Conditional Graphical Models for Protein Structure Prediction

Yan Liu

CMU-LTI-07-007

Language Technologies Institute
School of Computer Science
Carnegie Mellon University

Submitted in partial fulfillment of the requirements of the degree of Doctor of Philosophy.

Thesis Committee:
Jaime Carbonell (chair)
John Lafferty
Eric P. Xing
Vanathi Gopalakrishnan (University of Pittsburgh)

Abstract

Protein structures play key roles in determining protein functions, activities, stability and subcellular localization. However, it is extremely time-consuming and expensive to determine experimentally the structures for millions of proteins using current techniques. For instance, it may take months to crystalize a single protein. In this thesis, we design computational methods to predict protein structures from their sequences *in silico*. In particular, we focus on predicting structural topology (as opposed to specific coordinates of each atom) at different levels in the protein structure hierarchy. Specifically, given a protein sequence, our goal is to predict its secondary structure elements, how they arrange themselves in three-dimensional space, and how multiple chains associate with each other to form one stable structure. In other words, we strive to predict secondary, tertiary and quaternary protein structures from primary sequences and biophysical constraints.

In structural biology, traditional approaches for protein structure prediction are based on sequence similarities. They use string matching algorithms or generate probabilistic profile scores to find the most similar sequences in the protein database. These methods works well for simple structures with strongly conserved sequences, but fail when the structures are complex with many long-range interactions such as hydrogen and disulfide bonds among amino acids distant in sequence order. Moreover, evolution often preserves structures without preserving sequences. Hence structure prediction cannot rely just on sequence homology. These cases necessitate a more expressive model to capture the structural properties of proteins, and therefore developing a family of such predictive models is the core of this dissertation.

A new type of undirected graphical models are built based on *protein structure graphs*, whose nodes represent the state of either residues or a secondary structure element and whose edges represent interactions (e.g. bonds) either between adjacent nodes in the sequence order or long-range interactions among nodes in the primary sequence that fold back to establish proximity in 3D space. A discriminative learning approach is defined over these graphs, where the conditional probability of the states given the observed sequences is defined directly as exponential functions on local and topological features, without any assumptions regarding the data generation process. Thus our framework is able to capture the structural properties of proteins directly, including any overlapping or long-range interaction features. Within this framework, we develop conditional random fields and

kernel conditional random fields for protein secondary structure prediction; we extend these to create segmentation conditional random fields and chain graph model for tertiary fold recognition, and linked segmentation conditional random fields for quaternary fold prediction. These extensions are new contributions to machine learning, which enable direct modeling of long-distance interactions and enable scaling-up of conditional random fields to much larger complex structural prediction tasks.

With respect to computational biology, we contribute a novel and comprehensive paradigm for modeling and predicting secondary, super-secondary, tertiary and quaternary protein structures, surpassing the state of the art both in expressive power and predictive accuracy, as demonstrated in our suite of experiments. Moreover, we predict a large number of previously-unresolved beta-helical structures from the Swissprot data base, three of which have been subsequently confirmed via X-ray crystallography, and none have been disconfirmed. We hope that this work may shed light on the fundamental processes in protein structure modeling and may enable better processes for synthetic large-molecule drug design.

Acknowledgements

First and foremost, I would like to thank my advisor Prof. Jaime Carbonell for his great guidance and timely support over the past five years. I am greatly thankful that he would take me as a student, teach me the research methodology, guide me in choosing interesting and influential research topics, and encourage me when I am stuck. I could not make this far without him.

I would also like to thank my other committee members, John Lafferty, Eric Xing and Vanathi Gopalakrishnan, for their invaluable assistance, feedback and patience at all stages of this thesis. Their criticisms, comments and advices are critical in making this thesis more accurate, more complete and clearer to read.

In addition, a special thanks to my collaborators, Peter Weigele and Jonathan King, at Massachusetts Institute of Technology. They introduce to me these exciting biology problems and provide very valuable information when I work on the model. Their trust and dedication in our collaboration make the thesis as it is now. Besides, I would like to thank other collaborators in the biology field, Judith Klein-Seetharaman, Ivet Bahar and James Conway. They bring me into the wonderland of biology and teach me many things that I cannot learn from the class. I am also greatly thankful for the tremendous help I get from many other faculty, Yiming Yang, Chris Langmead, Alex Hauptman, Jie Yang and Roni Rosenfeld.

Good friends are difficult to come by. I owe many thanks to my good friends, former CMU graduate students Rong Jin, Jerry Zhu, Luo Si, Jian Zhang, Yi Zhang, Hua Yu, and Chengxiang Zhai, as well as fellow graduate students Jie Lu, Fan Li, Joy Zhang, Jun Yang, Jiazhi Ou, Zhong Xiu, Ke Yang, Ting Liu, Qifa Ke, Alicia Tribble, Madhavi K. Ganapathiraju and many others, for their good friendship and support.

Last, but definitely not the least, I would also like to thank my family for their love and support without which I would not have survived the Ph.D. process.

Contents

List of Figures

List of Tables

Glossary

α-helix One type of protein secondary structure, a rod-shape peptide chain coiled to form a helix structure 25

β-sheet One type of protein secondary structure, in which two peptide strands aligned in the same direction (parallel β-sheet) or opposite direction (antiparallel β-sheet) and stabled by hydrogen bonds 25

amino acid The unit component of proteins, a small molecule that contain an amino group (NH_2), a carboxyl group (COOH), a hydrogen atom (H) and a side chain (R-group) attached to a central alpha carbon (C_α) 24

coil One type of protein secondary structure with irregular regions 25

domain Subsections of a protein that represent structurally independent regions 26

fold Identifiable arrangement of secondary structure elements, which appear repeatedly in different protein domains 26

motif The unit made up of only a few secondary structural elements and appear repeatedly in different protein domains 26

PDB	Protein Data Bank , a worldwide repository for the processing and distribution of experimentally solved 3-D structure data	27
primary structure	The linear sequence of a protein	24
protein	Linear polymers of amino acids, responsible for the essential structures and functions of the cells	24
quaternary structure	The stable association of multiple polypeptide chains resulting in an active unit	25
residue	The amino acid connected by the peptide bonds through a reaction of their respective carboxyl and amino groups	24
SCOP	Structural Classification of Proteins, a database aims to provide a detailed and comprehensive description of the structural and evolutionary relationships between all protein structures	27
secondary structure	The local conformation of the polypeptide chain, or intuitively as building blocks for its three-dimensional structures	25
tertiary structure	Global three-dimensional structure of one protein chain	25

Chapter 1

Introduction

Proteins are a chain of amino acids that fold into three-dimensional structures, making up a large portion of all living organisms and performing most of the important functions, such as catalysis of biochemical reactions, receptors for hormones and other signaling molecules, and formation of tissues and muscular fiber. It is widely believed that protein structures reveal important information about their functions. However, it is extremely time- and labor-consuming to determine the structures of a protein via laboratory experiments. For instance, it may take months of concerted efforts to crystalize a single protein in order to enable x-ray diffraction methods that determine its 3D structure. Since the amino-acid sequence of a protein ultimately determines its three-dimensional structures, it is essential to design effective computational methods to predict the structures from the sequences, which is the main task of the thesis.

In order to better characterize the structural properties of proteins, biologists define a structure hierarchy of four levels: the *primary structure* is simply the linear chain or sequence of amino acids that make up the protein; the *secondary structure* is the local conformation of amino acids into regular structures – there are three types of major secondary structures, known as α-helices, β-sheets and coils or loops; the *tertiary structure* is the global three-dimensional structure of an entire protein or a domain within a protein; and sometimes, multiple protein chains unite together via hydrogen bonds resulting in *quaternary structures*. There are several challenging subtasks in protein structure prediction, some of which have been studied intensively for decades (Venclovas et al., 2003; Bourne & Weissig, 2003). In this thesis, we focus on predicting structural topology at all levels in the protein structure hierarchy. In other words, given a protein primary sequence, we aim at predicting what the secondary structure elements are, how they arrange themselves in three-dimensional space, and how multiple chains associate with each other to form stable structures. Protein structural motifs (sometimes referred to as protein folds) are identifiable spatial arrangements of secondary structures, which correspond to a domain within a protein or the entire protein tertiary structure. Although there are millions of distinct proteins, biologists hypothesize that there are only about a thousand topologically distinct folds, and many folds have one or more proteins with known structural and functional properties. Hence, topological fold prediction is a powerful tool in inferring the structure and function of other proteins with shared folds.

The traditional approaches for protein structure prediction are based on sequence similarities. They use string matching algorithms (e.g. PSI-BLAST (Altschul et al., 1997)) or generate probabilistic profiles (e.g. profile hidden Markov model (Durbin et al., 1998; Krogh et al., 1994; Karplus et al., 1998)) to find the most similar sequences in the protein database. These methods work well for simple structures with strong sequence conservation, but fail when the protein structures are complex or the sequence conservation is poor due to long-term evolutionary divergence. Therefore, several sophisticated probabilistic models have been developed: Delcher *et al* introduce probabilistic causal networks for protein secondary structure modeling (Delcher et al., 1993); Schmidle *et al* propose a Bayesian segmentation model for protein secondary structure prediction (Schmidler et al., 2000); Yanover and Weiss apply an undirected graphical model to side-chain prediction using various approximate inference algorithms (Yanover & Weiss, 2002); Chu *et al* extend segmental semi-Markov model under the Bayesian framework to predict secondary structures (Chu et al., 2004). These models achieve partial success; however, they are still far from fully capturing the structural properties of proteins.

From a computational perspective, the task of protein structure prediction is an instance of a more general machine learning problem, known as the *segmentation and labeling for structured data*. Namely, the goal is to predict a label or a sequence of labels given a set of observations with corresponding to inherent structures. For example, predicting whether a web page is the homepage of a student or that of a faculty given the web content and hyperlinks that connect each other, segmenting the contour of a house given the pixel grid of an image and so on. Conditional graphical models defined over undirected graphs, such as conditional random fields (CRFs) (Lafferty et al., 2001) and maximum-margin Markov networks (Taskar et al., 2003), prove to be the most effective tools to solve this type of problem (Kumar

& Hebert, 2003; Pinto et al., 2003; Sha & Pereira, 2003). Therefore we follow and extend the graphical model approaches and develop a series of new models for protein structure prediction. These models can be seen as a significant extension of CRF by joint modeling the constraints between the structural components. The key questions we address are: how can we represent the structural (primarily topological) properties of proteins using graphical models? Given the foreseeable complexity of such models for entire proteins (hundreds or thousands of amino acids), how can we learn the parameters of the model and make inferences efficiently?

1.1 Summary of Thesis Work

In this thesis, we develop a series of conditional graphical models for protein structure prediction. Specifically, we define a special type of undirected graph, namely *protein structure graph*, whose nodes represent the topological structural elements (either individual residues or a secondary structure elements) and whose edges indicate either local or long range interactions (chemical bonding). The conditional probability of the labels given the observed sequences are defined directly as exponential functions of all the features (local properties, long-distance interaction, bio-physical constraints, etc.). In this way, our models are able to capture the short and long-range interactions that matter in a direct manner.

Within the framework, we develop the following models:

- *Conditional random fields* for protein secondary structure prediction and β-sheet identification: we explore several combination strategies to refine the scores from multiple prediction algorithms by considering structural properties. We achieve encouraging improvement compared with the state-of-art algorithms in this very-well-studied subproblem, with prediction accuracy improvements of 6-8% for the β-sheet prediction over the previous state of the art.

- *Kernel conditional random fields* for protein secondary structure prediction: we introduce the notion of kernels in CRF so that recent advances in classification theory and practice can be used and extended to structure prediction problems. We achieve an improvement of 30-50% in secondary-strucutre topological transition accuracy.

- *Segmentation conditional random fields* (SCRFs) for tertiary motif recognition and alignment prediction: since the structural components of a tertiary motif are the secondary structures (sequences of amino acids that conform to one of the secondary structure elements, instead of individual amino acids), we extend the CRF model to a new semi-Markov version. In other words, the new model assigns a label to a subsequence of amino acids rather than an individual one. As a result, it can capture the structural constraints or associations on the secondary structure level and have the convenience to incorporate any relevant features at this level. We apply the model to predict protein folds, such as the right-handed β-helix fold, an important motif in bacterial infection of plants and binding of antigens, and achieve significantly better results than the state-of-art methods. We also hypothesize new examples of the β-helix proteins, three of which have been confirmed by recent biological experiments, and none of which have been refuted.

- *Chain graph model* for predicting tertiary motifs with structural repeats: based on the repetitive patterns of the target motifs, we decompose the complex graphs from SCRFs into subgraphs, and then connect them using directed edges via the chain graph framework. This model can be seen as a trade-off between globally optimal modeling and a locally optimal one. It helps to reduce the computational cost, while achieving a close approximation to the global optimal solutions. Our experiments on the β-helix motif and leucine-rich repeats demonstrate that the chain graph model performs similarly as SCRFs in prediction accuracy while the running time has been reduced by a factor of 50.

- *Linked segmentation conditional random fields* for quaternary motif recognition and alignment prediction: we extend SCRFs to jointly model the chemical bonding between multiple sequences in order to capture both within-sequence and cross-sequence interactions in quaternary topological structures. Quaternary structure prediction has been too challenging for earlier approaches. Therefore our approach extends the state of the art to enable us to address this much larger predictive problem. However, since the complexity involved with quaternary structures is much greater than that with tertiary structures, we derive a reversible jump MCMC sampling algorithm for efficient inference in the resulting complex graphs. The experiment results on triple β-spirals and double-barrel trimer motif demonstrate the effectiveness of our model. Ours is the first computational method to successfully predict these two complex quaternary structures.

1.2 Thesis Statement

We hypothesize that *conditional graphical models are effective for protein structure prediction.* Specifically, they can provide an expressive framework to represent the structural patterns in protein structures and enable the use of local, long-range and background-knowledge informative features. With our new extensions and model parameter estimation methods, these new graphical models are able to solve the long-range interaction problem in the task of topological structural motif recognition, given basic biophysical constraints and a limited number of structurally-resolved training examples, despite lack of sequence homology among the proteins that conform to target structural motifs.

Based on the thesis work, we conclude that the statement holds in general. Specifically, we make three strong claims and two weaker ones:
Strong claims:

1. Conditional graphical models with our extensions have the representational power to capture structural properties for accurate protein structure prediction

2. Conditional graphical models provide the ability to incorporate any types of informative features for better protein structure prediction, including overlapping features, segment-level features as well as long-range interaction features.

3. Although the complexity of conditional graphical models grows exponentially with the effective tree-width of the induced graphs, model estimation can be reduced to a polynomial complexity with approximate inference algorithms (such as reverse-jump MCMC) or with the chain graph model.

Weaker claims:

1. Conditional graphical models are able to solve the long-range interaction problem in protein motif recognition (either tertiary or quaternary), if the following priors are answered by domain experts: What are the possible structural components? How are they arranged in three-dimensional space? Without such information, the models only have limited power to capture the long-range interactions.

2. To our best knowledge, conditional graphical models are the most *expressive and accurate* models currently available for protein structure

prediction. They also have the ability to explore alternative feature spaces via kernels. However, the final prediction accuracy is bounded by the availability of training data and general topological knowledge about protein structures.

1.3 Thesis Outline

In this thesis, our primary goal is to seek effective computational tools for protein structure prediction. In addition, we target the design and validation of novel models to best capture the properties of protein structures, rather than a naive application of existing algorithms, so that we contribute both algorithmically and biologically. Therefore, we organize the rest of the thesis as follows:

In Chapter 2, we give an introduction to protein structures and explain the relevant terminology in computational biology. Next, we provide a brief overview of some basic concepts in machine learning;

In Chapter 3, we survey the state of the art pertaining to structured prediction, including variants of the CRF model, its extensions, and its applications;

In Chapter 4, we define a general framework for conditional graphical models. It can be seen as a generalized model for all the algorithms we develop in the thesis. We discuss the novelty of the framework and comment on its relationship with other models.

In Chapter 5, we discuss possible solutions to efficient learning and inference for conditional graphical models including our extended and scaled-up versions.

In Chapter 6, we describe our methods for protein secondary structure prediction, and results obtained therefrom, including: (1) a comparison study of score combination using CRFs; (2) specialized β-sheet prediction algorithm using CRFs; (3) the kernel CRF model to explore alternative feature spaces via kernels.

In Chapter 7, we describe in detail our new structure prediction method: segmentation conditional random fields and their application in tertiary motif recognition and alignment prediction. Next, we discuss how to use chain graph model to decompose complex graphs into subunits and reduce computational cost.

In Chapter 8, we derive the new linked segmentation conditional random fields for quaternary motif recognition and present results from the first general purpose prediction method for quaternary structures.

In Chapter 9, we summarize the thesis work, state its major contributions and limitations, and finally hint at future directions.

1.4 Related Publications

Part of the thesis work have been published in major conferences of computational biology and machine learning. Below is an incomplete list:

Related publications of Chapter 6 include:

- Yan Liu, Jaime Carbonell, Judith Klein-Seetharaman, Vanathi Gopalakrishnan. *Comparison of Probabilistic Combination Methods for Protein Secondary Structure Prediction.* Bioinformatics. 2004 Nov 22;20(17):3099-107.

- Yan Liu, Jaime Carbonell, Judith Klein-Seetharaman, Vanathi Gopalakrishnan. *Prediction of Parallel and Antiparallel β-sheets using Conditional Random Fields.* Biological Language Conference (BLC'03), 2003.

- John Lafferty, Xiaojin Zhu, Yan Liu. *Kernel Conditional Random Fields: Representation and Clique Selection.* The Twenty-First International Conference on Machine Learning (ICML'04), 2004.

Related publications of Chapter 7 include:

- Yan Liu, Jaime Carbonell, Peter Weigele, Vanathi Gopalakrishnan.*Protein Fold Recognition Using Segmentation Conditional Random Fields (SCRFs).* In Journal of Computational Biology.

- Yan Liu, Eric Xing, Jaime Carbonell. *Predicting Protein Folds with Structural Repeats Using a Chain Graph Model.* In international conference on Machine Learning (ICML05), 2005.

- Yan Liu, Jaime Carbonell, Peter Weigele, Vanathi Gopalakrishnan. *Segmentation Conditional Random Fields (SCRFs): A New Approach for Protein Fold Recognition.* ACM International conference on Research in Computational Molecular Biology (RECOMB05), 2005.

Related publications of Chapter 8 include:

- Yan Liu, Jaime Carbonell, Vanathi Gopalakrishnan. *Linked Segmentation Conditional Random Fields for Protein Quaternary Fold Recognition.* To appear in International Joint Conferences on Artificial Intelligence (IJCAI), 2007.

Chapter 2

Background

Most of the essential structures and functions of the cells are realized by proteins, which are chains of amino acids with stable three-dimensional structures. A fundamental principle in all of the protein science is that protein functions are determined by their structures. However, it is extremely difficult to experimentally solve the structures of the proteins. Therefore how to predict the protein structures from sequences using computational methods remains one of the most fundamental problems in structural bioinformatics and has been extensive studied for decades (Venclovas et al., 2003; Bourne & Weissig, 2003).

2.1 Introduction to Protein Structures

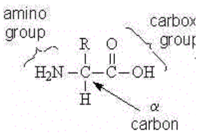
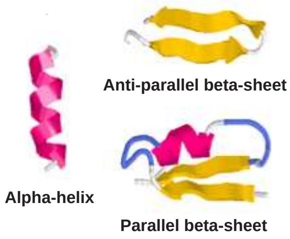
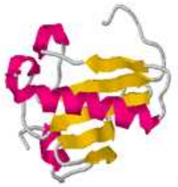
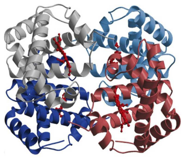
Before digging into the details of prediction algorithms, we start with introducing the common understanding of protein structures up to now and the knowledge databases built by the structure biologists over decades.

