layers. In fact, the network does not have to be organized in layers - any pattern of connectivity that permits a *partial ordering* of the nodes from input to output is allowed. In other words, there must be a way to order the units such that all connections go from "earlier" (closer to the input) to "later" ones (closer to the output). This is equivalent to stating that their connection pattern must not contain any cycles. Networks that respect this constraint are called $feed\ forward$ networks.

The Algorithm

We want to train a multi-layer feedforward network by gradient descent to approximate an unknown function, based on some training data consisting of pairs (x, t). The vector x represents a pattern of input to the network, and the vector t the corresponding $target$

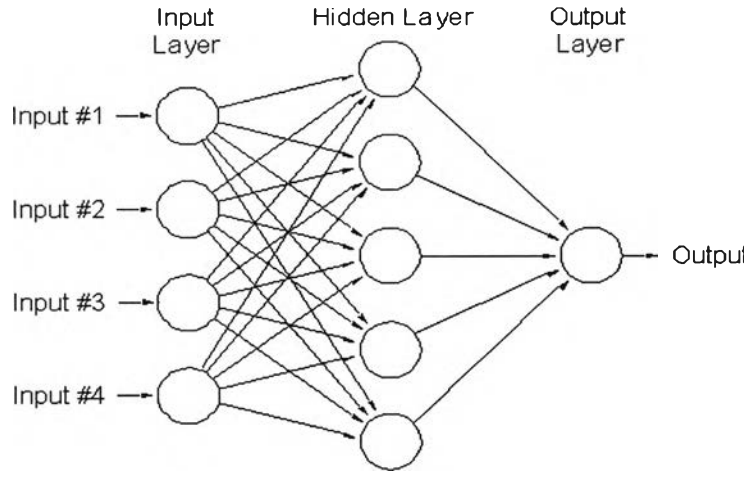

Figure 2.8: Structure of multi-layer networks

(desired output). The overall gradient with respect to the entire training set is just the sum of the gradients for each pattern; in what follows we will therefore describe how to compute the gradient for just a single training pattern. As before, we will number the units, and denote the weight from unit j to unit i by w_{ij}.

1. **Definitions**:

   - the *error* signal for unit j: $\delta_j = -\partial E/\partial net_j$
   - the (negative) *gradient* for weight w_{ij}: $\Delta w_{ij} = -\partial E/\partial w_{ij}$
   - the set of nodes *anterior* to unit i: $A_i = \{j : \exists w_{ij}\}$
   - the set of nodes *posterior* to unit j: $P_j = \{i : \exists w_{ij}\}$

2. **The gradient**. As we did for linear networks before, we expand the gradient into two factors using of the chain rule:

$$\Delta w_{ij} = -\frac{\partial E}{\partial net_i}\frac{\partial net_i}{\partial w_{ij}}$$

   The first factor is the error of unit i. The second one is

$$\frac{\partial net_i}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \sum_{k \in A_i} w_{ik} y_k = y_i$$

Putting the two together, we get

$$\Delta w_{ij} = \delta_i y_j$$

To compute this gradient, we thus need to know the activity and the error for all relevant nodes in the network.

3. **Forward activation.** The activity of the input units is determined by the network's external input x. For all other units, the activity is propagated forwardly:

$$y_i = f_i(\sum_{j \in A_i} w_{ij} y_j)$$

Note that before the activity of unit i can be calculated, the activity of all its anterior nodes (forming the set A_i) must be known. Since feedforward networks do not contain cycles, there is an ordering of nodes from input to output that respects this condition.

4. **Calculating output error.** Assuming that we are using the sum-squared loss

$$E = \frac{1}{2} \sum (t_o - y_o)^2$$

the error for output unit is simply

$$\delta_o = t_o - y_o$$

5. **Error backpropagation.** For hidden units, we must propagate the error back from the output nodes (hence the name of the algorithm). Again using the chain rule, we can expand the error of a hidden unit in terms of its posterior nodes:

$$\delta_j = -\sum_{i \in P_j} \frac{\partial E}{\partial net_i} \frac{\partial net_i}{\partial y_j} \frac{\partial y_j}{\partial net_j}$$

Of the three factors inside the sum, the first is just the error of node i. The second one is

$$\frac{\partial net_i}{\partial y_j} = \frac{\partial}{\partial y_j} \sum_{k \in A_i} w_{ik} y_k = w_{ij}$$

while the third factor is the derivative of node j's activation function

$$\frac{\partial y_j}{\partial net_j} = \frac{\partial f_j(net_j)}{\partial net_j} = f_j'(net_j)$$

For hidden unit h that uses the tanh activation function, we can make use of the special identity $tanh(u)' = 1 - tanh(u)^2$, giving us

$$f_h'(net_h) = 1 - y_h^2$$

Putting all the pieces together we get

$$\delta_j = f_j'(net_j) \sum_{i \in P_j} \delta_i w_{ij}$$

Note that in order to calculate the error for unit j, we must first know the error of all its posterior nodes (forming the set P_j). Again, as long as there are no cycles in the network, there is an ordering of nodes from the output back to the input that respects this condition. For example, we can simply use the reverse of the order in which activity was propagated forward.