2.1.1 Protein Structures

In this section, we review the hierarchy definition of protein structures, domains and motifs, as well as the common classification for protein structures.

Protein structure hierarchy Proteins are linear polymers of amino acids. *Amino acids* are small molecules that contain an amino group (NH_2), a carboxyl group (COOH), a hydrogen atom (H) and a side chain (R-group) attached to a central alpha carbon (C_α) (Fig. 2.1). It is the side chain that

Figure 2.1: Protein structures hierarchy

Amino Acid	Secondary Structures	Tertiary Structures	Quaternary Structures
 amino group carbox grour R O H2N – C – C – OH H α carbon	 Anti-parallel beta-sheet Alpha-helix Parallel beta-sheet		

distinguishes one amino acid from another, resulting in 20 types of standard amino acids altogether. During a protein folding process, amino acids are connected by the chemical bonds through a reaction of their respective carboxyl and amino groups. These bonds are called *peptide bonds* and the amino acids linked by the peptide bonds are called *peptides*, or *residues* . The linear sequence of a protein is also referred to as its *primary structures*.

The *secondary structure* of a protein can be thought of as the local conformation of the polypeptide chain, or intuitively as building blocks for its three-dimensional structures. There are two types of secondary structures dominant in this local conformation: α*-helix*, a rod-shape peptide chain coiled to form a helix structure, and β*-sheets*, two peptide strands aligned in the same direction (parallel β-sheet) or opposite direction (antiparallel β-sheet) and stabled by hydrogen bonds (Fig. 2.1). These two structures exhibit a high degree of regularity and they are connected by the rest irregular regions, referred to as *coil* or *loop*.

The *tertiary structure* of a protein is often defined as the global three-dimensional structures and usually represented as a set of 3-D coordinates for each atoms. It is widely believed that the side-chain interactions ultimately determine how the secondary structures are combined to produce the final structure. An important property of the protein folding process is that protein sequences have been selected by the evolutionary process to achieve a reproducible and stable structure.

The *quaternary structure* is the stable association of multiple polypeptide chains resulting in an active unit. Not all proteins can exhibit quaternary structures. However, it is found that the quaternary structures are stabilized mainly by the same noncovalent interactions as tertiary structures, such as hydrogen bonding, van der Walls interactions and ionic bonding. In rare instances, disulfide bonds between cysteine residues in different polypeptide chains are also involved.

Domains, motifs and folds *Domains* are subsections of a protein that represent structurally independent regions, i.e. a domain would maintain its characteristic structure even if separated from the overall protein. In addition, every domain often performs a separate function from others. Therefore most protein structure prediction methods are focused on domains.

In biology, people have used the word "motif" in a number of areas with different meanings. In structural biology, *motifs*, or super-secondary structures, refer to the unit made up of only a few secondary structural elements and appear repeatedly in different protein domains.

Protein folds are identifiable arrangement of secondary structure elements, which appear repeatedly in different protein domains. The difference between motif and fold are subtle. Usually the motifs are short while the folds usually refer to the structure topology of the whole domains.

Protein structure classification Various ways have been proposed to classify the protein structures. One popular classification is achieved by considering the biochemical properties of the proteins. In this classification, proteins are grouped into three major groups: globular, membrane and fibrous. *Globular proteins* fold as a compact structure with hydrophobic cores and polar surfaces. Most proteins with known structures belong to this group since they are easier to crystalize due to the chemical properties. *Membrane proteins* exist in the cell membranes surrounded by a hydrophobic environment. Therefore they must retain a hydrophobic surface to be stable. Interestingly, recent research work suggests that membrane proteins share the same secondary structural elements and follow the same general folding principles as globular proteins despite their different properties. *Fibrous proteins* are often characterized by a number of repetitive amino acid sequences . Some of them consist of a single type of regular secondary structures while others are composed of repetitive atypical secondary structures. Membrane proteins and fibrous proteins are also referred to as *non-globular proteins*.

Another way to classify the protein structures are based on their predominant secondary structural elements, which results in four main groups: all α, all β, α/β (a mixture of α helix and β sheet interwoven by each other) and $\alpha + \beta$ (discrete α helix and β-sheet that are not interwoven). This kind of classification has be well studied in SCOP and CATH databases as described in the next section.

2.1.2 Databases of Protein Structures

The PDB (Protein Data Bank) was established in Brookhaven National Laboratories in 1971 as an archive for biological macromolecular crystal structures of proteins and nucleic acids (Berman et al., 2000). Until now, it is the single worldwide repository for the processing and distribution of experimentally solved 3-D structure data (40354 structures deposited by Nov, 2006).

The UniProt (Universal Protein Resource) is the world's most comprehensive catalog of information on proteins (3,656,820 entries in Release 9.2 by Nov, 2006) (Leinonen et al., 2004). It is a central repository of protein sequences and their functions created by combining the information from Swiss-Prot (databases of existing protein sequences with 243,975 entries), TrEMBL (databases of proteins translated from EMBL nucleotide sequence with 3,412,835 entries), and PIR (functional annotation of protein sequences).

The SCOP (Structural Classification of Proteins) database aims to provide a detailed and comprehensive description of the structural and evolutionary relationships between all protein structures by *manually* labeling (25973 entries by Jul, 2005) (Murzin et al., 1995). There are many levels defined in the classification hierarchy. The principal levels are *fold* for proteins with major structural similarity, *superfamily* for proteins with probable common evolutionary origin and *family* for proteins with clear evolutionary relationship (there are 945 folds, 1539 superfamilies and 2845 families by Jul, 2005).

The CATH database is a *semi-automatic* hierarchical domain classification of protein structures in PDB, whose crystal structures are solved to resolution better than 3.0 $\mathring{A}$ together with NMR structures (Orengo et al., 1997). There are four major levels in this hierarchy; class (by the secondary structure composition and packing), architecture (by the orientations of the secondary structures), topology (by connectivity of the secondary structures) and homologous superfamily (by sharing common ancestors).

2.2 Lab Experiments for Determining Protein Structures

There are different techniques to experimentally determine the protein structures, such as X-ray crystallography, Nuclear Magnetic Resonance, circular dichroism and Cryo-electron microscopy. However, most of these methods are time-consuming and labor-expensive.

In the Protein Data Bank, around 90% of the protein structures have been determined by *X-ray crystallography.* It makes use of the diffraction pattern of X-rays that are shot through a crystallized object. The diffraction is the result of an interaction with the high energy X-rays and the electrons in the atom. The pattern is determined by the electron density within the crystal. The major bottleneck for X-ray crystallography is the growth of protein crystals up to 1 mm in size from a highly purified protein source. This process usually takes months to years, and there exists no rules about the optimal conditions for a protein solution to result in a good protein crystal. X-ray structures are high resolution structures enabling the distinction of two points in space as close as 2Å apart.

Roughly 9% of the known protein structures have been obtained by *Nuclear Magnetic Resonance* (NMR) techniques. NMR measures the distances between atomic nuclei, rather than the electron density in a molecule. With NMR, a strong high frequency magnetic field stimulates atomic nuclei of the isotopes H-1, D-2, C-13, or N-15 (they have a magnetic spin) and measures the frequency of the magnetic field of the atomic nuclei during its oscillation period back to the initial state. In contrast to protein crystals required for X-ray diffraction, NMR makes use of protein solutions allowing for the determination of structures at very short time ranges. Consequently those flexible loop and domain structures could be solved successfully.

The proportion of the secondary structures can be determined via other biochemical techniques such as *circular dichroism* (CD), which is the differential absorption of left- and right-handed circularly polarized light. *Cryo-electron microscopy* (Cryo-EM) has recently become a means of determining protein structures to low resolution (less than 5 Å) and is anticipated to increase in power as a tool for high resolution work in the next decade. Until then, this technique remains a valuable resource for studying very large protein complexes such as virus coat proteins and amyloid fibers.

2.3 Profile Analysis of Protein Sequences

Profile analysis has long been a useful tool in finding and aligning distantly related sequences and in identifying known sequence domains in new sequences. A profile is the description of the consensus of a multiple sequence alignment. It uses a position-specific scoring matrix (PSSM) to capture information about the degree of conservation at various positions in the multiple alignment. Two most commonly used profile methods are PSI-BLAST (Altschul et al., 1997) and profile hidden-Markov model (Durbin et al., 1998).

PSI-BLAST BLAST is a program to find high scoring local alignments between a query sequence and a target database (Altschul et al., 1990). In PSI-BLAST, a profile is constructed automatically from a multiple alignment of the highest scoring hits in an initial BLAST search. Then the profile, instead of the query sequence, is used to perform another round BLAST search and the results of each iteration are used to refine the profile. In this way, PSI-BLAST improves the sensitivity of the searching and therefore is effective at detecting sequence hits with weak homology. On the other hand, one or two noisy sequences misplaced in early iterations might lead to a profile diverged far from the query sequence.

Profile hidden-Markov model Profile hidden-Markov model is a Markov chain model with position specific parameterizations of emission probabilities (Durbin et al., 1998). Specifically, a profile HMM has three states: "match", "delete" and "insert", in which the "match" state emits amino acids with probability according to the profile, the "insert" state emits amino acids with probability according to a background distribution, and the "delete" state is a non-emitting state corresponding to a gap in the profile (see Fig. 2.2). Compared with PSI-BLAST, profile HMMs have a formal probabilistic foundation behind the gap and insertion scores. More importantly, it solves the problem of position independent assumptions from PSI-BLAST by explicitly considering the transition probabilities in the model.

2.4 Previous Work on Protein Structure Prediction

The prediction of three-dimensional structures of a protein from its primary sequence is a fundamental and well-studied area in structural bioinformatics.

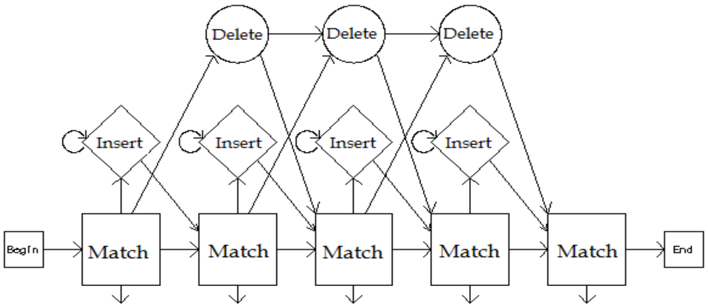


Figure 2.2: Profile hidden-Markov model: there are no transition from "delete" state to "insert" state. The "Begin" (B) and "end" states (E) are non-emitting states (Graph adapted from (Durbin et al., 1998))

Three main directions have been pursued to find better prediction methods for protein structures, including ab initio prediction, homolog modeling and fold recognition (Bourne & Weissig, 2003).

2.4.1 Ab Initio Methods

Ab initio structure prediction seeks to predict the native conformation of a protein from the amino acid sequence alone. The area is based on the beliefs that the native folding configuration of most proteins correspond to the lowest free energy of the sequence. Therefore the biggest challenge with regards to ab initio prediction is how to devise a free energy function that can distinguish native structures from incorrect non-native ones, as well as a search method to explore the huge conformational space. *Rosetta* is one of the most successful ab initio systems in recent years (http://robetta.bakerlab.org/). It is built upon accumulated domain knowledge of non-homologous sequences and their solved three-dimensional structures and then applies simulated annealing to create protein tertiary structures. However, the overall prediction accuracy using ab initio methods is still very low and a reliable free energy function is still under debate.

2.4.2 Fold Recognition (Threading)

Despite a good qualitative understanding of the physical forces in the folding process, present knowledge is not enough for direct prediction of protein structures from the first principle as in ab initio methods. An easier question is: which of the known folds in the databases are likely to be similar to the fold of a new protein given its primary sequence only. The problem stems from the fact that very often apparently unrelated proteins adopt similar folds. Therefore the main task in fold recognition is how to identify possible structural similarities even in the absence of sequence similarity. In general, threading works by computing a scoring function (usually based on free energy) that assesses the fit of a sequence against a given fold with the consideration of a pairwise atom contact and solvation terms. Since this is a combinatorial problem, the solutions can be extremely elaborate computationally, such as those involving double dynamic programming, dynamic programming with frozen approximation, Gibbs sampling, branch and bound heuristics, or as "simple" as a sequence alignment method such as profile hidden Markov models. The performance of fold recognition has been improved over years. However, in many cases the alignment of the query sequence to the structures are incorrect even when the fold has been corrected identified.

2.4.3 Homology Modeling

Homology modeling aims to predict the protein structures by exploiting the fact that evolutionarily related proteins with sequence similarity, as measured by the percentage of identical residues at each position based on an optimal structural superposition, share similar structures. This approach can be applied to any proteins that have more than 25-50% sequence identity to the proteins with known structures in the PDB. In practice, the homology modeling is a multi-step process that can be summarized in seven steps: template recognition and initial alignment, alignment correction, backbone generation, loop modeling, side-chain modeling, model optimization and model evaluation. At high sequence identities (60-95%), 90% of the comparative models can achieve an RMSD of less than 5 $\mathring{A}$ in regard to the experimentally determined structures. However, it is unreliable in predicting the conformations of insertions or deletions (the portions of the query sequence that do not align with the sequence of the template), as well as the details of side-chain positions.

2.5 Background of Machine Learning

Graphical models and discriminative models are the two major machine learning concepts related to the thesis work. Here we give a brief review of these two approaches.

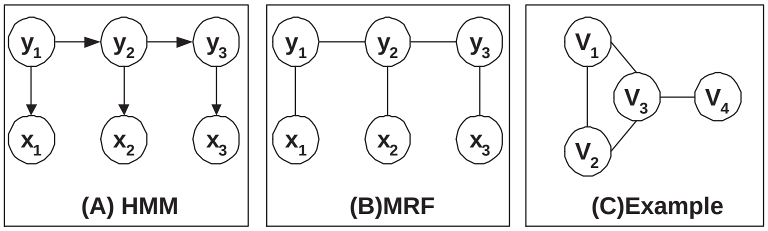


Figure 2.3: (A) graphical model representation of hidden Markov model; (B) graphical model representation of Markov random fields; (C) an example of undirected graph: (V_1, V_2, V_3) and (V_3, V_4) are maximal cliques

2.5.1 Graphical Models

Graphical models are a natural tool to deal with conditional probability distributions using graph theory. The nodes in the graph represent random variables (either observed or hidden), and the absence of arcs indicate conditional independence between random variables. The graphical model not only gives a compact representation of the joint probability distributions, but also provides inferential machinery for answering questions about probability distribution. The graph can be either directed, also known as Bayesian Networks or Belief Networks (BNs) or undirected, also called Markov Random Fields (MRFs) or Markov networks.

Directed graphical model A directed graph is a pair $G = \langle V, E \rangle$, where $V = \{V_i\}$ is a set of nodes and $E = \{(V_i, V_j) : i \neq j\}$ a set of edges with directions. We assume G is acyclic. Let V_i also refers to the random variable that the node V_i represents. Each node V_i has a set of parent nodes $pa(V_i)$. Since the structure of the graph defines the conditional independence relationship between random variables, the joint probability over all variables V can be calculated as the product of the conditional probability of each variable conditioned on its parents, i.e.

$$P(V) = \prod_{V_i \in V} P(V_i | pa(V_i)). \tag{2.1}$$

Hidden Markov models (HMMs) are one of the most popular directed graphical models for sequential data. Given an observable input $\mathbf{x} = x_1 x_2 \ldots x_N$, we want infer the state assignment (hidden) for each position $\mathbf{y} = y_1 y_2 \ldots y_N$. HMMs assume the first-order Markov assumption, i.e. the value of y_{i+1} is independent of y_{i-1} give the value of y_i. It also assumes the observation x_i

is independent of other states given the value of y_i. The graphical model representation of HMMs is shown in Fig.2.3 (A), and we can easy write out the joint probability as follows:

$$P(\mathbf{x},\,\mathbf{y}) = \prod_{i=1}^{N} P(x_i|y_i)P(y_i|y_{i-1}) \tag{2.2}$$

Undirected graphical model An undirected graphical model can also be represented by $G = \langle V, E\rangle$, except that the edges in E are undirected. As in the case of directed graphs, it is also desirable to obtain a "local" parametrization for undirected graphical models. A potential function ψ is any positive real-valued function associated with the possible realization v_c of the maximal clique c, where a maximal clique of a graph is a fully-connected subset of nodes that cannot be further extended (for example see Fig.2.3 (C)). It can be shown that the joint probability of the variables represented in the graph can be defined as the normalized product of the potential functions over all the maximal cliques in G, i.e.

$$P(V) = \frac{1}{Z}\prod_{c\in\mathcal{C}_G} \psi_c(V_c), \tag{2.3}$$

where $Z = \int_V \prod_{c\in\mathcal{C}_G} \psi_c(V_c)$ is the normalization factor.

Markov random field (MRF) in a chain is an undirected graphical model widely used for sequential data. Given the graph representation of MRF in Fig.2.2 (B), the joint probability of the data $\mathbf{x}$ and the labels $\mathbf{y}$ can be defined as

$$P(\mathbf{x},\,\mathbf{y}) = \frac{1}{Z}\prod_{i=1}^{N} \psi(y_i, y_{i-1})\psi(x_i, y_i), \tag{2.4}$$

By Hammersley-Clifford theorem, the potential function can be modeled as an exponential function of the features defined over the cliques (Hammersley & Clifford, 1971), i.e.

$$P(\mathbf{x},\,\mathbf{y}) = \frac{1}{Z}\prod_{i=1}^{N} \exp(\sum_{k=1}^{K_1} f(y_i, y_{i-1}))\exp(\sum_{k=1}^{K_2} f(x_i, y_i)), \tag{2.5}$$

where K_1 and K_2 are the number of features over the state-state cliques and state-observation cliques respectively.

Inference Algorithm Given a specific graphical model, the main task is to estimate the values of hidden (unobserved) nodes $\mathbf{Y}$ given the values of the observed nodes $\mathbf{X}$, i.e. $P(\mathbf{Y}|\mathbf{X})$. There are two major approaches to compute the target probabilities (marginal or conditional) in graphical models, including exact inference and approximate inference.

The elimination algorithm is the basic method for exact inference. The main idea is to efficiently marginalize out all the irrelevant variables using factored representation of the joint probability distribution. Consider the graph in Fig.2.2 (C), the probability $P(v_4)$ can be computed by

$$\begin{aligned}
P(v_4) &= \frac{1}{Z}\sum_{v_1}\sum_{v_2}\sum_{v_3}\psi(v_1,v_2,v_3)\psi(v_3,v_4) \\
&= \frac{1}{Z}\sum_{v_3}\psi(v_3,v_4)\sum_{v_1}\sum_{v_2}\psi(v_1,v_2,v_3) \\
&= \frac{1}{Z}\sum_{v_3}\psi(v_3,v_4)\sum_{v_1}m_2(v_1,v_3) \\
&= \frac{1}{Z}\sum_{v_3}\psi(v_3,v_4)m_1(v_3) \\
&= \frac{1}{Z}m_3(v_4).
\end{aligned}$$

The intermediate factors m_1, m_2 and m_3 can be seen as *messages* passing from the variables that have been integrated. When we want to compute several marginals at the same time, a dynamic programming can be applied to reuse some messages in the elimination algorithm. If the underlying graph is a tree, we can use *sum-of-product*, or belief propagation, which is a generalization of the forward-backward algorithm in HMMs (Rabiner, 1989). For a general graph, it has to be converted to into a clique tree by moralization and triangulation. After that, a local message passing algorithm can be applied, which could be either the sum-of-product algorithm or the junction tree algorithm, a variation designed for undirected models.

The computational complexity of the exact inference algorithms is exponential in the size of the largest cliques in the induced graph. For many cases, such as grids or fully connected graph, it is intractable to make exact inferences and therefore approximate algorithms, such as sampling, variational methods or loopy belief propagation, have to be applied. Sampling is a well-studied field in statistics and various sampling algorithms have been proposed. A very efficient approach for high dimensional data is

Markov Chain Monte Carlo (MCMC), which includes Gibbs sampling and Metropolis-Hastings sampling as special cases. *Variational methods* uses the convexity of the log function and iteratively updates the parameters so as to minimize the KL-divergence between the approximate and true probability distributions. *Loopy belief propagation* applies the original belief propagation algorithm to graphs even when they contain loops. There are no theoretical guarantees for convergence or whether the solution is optimal when it converges, however, the experimental results appear to be very successful (Murphy et al., 1999a).

Compared with exact inference, there are also some empirical problems with the approximate inference algorithm, for example, it might get trapped in the local optimal or never converge (within an affordable number of iterations). Therefore neither approach is dominant in the real applications. In order to make an efficient inference on a complex graph, we can combine these two approaches, for example, use exact inference algorithm locally within an overall sampling framework.

2.5.2 Generative Model v.s. Discriminative Model

For a supervised learning problem, there are two main types of models: generative models and discriminative models (Ng & Jordan, 2002). Discriminative models attempt to directly calculate the probability of the labels given the data, i.e., $P(y|x)$, while generative models alternatively estimate the class-conditional probability $P(x|y)$ as surrogates to find the maximal likely class based on Bayesian rules,

$$y^* = \arg\max_y P(y|x) = \arg\max_y \frac{P(x|y)P(y)}{P(x)}.$$

The success of generative models largely depends on the validity of the model assumptions. However, these assumptions are not always true, such as the Markov assumption in HMMs. In contrast, a discriminative model (e.g. logistic regression and support vector machines) typically makes less assumptions about the data and "let data speak for its own". It has been demonstrated more effective in many domains, such as text classification and information extraction. As pointed out by Vapnik (Vapnik, 1995), "one should solve the (classification) problem directly and never solve a more general problem (class-conditional) as an intermediate step". There are some empirical results showing that discriminative models tend to have a lower asymptotic error as the training set size increases (Ng & Jordan, 2002).

Chapter 3

Review of Structured Prediction

The breadth of tasks addressed by machine learning is expanding rapidly with the increase of vast amount of data available. The applications have varied from speech recognition, computer vision to natural language processing, computational biology, astronomy study, financial analysis and many other fascinating applications that change the life of people. With vast kinds of applications available, the machine learning fields have been extended to a number of new frontiers, one of which is the *prediction problem for structured outputs*, or succinctly as structured-prediction.

Structured prediction refers to the applications in which the observed data are sequential or with other simple structures while the output actually involve complex structures. For example, in protein structure prediction, we are given the observation as a sequence of amino acids, while the target output involves the complicated three-dimensional structures. Another example is the parsing problem in natural language processing, the input is one sentence, i.e. a sequence of words, and the output is a parsing tree. By considering the constraints or associations between outputs, we can achieve a better prediction performance. Those kinds of applications raises challenges to the I.I.D. (independently identically distributed) assumptions made by most statistical learning models and algorithms in previous study. In this chapter of the thesis, we provide an overview of current development on this topic, including a detailed discussion on the task description, an introduction to conditional random fields as well as its recent extensions, and finally its wide applications in different domains.

3.1 Prediction Problem with Structured Outputs

In supervised learning, we are given a set of training data with the observation x and the label y. Our goal is to learn a model f so that $f(x) \approx y$. There are two classical types of learning problem based on the value of y: one is classification problem, in which y takes discrete values in a predefined set (the simplest case is the binary classification, namely $y \in \{0, 1\}$); the other is the regression problem, in which y is a real-valued number. In either case, both x and y can be a vector although the dependency relations between the scalars are typically not explored.

In the prediction problem with structural outputs, the output y is a vector. Furthermore, the scalars of the vector y are not independent. They are either associated with others based on the locations, for example, the value of y_i is dependent on that of y_{i-1} and y_{i+1}; or they are associated based on type, for example, the value of y_i must be the same with that of y_{i-3}. Those types of constraints can be either deterministic (mostly referred as "associated") or probabilistic (referred as "correlated"). The essence of the structured prediction is to model these correlations or associations in one framework instead of treating them independently.

The prediction problem with structured outputs is closely related to the relational data. The study of relational data concerns itself with richly structured, involving entities of multiple types, which are related to each other through a network of different types of links. More specifically, the labels of different examples y_i are associated or correlated. Relational data mining has its roots in inductive logic programming, an area at the intersection of machine learning and programming languages. In addition, the structured prediction is also related to the multi-task learning, which aims at learning a task together with other related tasks at the same time via shared representation. This representation often leads to a better model for the main task, because it allows the learner to use the commonality among the tasks.

These three prediction problems are closely related, however, they differ significantly in both the task focus and principles for solutions. The structured prediction problem is initially extensively studied in speech recognition, later in computer vision, information extraction and computational biology. As discussed above, we have prior knowledge about the constraints or correlations between the elements of output vectors $\mathbf{y}$ (in most applications, the dependency relations are quite regular, for example, a chain or a grid). However, the outputs of different observations, $\mathbf{y}_i$ and $\mathbf{y}_j$, are treated independently in most algorithms for structured data. In contrast, the major subject in the relational data mining is to discover the relations between $\mathbf{y}_i$ and $\mathbf{y}_j$. In multi-task learning, existing approaches share the basic assumption that tasks are related to each other, that is, $y_i^{(k)}$ and $y_j^{(l)}$ are associated, while the assumptions about how they are associated vary from models to models.

The structured prediction problem also differs from the other two tasks in terms of principles for seeking solutions. More specifically, it usually finds the applications where the associations or relations are defined beforehand, either by domain knowledge or by assumptions. Then these relations can be easily represented by statistical graphical models, from early simple models, such as, hidden Markov model (Rabiner, 1989) and Markov random fields, later to maximum entropy Markov model (MEMM) (McCallum et al., 2000), and recently to the conditional random fields (CRF) (Lafferty et al., 2001). Other on-going research work along the directions include the study of alternative loss functions (Taskar et al., 2003; Altun et al., 2004; Tsochantaridis et al., 2004), efficient inference algorithms (Dietterich et al., 2004; Roth & Yih, 2005) as well as other extensions for broader applications (Kumar & Hebert, 2003; Sha & Pereira, 2003). The recent trend has shifted to imbalanced data and semi-supervised learning. The study of relational data, on the other hand, is originated from the relational database. Therefore the symbolic methods and first order logic algorithms are dominant in the solutions for relational data. In multi-task learning, the relatedness among tasks is hidden and to be uncovered. Based on the assumptions about how each tasks are associated, different models have been proposed, such as I.I.D tasks (Baxter, 2000), a Bayesian prior over tasks (Baxter, 2000; Heskes, 2000; Yu et al., 2005), linear mixing factors (Ghosn & Bengio, 2000; Teh et al., 2005), rotation plus shrinkage (Breiman & J, 1997) and structured regularization in kernel methods (Evgeniou et al., 2005).

3.2 Conditional Random Fields (CRF)

Conditional Random Fields (CRFs), first proposed by Lafferty et al., are *undirected* graphical models (also known as *random fields*) (Lafferty et al., 2001). It has been proven very effective in many applications with structured outputs, such as information extraction, image processing, parsing and so on (Pinto et al., 2003; Kumar & Hebert, 2003; Sha & Pereira, 2003). CRF has played an essential role in the recent development of structured prediction.

Before introducing the conditional random fields, we first review a simple model, the hidden Markov model(HMM), which is widely-known and has been applied to applications in many domains. HMM works by computing

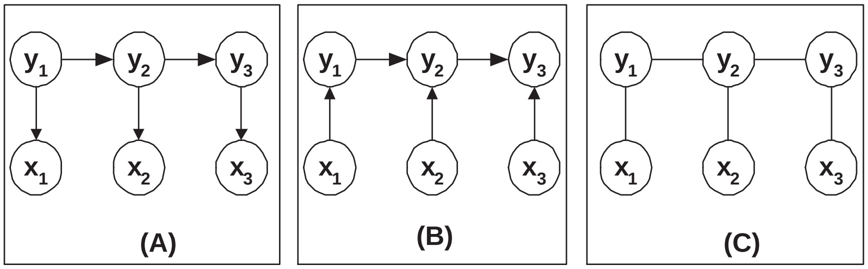


Figure 3.1: Graphical model representation of simple HMM(A), MEMM(B), and chain-structured CRF(C)

the joint distribution of observations $\mathbf{x}$ and states $\mathbf{y}$, $P(\mathbf{x}, \mathbf{y})$. The graphical model representation of HMMs is shown in Figure 3.1- (A). Two kinds of probability distributions have to be defined: (1) the transition probabilities $P(y_i|y_{i-1})$ and (2) the emission probabilities $P(x_i|y_i)$. By taking the first-order Markov assumption, i.e. $p(x_i|y_i) = p(x_i|y_i, y_{i-1})$, we have the joint probability as follows:

$$P(\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{n} P(x_i|y_i)P(y_i|y_{i-1}). \tag{3.1}$$

HMM has been very successful in applications such as speech recognition (Rabiner, 1989), and sequence analysis in bionformatics (Durbin et al., 1998). However, as a generative model, it has to assume a particular transition probability and emission probability, which results in many inconveniences if we use overlapping or long-range features. Therefore discriminative training models, for instance MEMM and CRF, are proposed.

The graphical model representation for a chain-structured CRF is shown in Figure 3.1, where we have one state assignment for each observation in the sequence. Specifically the conditional probability $P(\mathbf{y}|\mathbf{x})$ is defined as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \prod_{i=1}^{N} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i, y_{i-1}, y_i)), \tag{3.2}$$

where f_k can be arbitrary features, including overlapping or long-range interaction features. As a special case, we can construct an HMM-like model in which features can factorized as two parts, i.e. $f_k(\mathbf{x}, i, y_{i-1}, y_i) = g_k(\mathbf{x}, i)\delta(y_{i-1}, y_i)$, where $\delta(y_{i-1}, y_i)$ is the indicator function over each state pair (y_{i-1}, y_i) and $g_k(\mathbf{x}, i)$ for each state-observation pair $(\mathbf{x}, y_i)$.

CRF takes on a normalizer over the whole sequence, which results in a series of nice properties but at the same time introduces huge computational costs. Maximum Entropy Markov Models (MEMMs), proposed by McCallum et al, can be seen as a localized version of CRF (see Fig. 3.1-(B)). The conditional probability in MEMM is defined as

$$P(\mathbf{Y}|\mathbf{x}) = \prod_{i=1}^{N} \frac{1}{Z_i} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i, Y_{i-1}, Y_i)), \tag{3.3}$$

where Z_i is a normalizing factor only over the i^{th} position. MEMM reduces the computational costs dramatically, but at a cost suffers from the "label bias" problem, i.e. the total probability "received" by y_{i-1} must be passed on to labels y_i at time i even if x_i is completely incompatible with y_{i-1} (Lafferty et al., 2001). Empirical results show that for most applications CRF is able to outperform MEMM with either slight or significant improvement (Lafferty et al., 2001; Pinto et al., 2003).

3.3 Recent Development in Discriminative Graphical Models

The successes of the CRF model attract the interest of many researchers and various extensions of the model have been developed. From the machine learning perspective, recent enrichment of the CRF model includes the following: utilizing alternative loss functions, proposing efficient inference algorithms, extending to semi-Markov and segmented versions as well as Bayesian version.

3.3.1 Alternative Loss Function

The classification problem has been extensively studied in the past twenty years or so and many kinds of classifiers are proposed (Hastie et al., 2001). A unified view of the popular classifiers is that they belong to a generalized linear classifier family with different loss functions. For example, logistic regression uses the negative log-loss and support vector machine adopts the hinge loss. In the description of the original CRF model, a negative log-loss of the training data is used as the optimization criteria. Similar to the classification problem, other loss functions can be applied to the CRF formulation and result in various extensions, for example, the max-margin Markov networks, Gaussian process models, perceptron-like model as well as the Bayesian CRF. The detailed descriptions of these models are as follows:

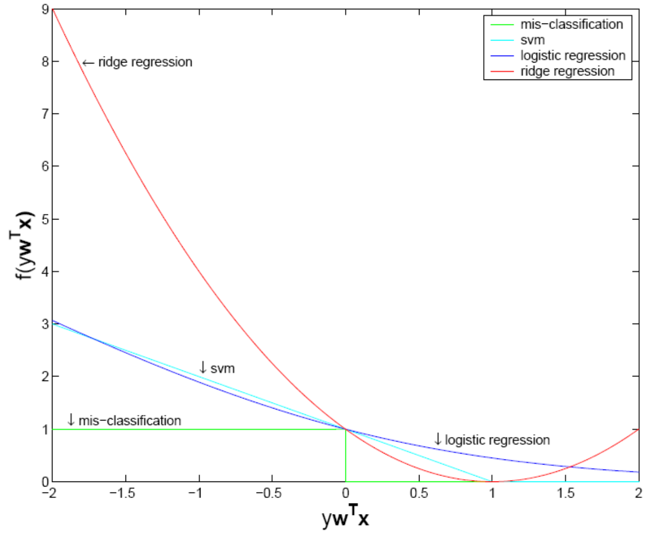


Figure 3.2: Loss functions of the logistic regression, support vector machines, ridge regression against the target 0-1 loss (Graph adapted from (Hastie et al., 2001))

Max-margin Markov networks Maximum margin Markov (M3) networks combines the graphical models with the discriminative setting as the support vector machines (SVM) (Taskar et al., 2003). As we know, SVM is a new generation learning system based on Structural Risk Minimization instead of Empirical Risk Minimization (Vapnik, 1995). It is both theoretically well-founded and practically effective. The primal form of SVM with linear kernel can be described as follows:

$$\min\{C\sum_{i=1}^{n}\xi_i + \frac{1}{2}w^T W\}, \text{ subject to:} y_i(w^T x_i + b) \leq 1 - \xi_i \text{ and } \xi_i \leq 0 \quad (3.4)$$

where ξ_i is the slack variable, C is a constant parameter. By using implicit constraints, we can transform the objective function as:

$$w^\star = \arg\min\{\frac{1}{n}\sum_{i=1}^{n}\max\{0, 1 - y_i(x^T x_i + b) + \lambda w^T w\}\}, \qquad (3.5)$$

where $\max\{0, 1 - y_i(x^T x_i + b)\}$ can be thought as the hinge loss for one instance, and the second term $\lambda w^T w$ is the regularization term. Due to the non-differentiable loss function, the fitting of SVM is usually solved in its dual form:

$$\max L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{i=N,j=N} \alpha_i \alpha_j y_i y_j \cdot \vec{x_i} \vec{x_j}$$

subject to

$$0 \leq \alpha_i \leq C_i, \text{ and } \sum_{i=1}^{i=N} \alpha_i y_i = 0.$$

The solution is given by $w^\star = \sum_{i=1}^{N} \alpha_i y_i x_i$, where α_i is Lagrange multipliers and C is a constant.

In M^{3}N, the goal is to learn a function h from the training data, so that $h(x) = \text{argmax}_y w^T f(x, y)$, where w^T is the model parameter and $f : \mathcal{X} \times \mathcal{Y}$ is the features or basis functions. Casting it into the classical SVM setting, we have the objective function

$$\max \gamma, \text{ subject to } \|w\| = 1; w^T \Delta f_(x, y) \geq \gamma \Delta h(x, y), \tag{3.6}$$

where $\Delta h(x, y) = \sum_{i=1}^{l} I(y_i \neq h(x)_i)$ and $\Delta f(x, y) = f(x, h(x)) - f(x, y)$. In (Taskar et al., 2003), a coordinate descent method analogous to the sequential minimal optimization (SMO) is used to seek the solutions to eq (3.6). Later, Taskar et al. formulate the estimation problem as a convex-concave saddle-point problem and apply the extragradient method (Taskar & Simon Lacoste-Julien, 2005). This yields an algorithm with linear convergence using simple gradient and projection calculations.

The M^{3}N model has the advantage to incorporate kernels inherited from the SVM formulation, which efficiently deal with the high-dimensional features. Furthermore, it can capture correlations in structured data with multiple formulations, either Markov networks, context free grammars, or combinatorial structures (Taskar, 2004) [1].

Support vector machines for structured output space In (Tsochantaridis et al., 2004), a support vector machine for structured outputs is developed. The idea is similar to the M^3N model discussed above, which uses the hinge loss for optimization criterion as in SVM. However, these two models differ significantly in the optimization algorithms to solve the induced quadratic-programming problems. In (Tsochantaridis et al., 2004), the problem is solved by a cutting plane algorithm that exploits the sparseness and structural decomposition of the objective function.

[1] There is also some concern about the consistency of the framework, that is, the solutions provided to the M^3N model are not the same as the optimization function originally stated in definition of the model.

Gaussian Process Models In addition to the SVM-style approaches, a Gaussian process model for segmenting and annotating sequences is developed (Altun et al., 2004). It generalizes the Gaussian Process (GP) classification by taking dependencies between neighboring labels into account. In the definition of the original GP classifier, we construct a two-stage model for the conditional probability distribution $p(y|\vec{x})$, by introducing an intermediate, unobserved stochastic process $(u(\vec{x}; y))$. Via some derivation, we have the objective function as follows

$$R(w|x, y) = w^T K w - \sum_{i=1}^{n} w^T K e_{(i,y_i)} + \sum_{i=1}^{n} \log \sum_{y} \exp(w^T K e_{(i,y)}), \quad (3.7)$$

where K is the kernel matrix. From eq (3.7), we can see that the GP classifier is very similar to the kernel logistic regression. In sequence labeling problem, we need to consider the labels for one sequences jointly. Therefore in (Altun et al., 2004), the kernel is defined as $k = k^1 + k^2$, where k^1 couples observations in both sequences that are classified with the same micro-labels at respective positions, k^2 simply counts the number of consecutive label pairs that both label sequences have in common. Two approaches are developed to seek the solutions for the optimization problem defined in eq(3.7). One is the dense algorithm, which involves the computation of Hessian matrix, the other is the sparse algorithm, which is similar to the greedy clique selection algorithm discussed in (Lafferty et al., 2004).

Perceptron CRF Motivated by the efficiency and consistency of the perceptron algorithm, a perceptron-like discriminative model for predicting structured outputs is introduced (Collins, 2002). The algorithm makes inferences on chain-structured graphs via Viterbi decoding of training examples, combined with simple additive updates. A theory is also provided to justify the convergence of the modified algorithm for parameter estimation, including both the voted version and averaged version. It has been shown that the perceptron-like CRF model performs similar as the original CRF empirically, while at the same time enjoys a less complex learning algorithm.

Bayesian conditional random fields Bayesian conditional random fields are a Bayesian approach to learn and make inferences for CRF (Qi et al., 2005). The major motivation of the model is to eliminate overfitting problem of the original CRF, and to offer the full advantages of a Bayesian setting. With the huge induced complexity, an extension of expectation propagation is developed for fast inferences. Bayesian CRF demonstrates superior performance over CRF in the computer vision domain.

3.3.2 Efficient Inference Algorithms

The CRF model enjoys several theoretical advantages compared with HMM and MEMM, and has demonstrated significant empirical improvement in many applications. However, in the training phase we need to calculate the expectation of the features for the iterative searching algorithms; and in the testing phase, we search the best assignments over all possible segmentation spaces for the structured outputs. For simple graph structure, such as a chain or a tree, the forward-backward and Viterbi styled algorithms can be used. However, the complexity increases exponentially with the induced tree-width of the graphs. As a result, exact inferences are computational infeasible for complex graphs. Therefore multiple efficient inference and learning algorithms are examined in the CRF setting, for example, the general approximate inference algorithms, such as loopy belief propagation, sampling algorithm, naive mean field and other variation methods. In addition, several specific algorithms have been developed for fast training and testing of CRF.

Gradient Tree Boosting In (Dietterich et al., 2004), a new algorithm is proposed for training CRFs by extending the gradient tree boosting method for classification (Hastie et al., 2001). Specifically, the potential functions defined in CRF are represented as weighted sums of regression trees, which are learned by stage-wise optimizations similar to Adaboost while the objective function is replaced by maximizing the conditional likelihood of joint labeling $P(\mathbf{y}|\mathbf{x})$. The algorithm successfully reduces the immense feature space via growing regression trees so that only the combinations of features defined by the trees are considered. As a result, the gradient tree boosting scales *linearly* in the order of the Markov model and the feature interactions, rather than *exponentially* as those previous algorithms based on iterative scaling and gradient descent.

Integer linear programming In (Roth & Yih, 2005), a novel inference procedure based on integer linear programming (ILP) is proposed to replace the Viterbi algorithm in the original CRF model. Specifically, the Viterbi solution can be seen as the shortest path in the graph constructed as follows: Let n be the number of tokens in the sequence, and m be the number of labels each token can take. The graph consists of $nm+2$ nodes and $(n-1)m^2+2m$

edges. In addition to two special nodes *start* and *end* that denote the start and end positions of the path, the label of each token is represented by a node v_{ij} , where $0 \leq i \leq n-1$, and $0 \leq j \leq m-1$. If the path passes node v_{ij}, then label j is assigned to token i. For nodes that represent two adjacent tokens $v_{(i-1)j}$ and $v_{ij'}$, there is a directed edge $x_{jj'}$ from $v_{(i-1)j}$ to $v_{ij'}$, with the cost $-\log(M_i(y_{i-1}y_i|x))$. Then the path is determined via minimizing $-\sum_{i=0}^{n-1} \log(M_i(y_{i-1}y_i|x))$, i.e. maximizing the function $\prod_{i=0}^{n-1} M_i(y_{i-1}y_i|x)$. The major advantage of such formulation is the convenience to add general constraints (e.g. NLP problems such as chunking, semantic role labeling, or information extraction) over the output space in a natural and systematic fashion. An efficient solution is developed to large scale applications in (Roth & Yih, 2005). The setting has the nice properties that when no additional constraints are added, the problem reduces back to one that can be solved efficiently by linear programming.

There are several other studies about the efficiency issues of CRF, such as accelerated training with stochastic gradient methods (Vishwanathan et al., 2006), numerical optimization using non-linear conjugate gradient or limited-memory variable-matric methods (Wallach, 2002). It remains a hot topic to design feasible inference and learning algorithms for CRF so that it can be applied in large-scale applications with complex graph structures.

3.3.3 Other Extensions of CRF

Conditional random fields as well as its direct extensions with different loss functions have been proven successful in multiple domains, such as natural language processing, computer vision, protein sequence analysis and so on. On the other hand, there are also many applications where the original CRF may not be the most appropriate due to their task-specific characteristics. For example in information extraction, the segment level features, such as phrase length or the segment starts with capitalized word, are very informative but they are difficult to incorporate in the CRF setting. Therefore many other extensions have been developed. Some examples include:

Semi-Markov conditional random fields Semi-Markov CRF outputs a segmentation of an input sequence x, in which labels are assigned to segments (i.e., subsequences) of x rather than to individual elements x_i (Sarawagi & Cohen, 2004). Given a sequence observation $\mathbf{x} = x_1 \ldots x_n$, the conditional probability of the segmentation given the observation is defined as

$$P(M, \{W_i\}|\mathbf{x}) = \frac{1}{Z}\exp(\sum_{i=1}^{M}\sum_{k=1}^{K}\lambda_k f(w_i, w_{i-1}, \mathbf{x})).$$

where M is the number of segments, $W_i = \{p_i, q_i, s_i\}$ and p_i, q_i, s_i are the starting position, ending position and state of the i^{th} segment. The biggest advantage of this revision allows features that measure properties of segments, and non-Markovian transitions within a segment. In spite of this additional power, the complexity of exact learning and inference algorithms for semi-CRFs are polynomial, often only a small constant factor slower than conventional CRFs. The model has shown significant improvement over the original CRF in the information extraction tasks.

Hidden conditional random fields The hidden CRF is another extension of the CRF, which introduces hidden variables between the labels and observations for the recognition of object classes and gestures (Quattoni et al., 2005; Wang et al., 2006). For each object class, the probability of a given assignment of parts to local features is modeled by a CRF. Then the parameters of the CRF are estimated in a maximum likelihood framework and recognition proceeds by finding the most likely class under the model. The main advantage of hidden CRF is the relaxation of the conditional independence assumptions of the observed data (i.e. local features), which are often used in generative approaches.

Other complex graph structures Up to now, the CRF model and its variations are mostly used in applications with simple graph structures, such as a chain, a tree or grids. It is not hard to imagine that many real applications might require quite complex graph structures, such as protein three-dimensional structures, or multiple layers of chains, involving both time and location scales. Therefore several models are developed along this direction, such as layout consistent random field (Winn & Shotton, 2006), dynamic CRF (Sutton et al., 2004) and so on.

3.4 Applications

The elegant combination of graphical models and discriminative settings enables CRF and its extensions widely applied in multiple domains, such as natural language processing, computer vision, speech recognition and computational biology. Below is an incomplete list of the exciting applications of CRF:

Natural language processing In the NLP area, CRF has been applied to shallow parsing (Sha & Pereira, 2003), word alignment (Tsochantaridis et al., 2004) and table extraction (Pinto et al., 2003). Some other examples include contrastive estimation, i.e. an unsupervised version of CRF, for part-of-speech (POS) tagging and grammar induction (Smith & Eisner, 2005), dynamic CRF for joint labeling of POS tagging and noun phrase extraction (Sutton et al., 2004), semi-Markov CRF for information extraction (Sarawagi & Cohen, 2004), 2-D CRF for web information extraction (Zhu et al., 2005) and CRF for co-reference resolution (Sutton & McCallum, 2006).

Computer vision In computer vision area, CRF was initially used for image segmentation (Kumar & Hebert, 2003); later a dynamic conditional random field model is proposed to capture the spatial and temporal dependencies for image sequences (Wang et al., 2006), Sminchisescu et al. applied CRFs to classify human motion activities (i.e. walking, jumping, etc) (Sminchisescu et al., 2005), Torralba et al. introduced boosted random fields, a model that combines local and global image information for contextual object recognition (Torralba et al., 2004), Quattoni developed the Hidden CRFs to model spatial dependencies for object recognition in unsegmented cluttered images (Quattoni et al., 2005), He et all propose the multi-scale CRF for modeling patterns of different scales (He et al., 2004).

Computational biology In computational biology, the CRF model was first used for protein secondary structure prediction (Liu et al., 2004). Later it has been applied to detecting overlapping elements in sequence data (Bockhorst & Craven, 2005), disulfide bond prediction (Taskar & Simon Lacoste-Julien, 2005), RNA secondary structural alignment (Do et al., 2006b), protein sequence alignment (Do et al., 2006a) and gene prediction [unpublised manuscript].

3.5 Summary and Other Sources

The prediction problem with structured outputs is one of the emerging trends in the fields of machine learning. It is closely related to the multi-task learning and relational learning, although they are originated from different motivations and used for distinct applications. The CRF-like model has played a central role in the solutions to predict structured outputs. Nowadays, many topics on the classification problem, such as unbalanced data and semi-supervised learning, have emerged in the study for structured prediction. However, the search for efficient inference and learning algorithms remain essential for wide applications of CRF.

In addition to the discussion above, there are also several useful information sources and software available on this topics. Below is an incomplete list:

Available information sources

- Website devoted for CRF:

  http://www.inference.phy.cam.ac.uk/hmw26/crf/

- Some tutorials include:
  Hanna M. Wallach. Conditional Random Fields: An Introduction. Technical Report MS-CIS-04-21. Department of Computer and Information Science, University of Pennsylvania.

  Charles Sutton and Andrew McCallum. An Introduction to Conditional Random Fields for Relational Learning. In *Introduction to Statistical Relational Learning.* http://www.cs.umass.edu/ casutton/publications/crf-tutorial.pdf

  Ben Taskar. Large-Margin Learning of Structured Prediction Models. UAI-2005 Tutorial.

  Trevor Cohn. Tutorial on Conditional Random Fields. In ALTA Workshop. http://homepages.inf.ed.ac.uk/tcohn/talks/crf_tutorial.pdf

Available software

- mallet-CRF - http://crf.sourceforge.net/

  Java implementation. An efficient implementation of CRFs which extensively relies on sparse matrix operations and Quasi-Newton optimization during training (including CRF and semi-Markov CRF).

- flexCRF - http://www.jaist.ac.jp/ hieuxuan/flexcrfs/flexcrfs.html

C/C++ implementation. It provides CRF with both first-order and second-order Markov assumptions.

- CRF - http://www.cs.ubc.ca/ murphyk/Software/CRF/crf.html

  Matlab implementation. The graph structure can be 1D chain, 2D lattice and general graph. It is also embedded with stochastic meta-descent for fast training.

- CRF ++ - http://chasen.org/ taku/software/CRF++/

  C++ implementation. The software can be applied to a variety of NLP tasks, such as named entity recognition, information extraction and text chunking.

- SVM$^{\text{struct}}$ - http://svmlight.joachims.org/

  C++ C++ version. SVMstruct is a Support Vector Machine (SVM) algorithm for predicting multivariate outputs.

- A free download code in Matlab will be available soon in MatlabArsernal

  http://finalfantasyxi.inf.cs.cmu.edu/MATLABArsenal/MATLABArsenal.htm.

Chapter 4

Conditional Graphical Models

Structural bioinformatics, as a subfield in computational biology, involves different aspects of protein structures, including the structural representation, structural alignment and comparison, structure and function assignments, and new structure design as drug targets. In this thesis, we focus on predicting the general protein structural topologies (as opposed to specific 3-D coordinates) of different levels, including secondary structures, super-secondary structures and quaternary folds for homogeneous multimers. Given these putative structural topologies of a protein, the backbone of the tertiary (or quaternary) structures is known and more importantly it can serve as a key indicator for certain functional or binding sites.

In contrast to the traditional ii assumptions in statistics and machine learning, one distinctive property of protein structures is that the residues at different positions are not independent. For example, neighboring residues in the sequence are connected by peptide bonds; some residues that are far away from each other in the primary structures might be close in 3-D and form chemical bonds, such as hydrogen bonds or disulfide bonds. These chemical bonds are essential to the stability of the structures and directly determine the functionality of the protein. In order to model the long-range interactions explicitly and incorporate all our tasks into a unified framework, it is desirable to have a powerful model that can capture the *interdependent structured* properties of proteins. Recent work on conditional graphical models shows that they are very effective in the prediction problem for structured data, such as information extraction, parsing, image classification and etc (Kumar & Hebert, 2003; Pinto et al., 2003; Sha & Pereira,

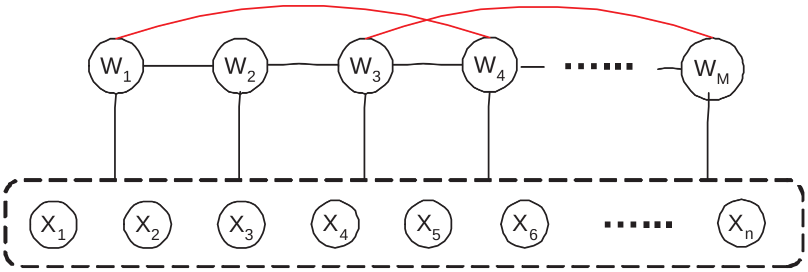


Figure 4.1: The graphical model representation of conditional graphical models. Circles represent the state variables, edges represent couplings between the corresponding variables (in particular, long-range interaction between units are depicted by red arcs). The dashed box over x's denote the sets of observed sequence variables. An edge from a box to a node is a simplification of dependencies between the non-boxed node to all the nodes in the box (and therefore result in a clique containing all x's).

2003). In addition, the graph representation in the model are intuitively similar to the protein structures, which simplifies the process to incorporate domain knowledge and also helps the biologists better understand the protein folding pathways.

4.1 Graphical Model Framework for Protein Structure Prediction

In this thesis, we develop a series of graphical models for protein structure prediction. These models can be generalized to the framework of conditional graphical models, which directly defines the probability distribution over the labels (i.e., segmentation and labeling of the delineated segments) underlying an observed protein sequence, rather than assuming particular data generating process as in the generative models. Specifically, our model can be represented via an undirected graph $G = \{\mathcal{V}, \mathcal{E}\}$, which we refer to as "protein structural graph" (PSG). $\mathcal{V}$ is the set of nodes corresponding to the specificities of structural units such as secondary structure assignments, motifs or insertions in the supersecondary structure (which are unobserved and to be inferred), and the amino acid residues at each position (which are observed and to be conditioned on). $\mathcal{E}$ is the set of edges denoting dependencies between the objects represented by the nodes, such as local constraints and/or state transitions between adjacent nodes in the primary sequence, or long-range interactions between non-neighboring motifs and/or insertions (see Fig. 4.1). The latter type of dependencies is unique to the protein structural graph and results in much of the difficulties in solving such graphical models.

The random variables corresponding to the nodes in PSG are as follows: M denotes the number of nodes in PSG. Notice that M can be either a constant or a variable taking values from a discrete sets $\{1, \ldots, m_{\max}\}$, where $m_{\max}$ is the maximal number of nodes allowed (usually defined by the biologists). W_i is the label for the i^{th} node, i.e. the starting and ending positions in the sequence and/or state assignment, which completely determine the node according to its semantics defined in the PSG. Under this setup, a value instantiation of $W = \{M, \{W_i\}\}$ defines a unique segmentation and annotation of the observed protein sequence $\mathbf{x}$ (see Fig. 4.1).

Let $\mathcal{C}_G$ denote the set of cliques in graph G. Furthermore, we use W_c to represent an arbitrary clique $c \in \mathcal{C}_G$. Given a protein sequence $\mathbf{x} = x_1 x_2 \ldots x_N$ where $x \in \{\text{amino acid}\}$, and a PSG G, the probabilistic distribution of the labels W given observation $\mathbf{x}$ can be postulated using the potential functions defined on the cliques in the graph (Hammersley & Clifford, 1971), i.e.

$$P(W|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \Phi(\mathbf{x}, W_c), \tag{4.1}$$

where Z is a normalization factor and $\Phi(\cdot)$ is the potential function defined over a clique. Following the idea of CRFs, the clique potential can be defined as an exponential function of the feature function f, i.e.

$$P(W|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, W_c)), \tag{4.2}$$

where K is the number of features. The definition of the feature function f varies, depending on the semantics of nodes in the protein structure graph (see the next section for details). The parameters $\lambda = (\lambda_1, \ldots, \lambda_K)$ are computed by minimizing the regularized log-loss of the conditional probability of the training data, i.e.

$$\lambda = \operatorname{argmax} \{\sum_{j=1}^{L} \log P(\mathbf{w}^{(j)}|\mathbf{x}^{(j)}) + \Omega(\|\lambda\|)\}, \tag{4.3}$$

where L is the number training sequences. Notice that the conditional likelihood function is convex so that finding the global optimum is guaranteed.

Given a query protein, our goal is to seek the segmentation configuration with the highest conditional probability defined above, i.e.

$$w^{\text{opt}} = \arg\max_{W} \ P(W|\mathbf{x}).$$

The major advantages of the conditional graphical model defined above include: (1) the intuitive representation of protein structures via graphs; (2) the ability to model dependencies between segments in a non-Markovian way, so that the chemical-bonding between distant residues (both inter-chain and intra-chain bonding) can be captured; (3) the ability to use any features that measure properties of segments or bonds that biologists have identified.

4.2 Protein Structure Graph Construction

In the previous section, we give the definition of the protein structure graph (PSG), which is an annotated graph $G = \{V, E\}$, with V as the set of nodes corresponding to the specificities of structural units and E as the set of edges denoting dependencies between the objects represented by the nodes, such as location constraints or long-range interactions between non-neighboring units. Our next question is how to construct the PSG for a target structure. This usually requires basic understanding about protein structures as well as the input from domain experts. More specifically, we need to address the following questions: what are the measures of a good PSG? how to construct a PSG for protein structures using prior knowledge? how to automatic generate a PSG for any types of protein structures without any prior knowledge?

As we can see, the definition of PSG is descriptive rather than instructive. Given one protein structure of concern, we can usually construct several reasonable PSG with different semantics for each node. Therefore there is a tradeoff between the graph complexity, fidelity of model and the real computational costs. The measures we take to evaluate a PSG is the expressiveness, i.e. we search for the graphs that capture most important properties of the protein structures while retaining as much simplicity as possible. In other words, the optimal PSG is the one yielding the highest scores defined as the likelihood of the training data minus its graph complexity.

In many cases, our target structures have been studied by the biologists over the years and some basic knowledge of their properties have been accumulated. Most of the prediction problems addressed in the thesis belongs to this category. The PSG of such structures can be constructed easily by communicating with the experts. The information we need to collect is:

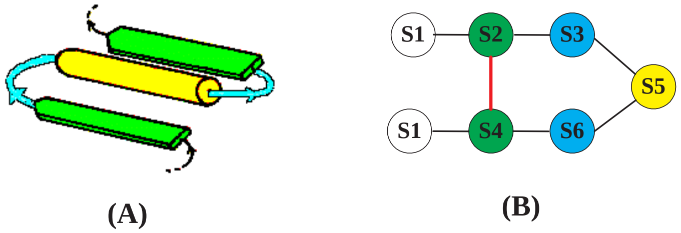


Figure 4.2: Graph structure of β-α-β motif (A) 3-D structure (B) Protein structure graph: node: Green=β-strand, yellow=α-helix, cyan=coil, white=non-β-α-β (I-node); edge: $E_1 = \{\text{black edges}\}$ and $E_2 = \{\text{red edges}\}$.

what are the structural components? how do they associate with each other via chemical bonds? which chemical bonds are unique or important for the stability of the structures? For example, for the β-α-β motif, we know that it consists of two β-strands with an α-helix in-between; the hydrogen bonds connecting the two β-strands uniquely identify the motif. Therefore we can construct a PSG as shown in Figure 4.2.

In some cases, we need to handle the protein structures that are quite new to the biologists and no prior knowledge of their structure properties are given. To solve the problem, we need to learn a PSG automatically from the data. This problem falls in the general task of structure learning in graphical model research. Compared with previous work in structure learning, the key challenges are the availability of training data since many novel structures have only 1 or 2 positive proteins for training. On the other hand, we are also provided additional information (i.e. the three-dimensional structures of positive proteins) which can guide the learning. In general, we can follow the systematic procedures below to construct an initial graph:

1. Build a multiple structure alignment of all the positive proteins (among themselves)

2. Segment the alignment into disjoint parts based on the secondary structures of the majority proteins

3. Draw a graph with nodes denoting the resulting secondary structure elements and then add edges between neighboring nodes to model local constraints

4. Add the long-range interaction edge between two nodes if the average distance between all the involved residues is below some threshold κ^{min} specified by the user.

After we get the initial graphs, the next step is to search for the optimal PSG by performing only two types of actions, merging nodes and deleting edges. We skip detailed discussion of the latter case as it is a separate line of research and assume that we are given a reasonably good graph over which we perform our learning.

4.3 Specific Contribution

To address the prediction problem on different protein structure hierarchies, several conditional graphical models are developed as a special case of the model defined in eq(4.2) . Table 4.1 summarizes the models, which are described in detail below.

4.3.1 Conditional Random Fields (CRFs)

Protein secondary structure prediction assigns the secondary structure label, such as helix, sheet or coil, for each residue in the protein sequence. Therefore the nodes in the PSG represent the states of secondary structure assignment and the graph structure is simply a chain as the protein sequence. As we can see, the model is the plain CRF with a chain structure (Lafferty et al., 2001). Its graphical model representation for a chain-structured CRF is shown in Figure 4.3, in which we have one node for the state assignment for each residue in the sequence. Mapping back to the general framework in the previous section, we have $M = n$ and $W_i = y_i \in \{\text{helix, sheets, coils}\}$. The conditional probability $P(W|\mathbf{x}) = P(\mathbf{y}|\mathbf{x})$ is defined as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z}\prod_{i=1}^{N}\exp(\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, i, y_{i-1}, y_i)), \tag{4.4}$$

where f_k can be arbitrary features, including overlapping or long-range interaction features. As a special case, we can construct HMM-like features that are factored as two parts: $f_k(\mathbf{x}, i, y_{i-1}, y_i) = g_k(\mathbf{x}, i)\delta(y_{i-1}, y_i)$, in which $\delta(y_{i-1}, y_i)$ is the indicator function over each pair of state assignments (y_{i-1}, y_i) (similar to the transition probability in HMM), and $g_k(\mathbf{x}, i)$ is any feature defined over the observations $(\mathbf{x}, y_i)$ (which mimics the emission probability without any particular assumptions about the data).

CRFs take on a global normalizer Z over the whole sequence. This results in a series of nice properties, but at the same time introduces huge computational costs. Maximum Entropy Markov Models (MEMMs) can be

Table 4.1: Thesis work: conditional graphical models for protein structure prediction of all hierarchies

Hierarchy	Secondary		Tertiary		Quaternary	
Task	secondary structure prediction	parallel/ antiparallel β-sheet prediction	Fold (motif) recognition	Structural repeats	Quaternary fold recognition (w/o sequence repeats)	Quaternary fold recognition (with sequence repeats)
Target Proteins	globular proteins	globular proteins	β-helix	β-helix, leucine-rich repeats	double barrel trimer	triple β-spiral
Structural modules	amino acid	amino acid	secondary structure	structural motifs/ insertions	secondary/super- secondary structures	
Module length	fixed	fixed	variable	variable	variable	
Graphical model	CRFs, kCRFs	CRFs	SCRFs	chain graph model	linked SCRFs	

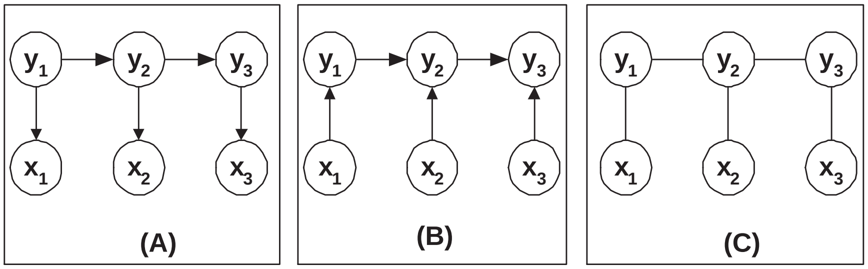


Figure 4.3: Graphical model representation of simple HMM(A), MEMM(B), and chain-structured CRF(C)

seen as a localized version of CRFs (see Fig. 4.3 (B)) (McCallum et al., 2000). The conditional probability in MEMMs is defined as

$$P(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{N} \frac{1}{Z_i} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i, y_{i-1}, y_i)), \tag{4.5}$$

where Z_i is a normalizer over the i^{th} position. MEMMs reduce the computational costs dramatically, but suffer from the "label bias" problem, that is, the total probability "received" by y_{i-1} must be passed on to labels y_i at time i even if x_i is completely incompatible with y_{i-1} (Lafferty et al., 2001). Empirical results show that for most applications CRFs are able to outperform MEMMs with either slight or significant improvement. The detailed comparison results with applications to protein secondary structure prediction are discussed in Section 6.3.

4.3.2 Kernel Conditional Random Fields (kCRFs)

The original CRFs model only allows linear combination of features. For protein secondary structure prediction, the state-of-art method can achieve an accuracy of around 80% using SVM with linear kernels, which indicates that the current feature sets are not sufficient for a linear separation.

Recent work in machine learning has shown that kernel methods are extremely effective in a wide variety of applications (Cristianini & Shawe-Taylor, 2000). Kernel conditional random fields, as an extension of conditional random fields, permits the use of implicit features spaces through Mercer kernels (Lafferty et al., 2004). Similar to CRFs, the conditional

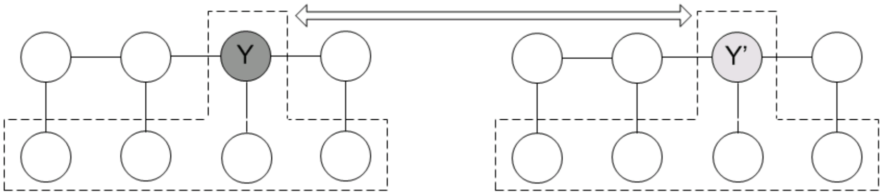


Figure 4.4: Kernels for structured graph: $K((\mathbf{x}, c, y_c), (\mathbf{x}', c', y'_c)) = K((\mathbf{x}, c), (\mathbf{x}', c'))\delta(y_c, y'_c)$.

probability is defined as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \exp f^*(\mathbf{x}, c, y_c),$$

where $f(\cdot)$ is the kernel basis function, i.e. $f(\cdot) = K(\cdot, (\mathbf{x}, c, y_c))$. One way to define the kernels over the structured graph can be $K((\mathbf{x}, c, y_c), (\mathbf{x}', c', y'_c)) = K((\mathbf{x}, c), (\mathbf{x}', c'))\delta(y_c, y'_c)$, whose first term is the typical kernels defined for ii examples, and the second term is the indicator function over each state pair $\delta(y_c, y'_c)$ (see Fig. 4.4). By the representer theorem, the minimizer of the regularized loss has the form

$$f^*(\cdot) = \sum_{j=1}^{L} \sum_{c \in \mathcal{C}_{G^{(j)}}} \sum_{y_c \in \mathcal{Y}^{|c|}} \lambda_{y_c}^{(j)} K(\cdot, (\mathbf{x}^{(j)}, c, y_c)).$$

Notice that the dual parameters λ depend on all the clique label assignments, not limited to the true labels. The detailed algorithms and experiment results of predicting protein secondary structures are shown in Section 6.5.

4.3.3 Segmentation Conditional Random Fields (SCRFs)

Protein folds or motifs are frequent arrangement patterns of several secondary structure elements. Therefore the layout patterns are usually described in secondary structure elements instead of individual residue. Since the topologies information are known in advance, it would be natural to build an undirected graph, with each node representing the secondary structural elements and the edges indicating the interactions between the elements in three-dimensional structures. Then, given a protein sequence, we can search

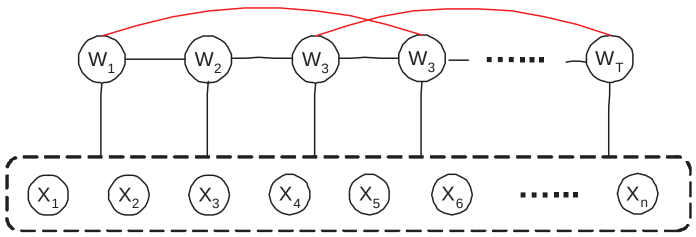


Figure 4.5: Graphical model representation for segmentation conditional random fields

for the best segmentation defined by the graph. Following the idea, a segmentation conditional random fields (SCRFs) model can be developed for general protein fold recognition (Liu et al., 2005).

For protein fold (or motif) recognition, we define a PSG $G =< \mathcal{V}, \mathcal{E} >$, where $\mathcal{V} = \mathcal{U} \bigcup \{\mathrm{I}\}$, $\mathcal{U}$ is the set of nodes corresponding to the secondary structure elements within the fold and I is the node that represents the elements outside the fold. $\mathcal{E}$ is the set of edges between neighboring elements in primary sequences or those indicating the potential long-range interactions between elements in tertiary structures (see Figure 4.5). Given the graph G and a protein sequence $\mathbf{x} = x_1 x_2 \ldots x_N$, we can have a possible segmentation of the sequence, i.e. $W = \{M, \{W_i\}\}$, where M is the number of segments, $W_i = \{s_i, p_i, q_i\}$, and s_i, p_i, q_i are the state, starting position and ending position of the i^{th} segment. Here the states are the set of labels to distinguish each structural component of the fold. The conditional probability of W given the observation $\mathbf{x}$ is defined as

$$P(W|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, w_c)), \tag{4.6}$$

where f_k is the k^{th} feature defined over the cliques c. In a special case, we can consider only the pairwise cliques, i.e.

$$f(\mathbf{x}, w_i, w_j) = g(\mathbf{x}, p_i, q_i, p_j, q_j)\delta(s_i, s_j)\delta(q_i - p_i)\delta(q_j - p_j),$$

where g is any feature defined over the two segments. Note that $\mathcal{C}_G$ can be a huge set, and each W_c can also include a large number of nodes due to various levels of dependencies. Designing features for such cliques is nontrivial because one has to consider all the joint configurations of all the nodes in a clique.

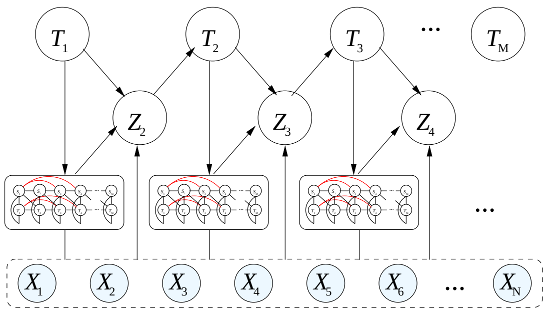


Figure 4.6: Chain graph model for predicting folds with repetitive structures

Usually, the spatial ordering of most regular protein folds is known *a priori*, which leads to a deterministic state dependency between adjacent nodes w_i and w_{i+1}. Thus we have a simplification of the "effective" clique sets (those need to be parameterized) and the relevant feature design. Essentially, only pairs of segment-specific cliques that are coupled need to be considered (e.g., those connected by the undirected "red" arc in Figure 4.5)[1], which results in the following formulation:

$$P(W|\mathbf{x}) = \frac{1}{Z}\prod_{i=1}^{M}\exp(\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, W_i, W_{\pi_i})), \tag{4.7}$$

where W_{π_i} denotes the spatial predecessor (i.e., with small position index) of W_i determined by a "long-range interaction arc". The detailed inference algorithm with application to regular fold recognition is described in Section 7.2.

4.3.4 Chain Graph Model

SCRF is a model for regular protein fold recognition. It can be seen as an exhaustive search over all possible segmentation configurations of the given protein and thus results in tremendous computational costs. To alleviate the problem, a chain graph model is proposed, which is designed for a special structure, i.e. protein folds with repetitive structural repeats. They are defined as repetitive structurally conserved secondary or supersecondary units, such as α-helices, β-strands, β-sheets, connected by *insertions* with variable number of residues, which are mostly short loops and sometimes α-helices

[1] Technically, neighboring nodes must satisfy the constraints on the location indexes, i.e. $q_{i-1} + 1 = p_i$. We omit it here for presentation clarity.

or/and β-sheets. These folds are believed to be prevalent in proteins and involve in a wide spectrum of cellular activities.

A *chain graph* is a graph consisting of both directed and undirected arcs associated with probabilistic semantics. It leads to a probabilistic distribution bearing the properties of both Markov random fields (i.e., allowing potential-based local marginals that encode constraints rather than causal dependencies) and Bayesian networks (i.e., not having a hard-to-compute global partition function for normalization and allowing causal integration of subgraphs that can be either directed or undirected) (Lauritzen & Wermuth, 1989; Buntine, 1995).

Back to the protein structure graph, we propose a *hierarchical segmentation* for a protein sequence. On the top level, we define an *envelop* Ξ_i, as a subgraph that corresponds to one repeat region in the fold (containing both motifs and insertions or the null regions, i.e. structures outside the protein fold). It can be viewed as a mega node in a chain graph defined on the entire protein sequence and its segmentation (Fig. 4.6). Analogous to the SCRF model, let M denote the number of envelops in the sequence, $\mathbf{T} = \{T_1, \ldots, T_M\}$ where $T_i \in \{\text{repeat, non-repeat}\}$ denote the structural label of the i^{th} envelop. On the lower level, we decompose each envelop as a regular arrangement of several motifs and insertions, which can be modeled using one SCRFs model. Let Ξ_i denote the internal segmentation of the i^{th} envelop (determined by the local SCRF), i.e. $\Xi_i = \{M_i, \mathbf{Y}_i\}$. Following the notational convention in the previous section, we use $W_{i,j}$ to represent a segment-specific clique within envelop i that completely determines the configuration of the j^{th} segment in the i^{th} envelop. To capture the influence of neighboring repeats, we also introduce a motif indicator Q_i for each repeat i, which signals the presence or absence of sequence motifs therein, based on the sequence distribution profiles estimated from previous repeat.

Putting everything together, we arrive at a chain graph depicted in Fig. 4.6. The conditional probability of a segmentation W given a sequence $\mathbf{x}$ can be defined as

$$P(W|\mathbf{x}) = P(M, \boldsymbol{\Xi}, \mathbf{T}|\mathbf{x}) = P(M) \prod_{i=1}^{M} P(T_i|\mathbf{x}, T_{i-1}, \Xi_{i-1}) P(\Xi_i|\mathbf{x}, T_i, T_{i-1}, \Xi_{i-1}) \quad (4.8)$$

$P(M)$ is the prior distribution of the number of repeats in one protein, $P(T_i|\mathbf{x}, T_{i-1}, \Xi_{i-1})$ is the state transition probability and we use the structural motif as an indicator for the existence of a new repeat:

$$P(T_i|\mathbf{x}, T_{i-1}, \Xi_{i-1}) = \sum_{Q_i=0}^{1} P(T_i|Q_i) P(Q_i|\mathbf{x}, T_{i-1}, \Xi_{i-1}),$$

where Q_i is a random variable denoting whether there exists a motif in the i^{th} envelop and $P(Q_i|\mathbf{x}, T_{i-1}, \Xi_{i-1})$ can be computed using any motif detection model. For the third term, a SCRFs model is employed, i.e.

$$P(\Xi_i|\mathbf{x}, T_i, T_{i-1}, \Xi_{i-1}) = \frac{1}{Z_i}\exp(\sum_{j=1}^{M_i}\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, W_{i,j}, W_{\pi_{i,j}})), \qquad (4.9)$$

where Z_i is the normalizer over all the configurations of Ξ_i, and $W_{\pi_{i,j}}$ is the spatial predecessor of $W_{i,j}$ defined by long-range interactions. Similarly, the parameters λ_k can be estimated by minimizing the regularized negative log-loss of the training data.

Compared with SCRFs, the chain graph model can effectively identify motifs by exploring their structural conservation and at the same time take into account the long-range interactions between repeat units. In addition, the model takes on a local normalization, which reduces the computational costs dramatically. Since the effects of most chemical bonds are limited to a small range in 3-D space without passing through the whole sequence, this model can be seen as a reasonable approximation for a global optimal solution as SCRFs. The details of the algorithm and experiment results are discussed in Section 7.3.

4.3.5 linked Segmentation Conditional Random Fields (l-SCRFs)

The *quaternary structure* is the stable association of multiple polypeptide chains via non-covalent bonds, resulting in a stable unit. Quaternary structures are stabilized mainly by the same non-covalent interactions as tertiary structures, such as hydrogen bonding, van der Walls interactions and ionic bonding. Unfortunately, previous work on fold recognition for single chains is not directly applicable because the complexity is greatly increased both biologically and computationally, when moving to quaternary multi-chain structures. Therefore we propose the linked SCRF model to handle protein folds consisting of multiple protein chains.

The PSG for a quaternary fold can be derived similarly as the PSG for tertiary fold: first construct a PSG for each component protein or a component monomeric PSG for homo-multimer, and then add edges between the nodes from different chains if there are chemical bonds, forming a more complex but similarly-structured quaternary PSG. Given a quaternary structure graph G with C chains, i.e. $\{\mathbf{x}_i|i = 1 \ldots C\}$, we have a segmentation initiation of each chain $\mathbf{y}_i = (M_i, \mathbf{w}_i)$ defined by the PSG, where M_i is the number of segments in the i^{th} chain, and $\mathbf{w}_{i,j} = (s_{i,j}, p_{i,j}, q_{i,j})$, $s_{i,j}$, $p_{i,j}$, $q_{i,j}$ are the state, starting position and ending position of the j^{th} segment. Following similar idea as the CRFs model, we have

$$P(\mathbf{y}_1, \ldots, \mathbf{y}_C | \mathbf{x}_1, \ldots, \mathbf{x}_C) = \frac{1}{Z} \prod_{\mathcal{C} \in G} \Phi(\mathbf{y}_{\mathcal{C}}, \mathbf{x}) \tag{4.10}$$

$$= \frac{1}{Z} \prod_{\mathbf{w}_{i,j} \in \mathcal{V}_G} \Phi(\mathbf{x}_i, \mathbf{w}_{i,j}) \prod_{\langle \mathbf{w}_{a,u}, \mathbf{w}_{b,v} \rangle \in \mathcal{E}_G} \Phi(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}_{a,u}, \mathbf{w}_{b,v}) \tag{4.11}$$

$$= \frac{1}{Z} \exp(\sum_{\mathbf{w}_{i,j} \in \mathcal{V}_G} \sum_{k=1}^{K1} \theta_{1,k} f_k(\mathbf{x}_i, \mathbf{w}_{i,j}) + \sum_{\langle \mathbf{w}_{a,u}, \mathbf{w}_{b,v} \rangle \in \mathcal{E}_G} \sum_{k=1}^{K2} \theta_{2,k} g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}_{a,u}, \mathbf{w}_{b,v})) \tag{4.12}$$

where Z is the normalizer over all possible segmentation assignments of *all* component sequences (see Figure 4.7 for its graphical model representation). In eq(4.12), we decompose the potential function over the cliques $\Phi(\mathbf{y}_{\mathcal{C}}, \mathbf{x})$ as a product of unary and pairwise potentials, where f_k and g_k are features, $\theta_{1,k}$ and $\theta_{2,k}$ are the corresponding weights for the features. Specifically, we factorize the features as the following way,

$$f_k(\mathbf{x}_i, \mathbf{w}_{i,j}) = f'_k(\mathbf{x}_i, p_{i,j}, q_{i,j}) \delta(\mathbf{w}_{i,j})$$
$$= \begin{cases} f'_k(\mathbf{x}_i, p_{i,j}, q_{i,j}) & \text{if } s_{i,j} = s \& q_{i,j} - p_{i,j} \in \text{length range}(s) \\ 0 & \text{otherwise,} \end{cases}$$

Similarly, we can factorize $g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}_{a,u}, \mathbf{w}_{b,v}) = g'_k(\mathbf{x}_a, \mathbf{x}_b, q_{a,u}, p_{a,u}, q_{b,v}, p_{b,v})$ if $q_{a,u} - p_{a,u} \in$ length range (s) and $q_{b,v} - p_{b,v} \in$ length range (s'), and 0 otherwise.

The major advantages of linked SCRFs model include: (1) the ability to encode the output structures (both inter-chain and intra-chain chemical bonding) using the graph; (2) dependencies between segments can be non-Markovian so that the chemical-bonding between distant amino acids can be captured; (3) it permits the convenient use of any features that measure the property of segments the biologists have identified. On the other hand, the linked SCRF model differs significantly from the SCRF model in that the quaternary folds with multiple chains introduce huge complexities for inference and learning. Therefore we develop efficient approximation algorithms that are able to find optimal or near-optimal solutions as well as their applications in Chapter 8.

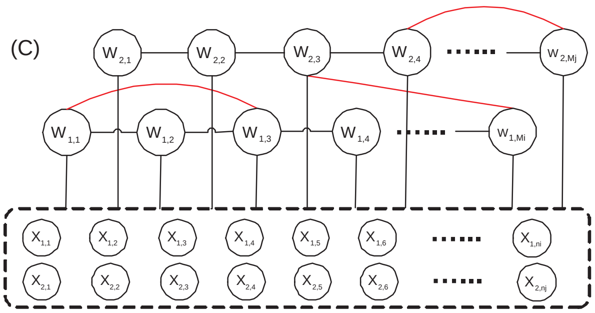


Figure 4.7: Graphical Model Representation of l-SCRFs model with multiple chains. Notice that there are long-range interactions (represented by red edges) within a chain and between chains

4.4 Discussion

In our previous discussion, we use the regularized log loss as the objective function to estimate parameters, following the original definition in CRFs model (Lafferty et al., 2001). In addition to CRFs, there are several other discriminative methods proposed for the segmentation and labeling problem of structured data, such as max-margin Markov networks (M^3N) (Taskar et al., 2003) and Gaussian process sequence classifier (GPSC) (Altun et al., 2004) (see Chapter 3 for full discussion). Similar to the classifiers for classification problem, these models can be unified under the exponential model with different loss functions and regularization terms.

4.4.1 Unified View via Loss Function Analysis

Classification problem, as a subfield in supervised learning, aims at assigning one or more *discrete* class labels to each example in the dataset. In recent years, various classifiers have been proposed and successfully applied in lots of applications, such as logistic regression, support vector machines, naive Bayes, k-Nearest neighbor and so on (Hastie et al., 2001). Discriminative classifiers, as opposed to generative models, computes the conditional probability directly and usually assumes a linear decision boundary in the original feature space or in the corresponding Hilbert space defined by the kernel functions. Previous research work indicate that the loss function analysis can provide a comprehensible and unified view of those classifiers with totally different mathematical formulation (Hastie et al., 2001).

In the following discussion, we focus on the binary classification problem and concern ourselves with three specific classifiers, including regularized logistic regression (with extension to kernel logistic regression) (Zhu & Hastie, 2001), support vector machines (SVM) (Vapnik, 1995) and Gaussian process (GP) (Williams & Barber, 1998). All these three classifiers can be seen as a linear classifier which permits the use of kernels. Specifically, the decision function $f(x)$ has the form as

$$f(x) = \sum_{i=1}^{L} \lambda_i K(x_i, x), \tag{4.13}$$

where λ are the parameters of the model and K is the kernel function. λ are learned by minimizing a regularized loss function and the general form of the optimization function can be written as

$$\lambda = \text{argmax} \sum_{i=1}^{L} g(y_i f(x_i)) + \Omega(\|f\|_{\mathcal{F}}), \tag{4.14}$$

where the first term is the training set error, g is specific loss function and the second term is the complexity penalty or regularizer.

The essence of different classifiers can be revealed through their definitions of the loss functions as follows:

- Kernel logistic regression defines the loss function as the logistic loss, i.e. $g(z) = \log(1+\exp(-z))$. In the model described in (Zhu & Hastie, 2001), a Gaussian prior with zero mean and diagonal covariance matrix is applied, which equals to an L_2 regularizer.

- Support vector machines uses the hinge loss, i.e. $g(z) = (1 - z)_+$, which results in the nice properties of sparse parameters (most values are equal to 0). Similar to logistic regression, an L_2 regularizer is employed.

- Gaussian process classification can be seen as a logistic loss with Gaussian prior defined over infinite dimensions over f. Since it is intractable to integrate out all the hidden variables, maximum a posterior (MAP) estimate has to be applied. This formulation has a very similar loss function expression as the kernel logistic regression except it is more general in terms of the definition for mean and variance in the Gaussian prior.

Previous analysis on loss functions provides a general view for different classifiers and helps us better understand the classification problem. For the prediction problem (segmentation and labeling) of structured data, a similar analysis can be derived accordingly. As discussed in Section 4.1, conditional graphical models define the probability of the label sequence $\mathbf{y}$ given the observation $\mathbf{x}$ directly and use exponential model to estimate the potential functions. The decision function $f(x)$ has the form as:

$$f^*(\cdot) = \sum_{j=1}^{L} \sum_{c \in \mathcal{C}_{G(j)}} \sum_{y_c \in \mathcal{Y}^{|c|}} \lambda_{y_c}^{(j)} K(\cdot, (\mathbf{x}^{(j)}, c, y_c)).$$

where λ is the model parameters which can be learned by minimizing the loss over the training data. Similar to kernel logistic regression, kernel CRFs take on a logistic loss with an L_2 regularizer. Max-margin Markov networks, like SVM, employs a hinge loss. On the other hand, the Gaussian process classification for segmenting and labeling (GPSC) are motivated from the gaussian process point of view, however, its final form are very close to kCRFs.

In summary, although our work is mostly focused on the logistic loss, they can easily be adapted to other loss functions and regularizer, depending on the tradeoff between complexity and effectiveness in specific applications.

4.4.2 Related Work

From machine learning perspective, our conditional graphical model framework is closely related to the semi-Markov conditional random fields (Sarawagi & Cohen, 2004) and dynamic conditional random fields (Sutton et al., 2004) (see Chapter for detail). All these three models are extensions of the CRF model, however, ours is more representative in that it allows both the semi-Markov assumptions, i.e. assigning the label to a segment (i.e. subsequence) instead of individual element, and graph structures involving multiple chains. Furthermore, our models are able to handle the interactions or associations between nodes even on different chains thanks to the flexible formulation and efficient inference algorithms we developed.

In structural biology, the conventional representation of a protein fold is the use of a graph (Westhead et al., 1999), in which nodes represent the secondary structure components and the edges indicate the inter- and intra-chain interactions between the components in the 3-D structures. Therefore the graph representation for protein structures is not novel from that perspective. However, there have been very few studies about combining the graph representation and probability theory via graphical models for protein structure prediction. Furthermore, there has been no work about developing discriminative training of graphical models on this topics.

Chapter 5

Efficient Inference Algorithms

In the previous chapter, we describe the general framework of conditional graphical models. Given an observation sequence $\mathbf{x} = x_1 x_2 \ldots x_N$, the *conditional* probability of a possible segmentation $W = \{M, \{W_i\}\}$ according to a protein structure graph G, is defined as

$$P(W|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, W_c)), \tag{5.1}$$

The parameters $\lambda = (\lambda_1, \ldots, \lambda_K)$ can be computed by minimizing the regularized log-loss of the training data, i.e.

$$\lambda = \text{argmax} \, \{\sum_{j=1}^{L} \log P(\mathbf{w}^{(j)}|\mathbf{x}^{(j)}) + \Omega(\|\lambda\|)\}, \tag{5.2}$$

where L is the number of training sequences. The conditional likelihood function is convex so that finding the global optimum is guaranteed. Since there is no closed form solution to the optimization function above, we compute the first derivative of right side of eq(5.2) with respect to λ and set it to zero, resulting in the equation below:

$$\sum_{j=1}^{L} f_k(\mathbf{x}, W_c) - \sum_{j=1}^{L} E_{P(W|x)}[f_k(\mathbf{x}, W_c)] + \Delta\Omega(\|\lambda\|) = 0 \tag{5.3}$$

The intuition of eq (5.3) is to seek the direction of λ_k where the model expectation agrees with the empirical distribution.

Given a testing sequence, our goal is to seek the segmentation configuration with the highest conditional probability defined above, i.e.

$$W^{\text{opt}} = \text{argmax} \sum_{c \in \mathcal{C}_G} \sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, W_c). \tag{5.4}$$

It can be seen that we need to compute the expectation of the features over the models in eq(5.3) and search over all possible assignments of the segmentation to ensure the maximum in eq(5.4). A naive exhaustive search would be prohibitively expensive due to the complex graphs induced by the protein structures. In addition, there are millions of sequences in the protein sequence database. Such large-scale applications demand efficient inference and optimization algorithms. It is known that the complexity of the inference algorithm depends on the graphs defined by the models. If it is a simple chain, or tree-structure, we can use exact inference algorithms, such as belief propagation. For complex graphs, since computing exact marginal distributions is in general infeasible, approximation algorithms have to be applied. There are three major approximation approaches for inference in graphical models, including sampling, variational methods and loopy belief propagation. In this chapter, we focus on surveying the possible solutions for the inference and learning problem. In the next three chapters, we develop the specific learning and inference algorithms that are most appropriate for our models and applications.

5.1 Sampling algorithm

Sampling has been widely used in the statistics community due to its simplicity. However, there is a problem if we use the naive Gibbs sampling for our conditional graphical models since the output variables $Y = \{M, \{\mathbf{w}_i\}\}$ may have different dimensions in each sampling iteration, depending on the value of M_i (the number of segments in the i^{th} sequence). The reversible jump Markov chain Monte Carlo algorithms have been proposed to handle the sampling from variable dimensions (Green, 1995). It has demonstrated successes in various applications, such as mixture models, hidden Markov models for DNA sequence segmentation and phylogenetic trees (Huelsenbeck et al., 2004; Boys & Henderson, 2001).

Reversible jump MCMC sampling Given a segmentation $\mathbf{y} = (M, \mathbf{w_i})$, our goal is propose a new move $\mathbf{y}^*$. To satisfy the detailed balance defined by the MCMC algorithm, auxiliary random variables v and v^* have to be introduced. The definitions for v and v^* should guarantee the *dimension-matching requirement*, i.e. $\dim(y) + \dim(v) = \dim(y^*) + \dim(v')$ and there is a one-to-one mapping from (y, v) to (y^*, v'), i.e. there exists a function Ψ so that $\Psi(y, v) = (y^*, v')$ and $\Psi^{-1}(y^*, v') = (y, v)$. Then the acceptance rate for the proposed transition from y to y^* is

$$\min\{1, \text{posterior ratio} \times \text{proposal ratio} \times \text{Jacobian}\} = \min\{1, \frac{P(\mathbf{y}^*|\mathbf{x})}{P(\mathbf{y}|\mathbf{x})} \frac{P(v')}{P(v)} \left| \frac{\partial(\mathbf{y}_i^*, v')}{\partial(\mathbf{y}_i, v)} \right| \},$$

where the last term is the determinant of the Jacobian matrix.

To construct a Markov chain on the sequence of segmentations, we define four types of Metropolis operators (Green, 1995):
(1) *State switching*: given a segmentation $\mathbf{y}$, sample a segment index j uniformly from $[1, M]$, and set its state to a new random assignment.
(2) *Position Switching*: given a segmentation $\mathbf{y}$, sample the segment index j uniformly from $[1, M]$, and change its starting position to a number sampled from $\mathrm{U}[p_{i,j-1}, q_{i,j}]$.
(3) *Segment split*: given a segmentation $\mathbf{y}$, propose a move with $M_i^* = M_i + 1$ segments by splitting the j^{th} segment, where j is randomly sampled from $\mathrm{U}[1, M]$.
(4) *Segment merge*: given a segmentation $\mathbf{y}$, sample the segment index j uniformly from $[1, M]$, propose a move by merging the j^{th} segment and $j + 1^{th}$ segment.

Contrastive divergence There are two main problems if we use the sampling algorithms described above, i.e. inefficient due to long "burn-in" period and large variance in the final estimation. To avoid the problem, contrastive divergence (CD) was proposed in (Welling & Hinton, 2002), in which a single MCMC move is made from the current empirical distribution data and thus reduce the computational costs dramatically. More specifically, In each step of the gradient update, instead of computing the model expectation $\langle \cdot \rangle_p$, CD runs the Gibbs sampling for up to only a few iterations and uses the resulting distribution q to approximate the model distribution p. It has been proved that the final values of the parameters by this kind of update will converge to the maximum likelihood estimation (Welling & Hinton, 2002).

The uncorrected Langevin method The uncorrected Langevin method (Murray & Ghahramani, 2004) is originated from the Langevin Monte Carlo method by accepting all the proposal moves. It makes use of gradient information and resembles noisy steepest ascent to avoid local optimal. Similar to the gradient ascent, the uncorrected Langevin algorithm has the following update rule:

$$\lambda_{ij}^{\text{new}} = \lambda_{ij} + \frac{\epsilon^2}{2}\frac{\partial}{\partial \lambda_{ij}} \log p(X, \lambda) + \epsilon n_{ij} \tag{5.5}$$

where $n_{ij} \sim N(0, 1)$ and ϵ is the parameter to control the step size. Via the contrastive divergence algorithm, only a few iterations of Gibbs sampling are needed to approximate the model distribution p.

5.2 Loopy belief propagation

Loopy belief propagation (Loopy BP) has been proven to be very effective in multiple experimental studies (Murphy et al., 1999b). The algorithm maintains a message $m_{b,q}(\mathbf{w}_{a,p})$ between pairs of vertices $\mathbf{w}_{b,q}$ and $\mathbf{w}_{a,p}$. The update from $\mathbf{w}_{b,q}$ to $\mathbf{w}_{a,p}$ is given by:

$$m_{b,q}(\mathbf{w}_{a,p}) \leftarrow \sum_{s_{b,q} \in S \cap d_{b,q} \in \text{range}(s_{b,q})} \Phi(\mathbf{w}_{b,q}, \mathbf{x}_b)\Phi(\mathbf{w}_{b,q}, \mathbf{w}_{a,p}, \mathbf{x}_b, \mathbf{x}_a) \prod_{\mathbf{w}_{i,j} \in \mathcal{T}_{b,q}/\mathbf{w}_{a,p}} m_{i,j}(\mathbf{w}_{b,q}),$$

where $\mathcal{T}_{b,q}$ is the spanning tree of $\mathbf{w}_{b,q}$. In the experiments, tree-based algorithm or random schedules can be applied to determine $\mathcal{T}$. Given the message vector m, approximate marginals can be computed as

$$p(\mathbf{w}_{a,p}) \leftarrow \frac{1}{Z_1'}\Phi(\mathbf{w}_{b,q}, \mathbf{x}_b) \prod_{\mathbf{w}_{i,j} \in \mathcal{N}_{a,p} \cap \mathbf{w}_{i,j} \neq \mathbf{w}_{b,q}} m_{i,j}(\mathbf{w}_{a,p})$$

$$p(\mathbf{w}_{b,q}, \mathbf{w}_{a,p}) \leftarrow \frac{1}{Z_2'}\Phi(\mathbf{w}_{b,q}, \mathbf{w}_{a,p}, \mathbf{x}_b, \mathbf{x}_a) \prod_{\mathbf{w}_{i,j} \in \mathcal{N}_{b,q}/\mathbf{w}_{a,p}} m_{i,j}(\mathbf{y}_{b,q}) \prod_{\mathbf{w}_{i,j} \in \mathcal{N}_{a,p}/\mathbf{w}_{b,q}} m_{i,j}(\mathbf{w}_{a,p}).$$

Then the expectation of features can be computed directly using the approximated marginal. The Loopy BP has demonstrated success in many empirical studies, and recently proved to minimize the Bethe free energy theoretically (Yedidia et al., 2000).

It is straightforward to use the loopy BP for CRF or kCRF model. However, it is not directly applicable to complex models, such as SCRF or linked SCRF, since these models allow the number of nodes in the graph to be also a variable.

5.3 Variational approximation

Variational methods exploit laws of large numbers to transform the original graphical model into a simplified graphical model in which inference is efficient (Jordan et al., 1999; Jaakkola, 2000). Mean field (MF) is the simplest variational method that approximates the model distribution p through a factorized form as a product of marginals over clusters of variables (Xing et al., 2003). It is straightforward to derive the naive mean field for the CRF model, where the conditional probability in the CRF p is approximated by an surrogate distribution q as a product of singleton marginals over the variables: $q(\mathbf{y}|\mathbf{x}) = \prod_i q(y_i|\mathbf{x})$, where $q(y_i)$ is defined as a multinomial distribution. By minimizing the KL divergence between q and p, we can get a mean field approximation of the marginals. For SCRF or linked SCRF model, we have the long-range interaction edges that make the inferences complicated. Structured variational approximation can be applied, where the surrogate distribution q is defined as a semi-Markov CRF model, i.e.

$$q = \frac{1}{Z}\exp(\sum_{i=1}^{M}\sum_{k=1}^{K}\lambda_k f_k(w_i, w_{i-1}, \mathbf{x})).$$

5.4 Pseudo point approximation

In addition to the approaches above, we can also use some naive while fast approximations by point estimation. Even though this approach is less preferred, it does find applications where the graph consists of hundreds of nodes, for example, object recognition in computer vision.

Saddle Point Approximation A straightforward approximation method is based on approximating the partition function (Z) using the saddle point approximation(SPA), that is,

$$Z \approx \exp(\sum_{\mathbf{w}^*_{i,j}\in\mathcal{V}_G}\sum_{k=1}^{K1}\lambda_k f_k(\mathbf{x}_i, \mathbf{w}^*_{i,j}) + \sum_{\langle\mathbf{w}^*_{a,p},\mathbf{w}^*_{b,q}\rangle\in\mathcal{E}_G}\sum_{k=1}^{K2}\mu_k g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}^*_{a,p}, \mathbf{w}^*_{b,q}),$$

where $\mathbf{y}^* = \text{argmax } P(\mathbf{Y}|\mathbf{x})$.This also leads to the simple approximation for the expectations, i.e.

$$E[f_k(\mathbf{x}_i, \mathbf{W}_{i,j})] = f_k(\mathbf{x}, \mathbf{y}^*_{i,j}),\ E[g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{W}_{a,p}, \mathbf{W}_{b,q})] = g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}^*_{a,p}, \mathbf{w}^*_{b,q}).$$

Maximum Margin Approximation A further simplification can be made by assuming all the mass of Z is concentrated on the maximum margin configuration, i.e. $\mathbf{w}^\star_{i,j} = \text{argmax } P(\mathbf{w}_{i,j}|\mathbf{w}_{\mathcal{N}_{i,j}}, \mathbf{x})$. Then the normalizer and expectation can be calculated using the value of $\mathbf{y}^\star$.

5.5 Alternatives to maximum a posterior

Given a testing sequence, our task is to compute $\mathbf{y}^* = \text{argmax } P(\mathbf{y}|\mathbf{x})$. There are several algorithms to compute the maximum a posterior (MAP), for example, we can use the same propagation algorithm described above, except that the summation is replaced by maximization. Other alternative solutions include:

Maximum posterior marginal Following the idea of greedy search, we can get the optimal $\mathbf{y}$ by maximizing each individual cliques, i.e.

$$\mathbf{w}_{i,j} = \text{argmax } P(\mathbf{w}_{i,j}|\mathbf{w}_{\mathcal{N}_{i,j}}, \mathbf{x}).$$

Iterated conditional modes Given an initial label configuration, iterated conditional modes (ICM) maximizes the local conditional probability iteratively, i.e.

$$\mathbf{w}_{i,j}^{(t+1)} = \text{argmax } P(\mathbf{w}_{i,j}|\mathbf{w}_{\mathcal{N}_{i,j}}^{t}, \mathbf{x})$$

In addition, the MAP problem belongs to the general task of search in artificial intelligence. Many searching algorithms, such as branch-and-bound and dead-end elimination, can be applied. Several algorithms along the direction are developed for energy minimization in protein folding, drug design and ab initio protein structure prediction (Desmet et al., 1992; Dahiyat & Mayo, 1997).

Chapter 6

Protein Secondary Structure Prediction

It is widely believed that protein secondary structures can contribute valuable information to discerning how proteins fold in three-dimensions. Protein secondary structure prediction, which projects primary protein sequences onto a string of secondary assignments, such as helix, sheet or coil, for each residue, has been extensively studied for decades (Rost & Sander, 1993; King & Sternberg, 1996; Jones, 1999; Rost, 2001). Recently the performance of protein secondary structure prediction has been improved to as high as 78 - 79% in accuracy in general and 80-85% for predicting helix and coils (Kim & Park, 2003). The major bottleneck lies in the β-sheets prediction, which involves hydrogen bonding between residues that are not necessarily consecutive in the primary structure.

The architecture of a typical protein secondary structure prediction system is outlined in Fig. 6.1. In the first step, profile generation or feature extraction ([A] in Fig. 6.1), converts the primary protein sequences to a set of features that can be used to predict the labels of secondary structures. Divergent profiles of multiple sequence alignments and a large variety of physical or biochemical features have been explored (Rost & Sander, 1993; Jones, 1999). Next, a sequence-to-structure mapping process ([B] in Fig. 6.1) outputs the predicted scores for each structure type using the features from [A] as input. Various machine learning algorithms have been applied, including neural networks (Rost & Sander, 1993), recurrent neural networks (Pollastri et al., 2002), Support Vector Machines (SVMs) (Hua & Sun, 2001) and Hidden Markov Models (HMMs) (Bystroff et al., 2000). Then, the output scores from [B] are converted to secondary structure labels. This involves considering the influence of neighboring structures by structure-to-structure mapping [C] and removing physically unlikely conformations by a Jury system [D], also referred as "filters" or "smoothers". Some systems separate [C] and [D] for explicit evaluation, while others keep them in one unit (Rost & Sander, 1993; King & Sternberg, 1996). Finally, a consensus is formed by combining predicted scores or labels from multiple independent systems into a single labeled sequence. Several methods have been applied to consensus formation, such as a complex combination of neural networks (Cuff & Barton, 2000), multivariate linear regression (Guermeur et al., 1999), decision trees (Selbig et al., 1999) and cascaded multiple classifiers (Ouali & King, 2000).

From recent advances in protein secondary structure prediction, there are three major approaches that have been demonstrated effective to improve the performance, including (1) incorporating features with statistical evolutionary information, such as PSI-BLAST (Jones, 1999), (2) combining the results of multiple independent prediction methods into a consensus prediction (Cuff & Barton, 2000), and (3) extracting coupling features from predicted tertiary 3-D structures as long-range interaction information (Meiler & Baker, 2003). Most existing systems employ a sliding window-based method, i.e. constructing the output of a specific position using the observations within a window size around it, or a simple hidden Markov model approach, both of which fail to consider the long-range interactions in the protein structures. Therefore in this thesis, we propose to tackle the problem from those three aspects using conditional graphical models (Section 6.3.1, Section 6.4, Section 6.5).

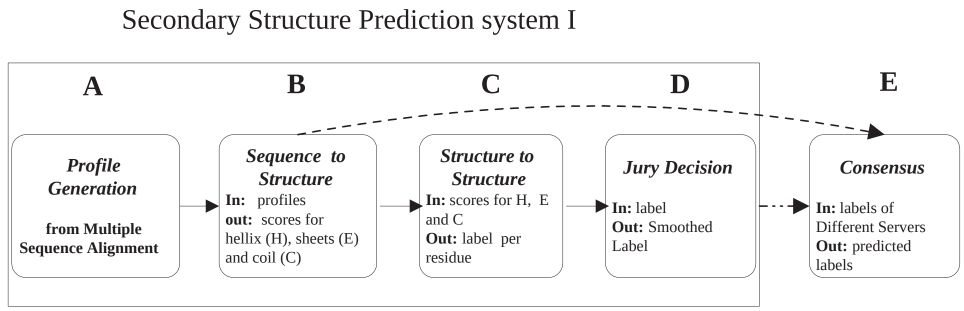

Secondary Structure Prediction system I
A
B
C
D
E
Profile Generation
from Multiple Sequence Alignment
Sequence to Structure
In: profiles
out: scores for hellix (H), sheets (E) and coil (C)
Structure to Structure
In: scores for H, E and C
Out: label per residue
Jury Decision
In: label
Out: Smoothed Label
Consensus
In: labels of Different Servers
Out: predicted labels

Figure 6.1: The architecture of a typical protein secondary structure prediction system (adapted from (Rost & Sander, 1993))

6.1 Materials and Evaluation Measure

Two datasets were used to evaluate the effectiveness of the proposed methods. One is the RS126 dataset, on which many existing secondary structure prediction methods were developed and tested (Cuff & Barton, 1999). It is a non-homologous dataset by the definition in (Rost & Sander, 1993), namely no two proteins of 126 protein chains share more than 25% sequence identity over a length of more than 80 residues. However, Cuff and Barton found that there are 11 pairs of proteins in the RS126 set that have an SD score, i.e. Z score for comparison of the native sequences given by $(V-\overline{x})/\sigma$, of greater than 5 (Cuff & Barton, 1999). Therefore in our experiments we use the datasets that intentionally removed the 11 homologous proteins to better evaluate our system. The other dataset is CB513 created by Cuff & Barton (Cuff & Barton, 1999), which most recent methods reported results on (Hua & Sun, 2001; Kim & Park, 2003; Guo et al., 2004). It consists of 513 non-homologous protein chains which have an SD score of less than 5 (Cuff & Barton, 1999). The dataset can be downloaded from the web http://barton.ebi.ac.uk/. We followed the DSSP definition for protein secondary structure assignment (Kabsch & Sander, 1983). The definition is based on hydrogen bonding patterns and geometrical constraints. Based on the discussion in (Cuff & Barton, 1999), the 8 DSSP labels are reduced to a 3 state model as follows: H & G to Helix (H), E & B to Sheets (E), all other states to Coil (C).

For protein secondary structure prediction, the state-of-art performance is achieved by window-base methods using the PSI-BLAST profiles (Jones, 1999). In our experiments, we apply a linear transformation f to the PSSM matrix elements according to

$$f(x) = \begin{cases} 0 & \text{if } (x \leq -5) \\ \frac{1}{2} + \frac{x}{10} & \text{if } (-5 \leq x \leq 5), \\ 1 & \text{otherwise.} \end{cases} \tag{6.1}$$

This is the same transform used by (Kim & Park, 2003) in the CASP5 (Critical Assessment of Structure Predictions) competition, which achieved one of the best results for protein secondary structure prediction. The window size is set to 13 by cross-validation.

Various measures are used to evaluate the prediction accuracy, including overall per-residue accuracy (Q_3), Matthew's correlation coefficients per structure type (C_H,C_C,C_E) and segment of overlap (SOV) (Rost et al., 1994; Zemla et al., 1999), and the per-residue accuracy for each type of secondary structure ($Q_H^{\text{rec}}, Q_E^{\text{rec}}, Q_C^{\text{rec}}$; $Q_H^{\text{pre}}, Q_E^{\text{pre}}, Q_C^{\text{pre}}$) (see Table 6.1 for detailed definition).

Table 6.1: Commonly used evaluation measures for protein secondary structures

Contingency Table

Predicted\True	+	−
+	l_{11}	l_{12}
−	l_{21}	l_{22}

accuracy: $Q = \frac{l_{11}+l_{22}}{l_{11}+l_{12}+l_{21}+l_{22}}$

Matthew's coefficients: $C = \frac{l_{11}*l_{22}-l_{12}*l_{21}}{\sqrt{(l_{11}+l_{12})(l_{11}+l_{21})(l_{22}+l_{12})(l_{22}+l_{21})}}$

recall: $Q^{\text{rec}} = \frac{l_{11}}{l_{11}+l_{21}}$ precision: $Q^{\text{pre}} = \frac{l_{11}}{l_{11}+l_{12}}$

$$SOV = \frac{1}{N}\sum_S \frac{\text{minov}(S^{pred},S^{true})+\text{delta}(S^{pred},S^{true})}{\text{maxov}(S^{pred},S^{true})} * \text{length}(S^{true})$$

minov(S_1, S_2): length of overlap between S_1 and S_2;

maxov(S_1, S_2): the length of extent over either S_1 and S_2

delta(S_1, S_2) $= \min(\text{maxov}(S_1,S_2) - \text{minov}(S_1,S_2), \text{minov}(S_1,S_2), \lceil \text{len}(S_1) \rceil, \lceil \text{len}(S_2) \rceil)$

6.2 Conditional Random Fields for Sequential Data

Some sequential graphical models, such as hidden Markov models (HMMs) or Markov random fields (MRFs), have been successfully applied to secondary structure prediction (Bystroff et al., 2000; Karplus et al., 1998). These methods, as generative models, assume a particular generating process of the data. It works by computing the joint distribution $P(\mathbf{x}, \mathbf{y})$ of observation $\mathbf{x} \in$ {amino acids} and state sequences $\mathbf{y} \in \mathcal{Y} =$ {secondary structure assignments}, and make predictions using Bayes rules to calculate $P(\mathbf{y}|\mathbf{x})$. Though successfully applied to many applications with sequential data, HMMs may not be the most appropriate for our task. First, it is difficult to include overlapping long-range features due to the independence assumptions. Second, generative models as HMMs, work well only when the underlying assumptions are reasonable. In contrast, discriminative models do not make any assumptions and compute the posterior probability directly. Conditional Random fields (CRFs), as a discriminative model for structured prediction, has been successfully applied to many applications, including information retrieval and computer vision, and achieved significant improvement over HMMs (Kumar & Hebert, 2003; Pinto et al., 2003; Sha & Pereira, 2003).

Conditional Random Fields (CRFs), proposed by Lafferty et al., are *undirected* graphical models (also known as *random fields*) (Lafferty et al., 2001). As a discriminative model, it calculates the conditional probability $P(\mathbf{y}|\mathbf{x})$ directly as follows:

$$P(\mathbf{Y}|\mathbf{x}) = \frac{1}{Z}\prod_{i=1}^{N}\exp(\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, i, Y_{i-1}, Y_i)),$$

where f_k can be arbitrary features, such as overlapping features or long-range interaction features. The feature weight λ_k is the model parameters. Compared with MEMMs, CRFs takes on a global normalizer Z, which results in a convex function so that the global optimal solutions of λ_k are guaranteed (Lafferty et al., 2001). The parameters λ are learnt by minimizing the regularized negative log loss of the training data, i.e.

$$\lambda = \text{argmax}\ \{\sum_{i=1}^{N}\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, i, y_{i-1}, y_i) - \log Z\}. \tag{6.2}$$

Setting the first derivative to be zero, we have

$$\sum_{i=1}^{N}\{f_k(\mathbf{x}, i, y_{i-1}, y_i) - E_{P(Y|x)}[f_k(\mathbf{x}, i, Y_{i-1}, Y_i)]\} = 0. \tag{6.3}$$

There is no closed form solution to eq(6.3) and iterative searching algorithm can be applied (Minka, 2001), among which the L-BFGS method is shown to be significantly more efficient (Sha & Pereira, 2003) (which is also confirmed in our experiments).

Similar to HMMs and MEMMs, there is still an efficient inferencing algorithm for CRFs as long as the graph is a tree or a chain. Specifically, the forward probability $\alpha_i(y)$ is defined as the probability of being in state y at time i *given* the observation up to time i; the backward probability $\beta_i(y)$ is the probability of starting from state y at time i *given* the observation sequence after time i. The recursive steps are:

$$\alpha_{i+1}(y) = \sum_{y' \in \mathcal{Y}} \alpha_i(y') \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i+1, y', y)), \quad \beta_i(y') \tag{6.4}$$

$$= \sum_{y \in \mathcal{Y}} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i+1, y', y)) \beta_{i+1}(y). \tag{6.5}$$

The normalizer Z can be computed via $Z = \sum_{y \in \mathcal{Y}} \alpha_n(y)$. The Viterbi algorithm can be derived accordingly, where $\delta_i(y)$ is defined as the best score (i.e. the highest probability) over all possible configurations of state sequence ends at the time i in state y *given* the observation up to time i. By induction, we have

$$\delta_{i+1}(y) = \max_{y' \in \mathcal{Y}} \delta_i(y') \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i+1, y', y)), \tag{6.6}$$

and $\phi_{i+1}(y)$ is used to keep track of the state configuration of time i that maximize eq(6.6).

6.3 Thesis work: CRFs for Protein Secondary Structure Prediction

Recent analysis by information theory indicates that the correlation between neighboring secondary structures are much stronger than that of neighboring amino acids (Crooks & Brenner, 2004). In literature, while feature extraction [A] and sequence-to-structure mapping [B] have been studied extensively, the structure-to-structure mapping and jury system [C, D] have not been explored in detail although they are commonly used in various systems (Figure 6.1). From a machine learning perspective, both the jury system [C, D] and the consensus [E] can be formulated as the *combination problem for sequences*: given the predicted scores or labels, how should we combine them into the final labels, taking into account the dependencies of neighbors and constraints of a single protein sequence?

Note that the combination problem for sequences is distinct from another closely-related task: given the predicted scores or labels from different systems for one residue, how can we combine them into the optimal labels? This task is a classical problem for machine learning known as an ensemble approach and many ensemble methods have been used for consensus formation. The difference between our task and the ensemble problem is that ensemble treats each residue as independent and does not consider the extra information from neighboring structures or constraints of a single sequence. Therefore our combination problem is more general and difficult than a classical ensemble problem.

6.3.1 Probabilistic Combination Methods

We formulate our combination problem as follows: given a protein sequence $\mathbf{x} = x_1 x_2 \ldots x_N$, the raw output by a secondary structure prediction system is either a label sequence $\mathbf{p} = p_1 p_2 \ldots p_N$, or a $N \times 3$ score matrix Q, where $Q_{ij} = Q_j(x_i)$ is the score of residue x_i for class j. Taking the predicted labels $\mathbf{p}$ or score matrix Q, we try to predict the true label $Y_1 Y_2 \ldots Y_N$. Without loss of generality, we assume that (1) the predicted scores are non-negative and normalized; (2) for one residue x_i, the higher the score Q_{ij}, the larger the probability that the residue x_i belongs to class j.

Previously proposed methods for combination are mostly window-based, which include:

Window-Based Method for Label Combination The standard method for converting scores to predicted secondary structure labels is to assign the class with the highest score. After that, many systems employ rule-based methods to improve upon the first-pass assignment, i.e. the *label combination.* (Rost & Sander, 1993; King & Sternberg, 1996). The window-based label combination works as follows: given the labels predicted by a system $p_1 p_2 \ldots p_N$, and the window size R, let $D = (R-1)/2$ be the half of the window size. The input features for residue x_i are the predicted labels within the window R, i.e. $\langle p_{i-D}, p_{i-D+1}, \ldots, p_{i+D-1}, p_{i+D} \rangle$ (a null label is assigned if the label does not exist). Then a rule-based classifier, such as decision tree or CART, is applied to make the outputs easy for the biologists to interpret (Rost & Sander, 1993). The window size R is a parameter with which we can tune the trade-off between including useful information and excluding "noisy" more remote features.

Window-Based Method for Score Combination In current secondary structure prediction systems, *score combination* is used widely. Window-based score combination works similar to label combination except: (1) the input features are scores Q instead of labels; (2) more powerful classifiers, such as neural networks and k-Nearest-Neighbor, are used instead of rule-based classifiers. Empirically, score combination has demonstrated more improvement in accuracy than label combination since the score indicates the confidence of the prediction and thus contains more information than a single label (Rost & Sander, 1993; Salamov & Solovyev, 1995; Jones, 1999; Guo et al., 2004).

The window-based combination approach has the disadvantages of considering the local information only. Conditional graphical models have been proved to achieve the best performance for applications of structured data prediction (Kumar & Hebert, 2003; Pinto et al., 2003; Sha & Pereira, 2003). In addition to CRFs, there are also alternative models, such as MEMMs and its extensions, that could be used for our combination task.

Maximum Entropy Markov Models (MEMM) The graphical representation of MEMMs is shown in Figure 6.2-(A). Similar to CRFs, MEMMs calculates the conditional probability $P(\mathbf{Y}|\mathbf{x})$ directly but uses a local normalizer over each position (McCallum et al., 2000), i.e.:

$$P(\mathbf{Y}|\mathbf{x}) = \prod_{i=1}^{N} \frac{1}{Z_i} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, i, Y_{i-1}, Y_i))$$

where Z_i is a normalizing factor over position i. Compared with CRFs, MEMMs can also handle arbitrary, non-independent features f_k. There is also an efficient dynamic programming solution to the problem of identifying the most likely state sequence *given* an observation. In addition, MEMMs are much cheaper computationally but suffer from local optimal solutions to parameter estimation and the label bias problem, namely the total probability "received" by y_{i-1} must be passed on to labels y_i at time i even if x_i is completely incompatible with y_{i-1} (see (Lafferty et al., 2001) for full

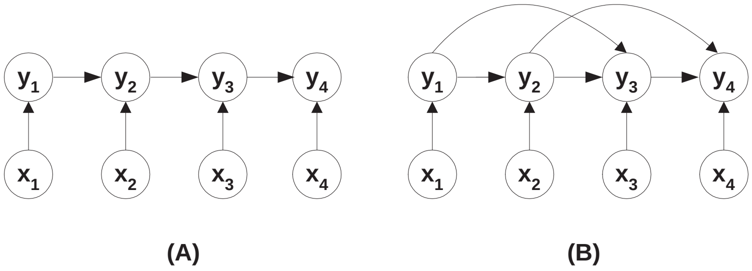


Figure 6.2: The graph representation of MEMM (A) and high-order MEMM (B)

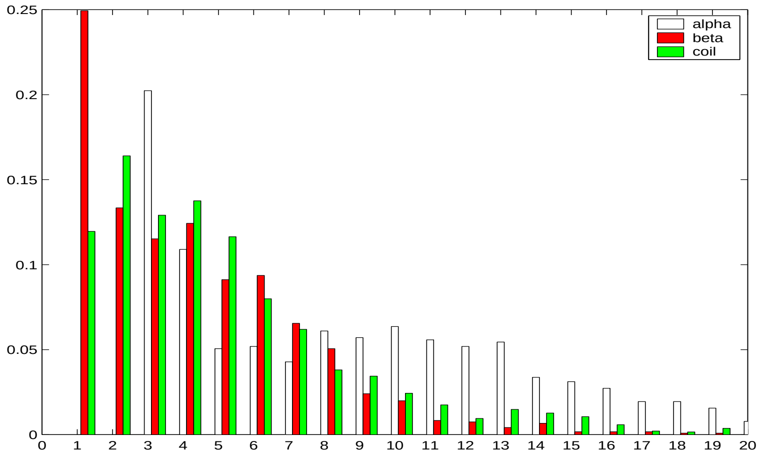


Figure 6.3: The distribution of the segment length for different structures

discussion). For score combination, we define the features to be

$$f_{\langle k_1,k_2\rangle}(\mathbf{x}, i, y_{i-1}, y_i) = \begin{cases} Q_{ik_2} & \text{if } y_{i-1} = k_1 \text{ and } y_i = k_2; \\ 0 & \text{otherwise.} \end{cases} \tag{6.7}$$

Higher-order MEMMs (HOMEMMs) MEMMs assume the first-order Markov assumption, i.e. $P(y_{i+1}|y_i) = P(y_{i+1}|y_i, y_{i-1})$. On one hand, it simplifies the model and reduces the computational cost dramatically; on the other hand, this assumption is clearly inappropriate for secondary structure prediction, where the structure dependencies extend over several residues and even involve long-distance interactions. To solve this problem, higher-order MEMMs can be developed (Rabiner, 1989). For simplicity, we only consider second-order MEMMs, in which the next state depends upon a history with two previous states (see Fig. 6.2-B). A second-order MEMMs can be transformed to an equivalent first-order Markov Model by redefining the state $\hat{y}_i$ as $\hat{y}_i = \langle y_i, y_{i-1}\rangle \in \mathcal{Y} \times \mathcal{Y} = \mathcal{Y}^2$. In secondary structure prediction the set of new states is $\mathcal{Y}^2 = \{$HC, HE, HH, EC, EE, EH, CC, CE, CH$\}$ and the features can be redefined accordingly.

Pseudo State Duration MEMMs (PSMEMMs) Higher-order MEMMs provide a solution to circumvent the state independence assumptions. However, the number of new states and features is an exponential function of the order o, which makes the computational costs intractable when o grows large. To solve the problem, we devise a heuristic method which is able to encompass more history information with the same computational cost as MEMMs, namely pseudo state duration MEMM. Our heuristics are based on the observation that the distribution of the segment length varies for different structures, as shown in Fig. 6.3 (only segments less than 20 residues are shown). From the graph, we can see that different segment lengths are preferred by different secondary structures. To incorporate such kind of information, we define $P(y|y', d)$ as the probability that the current state is y given the recent history of d consecutive y', i.e.

$$P(y|y', d) = \frac{\#\text{ of occurences } \bar{y}'y'y'\ldots y'y}{\#\text{ of occurences } \bar{y}'y'y'\ldots y'}.$$

Data sparsity problems might occur when d grows larger and it can be addressed by smoothing methods, such as Laplace smoothing. All the algorithms and definitions are similar as MEMMs except that we use another kind of features as below:

$$f_{<k_1,k_2,d>}(\mathbf{x}, i, y_i, y_{i-1}) = \begin{cases} Q_{ik_2} P(y_i|y_{i-1}, d) & \text{if } y_i = k_1 \text{ and } y_{i-1} = k_2 \\ 0 & \text{otherwise.} \end{cases}$$

6.3.2 Experiment Results

Table 6.2 summarizes the representation power of the graphical models discussed above. We can see that all the models except HMMs have the flexibility to allow arbitrary features over the observation and therefore are good for score combination. Table 6.3 lists the results of the window-based methods:

Table 6.2: Summary: pros and cons of different conditional graphical models

	1st-order Markov	Label Bias	Flexibility of Features	Globally Optimal
HMMs	+	+	−	−
MEMMs	+	−	+	−
HOMEMMs	−	−	+	−
PSMEMMs	−	−	+	−
CRFs	+	+	+	+

- Generally speaking, the window-based score combination improved the prediction more than the label combination. This confirms our expectation since the scores contain more information than a single label.
- The label combination resulted in maximum improvement for predicting helices rather than other structures. King and Sternberg reported a similar observation and showed that the extracted rules are most relevant to helices (King & Sternberg, 1996).
- The prediction accuracy has increased for both helices and sheets by score combination.

In terms of the graphical models for score combination, we examined the four methods discussed before. To fairly compare with window-based methods, only the score features are used for the prediction, although we believe incorporating other features will improve the predictions more. Table 6.4 shows the results of the four graphical models for score combination:

- Generally speaking, the graphical models for score combination are consistently better than the window-based approaches, especially in SOV measure.
- For the MEMMs, the prediction accuracy using Viterbi algorithm is better than using marginal mode. It is interesting to note that the opposite is true for CRFs.
- Compared with MEMMs, HOMEMMs and PSMEMMs improve SOV slightly since these methods consider more history information. However, there is little difference in performance between HOMEMMs and PSMEMMs. This might indicate that higher-order MEMMs will hardly add more value than second-order MEMMs.
- CRFs perform the best among the four graphical models. It exhibits moderate improvements for predicting helices and especially sheets. Global optimization and removing label bias seem to help since these are the only differences between MEMMs and CRFs.

Table 6.5 summarizes our discussion above and provides a qualitative estimation of computational costs as well as the performance for each method.

In this section, we surveyed current secondary structure prediction methods and identified the combination problem for sequences: how to combine the predicted scores or labels from a single or multiple systems with the consideration of neighbors and long-distance interactions. Our experiments

Table 6.3: Results of protein secondary structure prediction on CB513 dataset using window-based combination methods

Combination Method	SOV(%)	Q_3(%)	Q_H^{rec}(%)	Q_C^{rec}(%)	Q_E^{rec}(%)	Q_H^{pre}(%)	Q_C^{pre}(%)	Q_E^{pre}(%)	C_H	C_C	C_E
None	75.6	76.7	78.0	83.2	62.7	83.6	72.1	77.2	0.71	0.58	0.62
Dtree	75.7	76.7	78.0	83.2	62.8	**83.7**	72.1	77.1	**0.72**	0.58	0.62
SVM	75.7	76.9	**81.4**	76.7	**70.5**	82.1	75.2	72.2	**0.72**	0.58	**0.63**

Table 6.4: Results on CB513 dataset using different combination strategies. MEMM^p, CRF^p: p refers to different way to compute the labels; $p = 1$: marginal model; $p = 2$: Viterbi algorithm

Combination Method	SOV(%)	Q_3(%)	Q_H^{rec}(%)	Q_C^{rec}(%)	Q_E^{rec}(%)	Q_H^{pre}(%)	Q_C^{pre}(%)	Q_E^{pre}(%)	C_H	C_C	C_E
None	75.6	76.7	78.0	83.2	62.7	83.6	72.1	77.2	0.71	0.58	0.62
MEMM^1	75.6	76.7	77.8	83.6	62.1	83.7	71.8	77.8	0.71	0.58	0.62
MEMM^2	**76.0**	76.8	78.2	83.4	62.2	83.7	72.0	**78.0**	0.71	0.58	0.62
HOMEMMs^2	**76.1**	**76.9**	**78.3**	83.4	62.4	83.6	72.1	77.9	0.71	0.59	0.62
PSMMEMMs^2	**76.1**	**76.9**	**78.3**	83.3	62.2	83.6	72.0	**78.0**	0.71	0.58	0.62
CRF^1	**76.2**	**77.0**	**78.3**	83.4	**63.4**	83.7	72.1	**78.0**	**0.72**	0.58	**0.63**

Table 6.5: Summary of computational costs and effectiveness for different combination strategies. H/L/M: high/low/medium computational costs; +/−: improvement/no improvement over the baseline results without combination

	Train	Test	Helices	Sheets	Coil	Segment
DTree	*M*	*L*	+	−	−	−
SVM	*H*	*H*	+	+	−	−
MEMMs	*H*	*L*	−	−	−	+
HOMEMMs	*H*	*L*	−	−	−	+
PSMEMMs	*H*	*L*	−	−	−	+
CRFs	*H*	*L*	+	+	−	+

show that graphical models are consistently better than the window-based methods. In particular, CRFs improve the predictions for both helices and sheets, while sheets benefitted the most. Our goal is to evaluate different combination methods and provide a deeper understanding of how to effectively improve secondary structure prediction. Although our discussion is focused on combining predictions from a single secondary structure prediction system, all the methods discussed above can be applied to combine results from different systems and include other physio-chemical features.

6.4 Thesis work: CRFs for Parallel and Anti-parallel β-sheet Prediction

As discussed in the previous section, the major bottleneck for current structure prediction systems is the β-sheets, which involves long-range interactions in 3-D space. Therefore designing an algorithm that effectively detects β-sheets not only will improve the prediction accuracy of secondary structures, but also helps to determine how they aggregate on each other to form tertiary structures. In this section, we focus on the prediction of parallel and antiparallel β-sheets. Specifically, we are interested in answering two questions: (1) given a residue in a protein sequence, how to accurately predict whether it belongs to β-strands or not? (2) given the secondary structure assignment of a protein, how to predict which two strands form a parallel or antiparallel pair and how each β-strand pair is aligned?

The first question can be seen as a standard secondary structure prediction problem except that we are only concerned with a binary classification problem (β-sheet or non-β-sheet). Therefore all the approaches for general secondary structure prediction can be applied. In addition, various approaches have been proposed specifically to capture the long-range interaction properties of β-sheets. Mamitsuka & Abe used stochastic tree grammars for β-sheet detection (Mamitsuka & Abe, 1994). Pollastri et al. applied bi-directional recurrent neural networks (BRNN) to both 3-state and 8-state secondary structure prediction (Pollastri et al., 2002). In the meantime, there are also a lot of attempts to address the second question. Baldi et al. extracted a number of statistics informative of the β-strands, then feed them into a bi-directional recurrent neural network (BRNN). Steward & Thornton developed a set of tables with the propensities to form β-sheets for each pair of amino acids using an information theoretic approach (Steward & Thornton, 2002).

6.4.1 Feature extraction

Previous methods have boosted the prediction of β-sheet to some extent, however, the accuracy is still very low and the problem is far from being solved. Recently Meiler & Baker improved the secondary structure prediction accuracy by 7-10% on average by extending the sequence alignment profiles to the non-local tertiary structure neighbors as an additional input (Meiler & Baker, 2003). This demonstrates that close neighbors in three-dimensional space contain very useful information for structure prediction. By knowing the alignment of the β-sheets, we can directly infer most of the non-local close neighbors. Therefore we propose to improve the β-sheet detection by combining two kinds of features in CRFs, including the predicted alignments of β-strands as long-range information, and the window-based local information.

Long-range interaction features for β-sheets Many sequence-based statistics and physico-chemical properties have been investigated for predicting the β-sheet partnership. The features we use for detecting the β-strand pairing include: the pairwise information values (Steward & Thornton, 2002), the distances between paired strands and the lengths of parallel and antiparallel β-strands respectively.

A pairwise information values were derived for the preferences of an amino acid for the residues on its pairing strand, following an information theory approach (Steward & Thornton, 2002). The values are the self-information scores s-score(A_1), which accounts for the propensities of amino acid A_1 in a parallel (or antiparallel) β-strand, and pair-information scores p-score(A_1, A_2, m), which calculates the propensities of an amino acid A_1 to have another amino acid A_2 on its pairing strand with an offset of m. The total score for a particular alignment of $x_i x_{i+1} \dots x_{i+w}$ and $x_{i'} x_{i'+1} \dots x_{i'+w}$ (w is the length of the segment) is the sum of the self-information value and the pair-information value, i.e.

$$\begin{aligned}\text{pairwise score} &= \sum_{j=1}^{w} (\text{s-score}(x_{i+j}) + \text{s-score}(x_{i'+j}) + \\ &\quad \sum_{k=-2}^{2} (\text{p-score}(x_{i+j}, x_{i'+j}, k) + (x_{i'+j}, x_{i+j}, k))) \quad (6.8)\end{aligned}$$

Histograms of distances between parallel and antiparallel β-strand pairs are plotted against the non-homologous 2013 protein sequences in the training set of PSIPRED (Jones, 1999) (see Fig. 6.4-(I, II)). From the plots, we can see that (1) the number of antiparallel strands is much larger than that of parallel strands. The ratio is around 2.5:1; (2) the average distance between parallel strand pairs is much longer than that of antiparallel pairs since the anti-parallel β-strands are usually connected by a short β-turn while the other comes with a long α-helix to form a $\beta - \alpha - \beta$ motif. From Fig. 6.4 (III, IV), we can see that the lengths of antiparallel β-strands are generally shorter than the parallel ones.

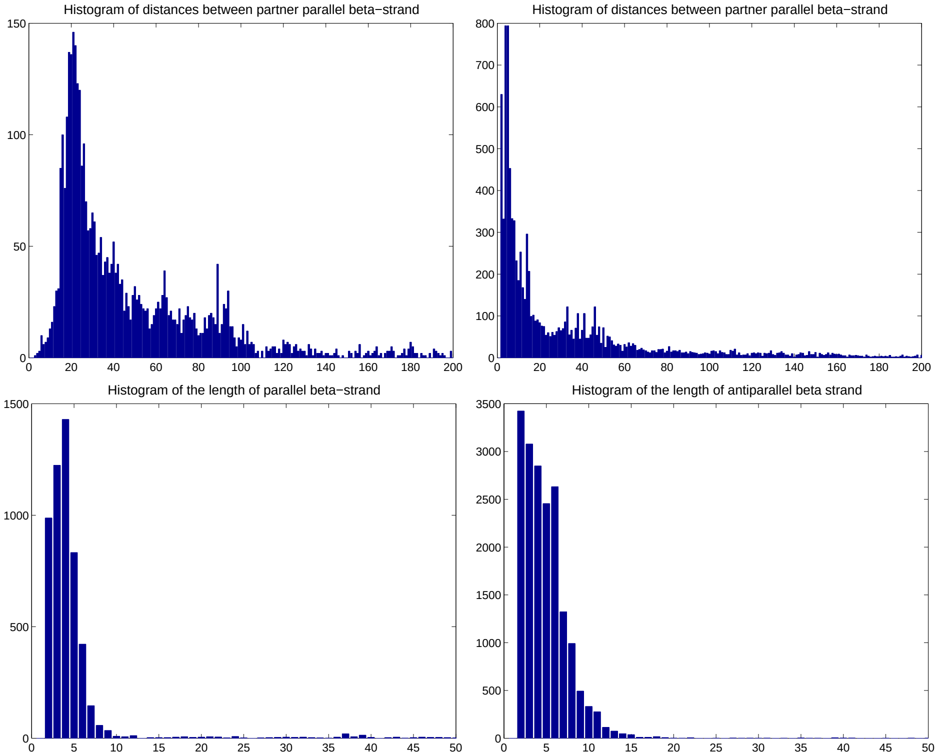


Figure 6.4: Histograms of the distances between paired strands in parallel and antiparallel β-strand (I, II); Histograms of the lengths of parallel and antiparallel β-strand (III, IV)

β-sheet alignment prediction We formulate the problem of β-sheet alignment as follows: given a protein sequence $\mathbf{x} = x_1 x_2 \ldots x_N$ and the secondary structure labels (or predicted labels) $\mathbf{y} = y_1 y_2 \ldots y_N$, where $y_i \in \{H, E, C\}$, predict the β-strand pairs for each residue x_i if its assignment y_i is E and the direction of the alignment for each pair (parallel or antiparallel). Notice that by definition the number of paired residues for each amino acid can be 1 or 2 only. To identify the β-sheet alignment, we use an exhaustive search over all possible alignments of all pairs of β-strands in the sequence. The detailed algorithm is shown in Table 6.6.

Table 6.6: Alignment algorithm for parallel and antiparallel β-sheet prediction

Input:	a set of β-strand segments in the query protein sequence $\{x_{j1}x_{j2}\ldots x_{jw_j} \| j = 1 \ldots B\}$, where B is the number of segments
Output:	a set of residue pairs and the alignment direction for each pair $\{(x_i, x_j, R_{ij})\}$, where $R_{ij} \in \{\text{parallel, antiparallel}\}$
Step 1:	Initialize the active lists $\mathcal{A} = \emptyset$, and set $\mathcal{B} = \emptyset$
Step 2:	For each pair of segments, find the alignment and its direction whose alignment score is the highest and put it into the active list $\mathcal{A}$. The alignment score is defined as a linear combination of the long-range features.
Step 3:	Sort the active list $\mathcal{A}$
Step 4:	Iterate until $\mathcal{A} = \emptyset$ or all the residues has more than 1 pairs in $\mathcal{B}$: Remove the alignment ϖ with the highest score in current $\mathcal{A}$; if any residue in ϖ has no more than 2 paired residues in $\mathcal{B}$, put ϖ in $\mathcal{B}$;
Step 5:	Output the residue pairs and their alignment directions in $\mathcal{B}$

6.4.2 Experiment Results

If the true secondary structure assignments are not available, we use the predicted labels instead and set a threshold on the alignment scores as additional stopping criterion in step 4. We use the offset of the predicted alignment from the correct register to evaluate the quality of the alignment (Steward & Thornton, 2002). The distribution of the offset for those correctly predicted pairs is shown in Figure 6.5. From the results, we can see that around 55% of the alignment has been predicted correctly and over 90% percent of the alignment has an offset of less than 2. Given the encouraging results, we can use this information as additional 3-D features to improve our first-round secondary structure prediction.

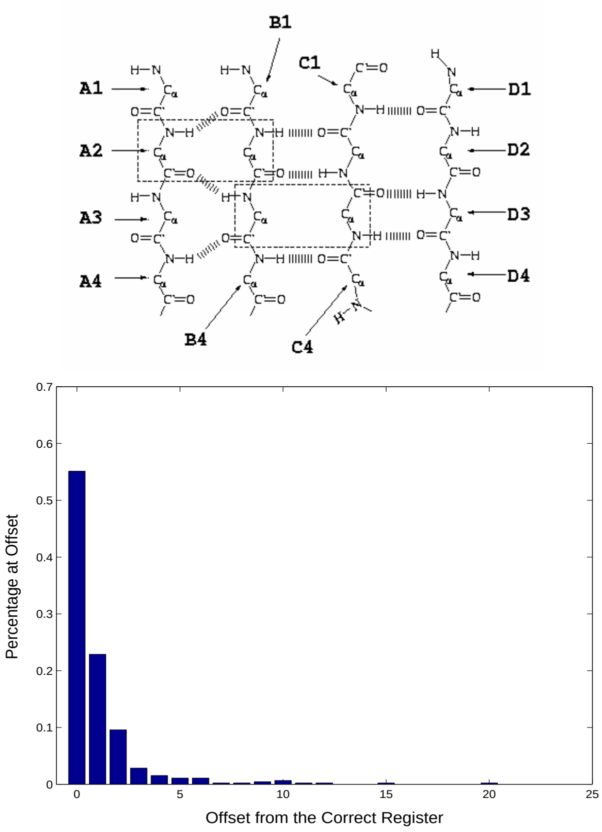


Figure 6.5: Histograms of Offset from the correct register for the correctly predicted β-strand pairs on CB513 dataset

Table 6.7: Comparison results of β-sheet prediction using differen approaches on RS-126 dataset and CB513 dataset by seven-fold cross-validation

	RS126 dataset				CB513 dataset			
Method	Q_2	Q_E^{rec}	Q_E^{pre}	C_E	Q_2	Q_E^{rec}	Q_E^{pre}	C_E
Window-based method	87.48	60.30	75.87	60.10	88.30	64.73	77.00	63.51
β-strand alignments	76.56	50.34	55.10	N/A	72.25	48.33	53.88	N/A
CRFs	88.20	64.78	74.98	**61.50**	89.43	68.93	75.68	**64.49**

Further improvement on general β-sheet prediction Based on our discussion in Section 6.3.1, CRFs have been proved to be most effective for score combination and handling the long-range interactions. Therefore in the refined method, we have two types of features for CRFs: one is the prediction scores using window-based method, which is the same setting as Section 6.3.1; the other is the long-range information, i.e. the strand pairing information defined as follows:

$$f_{\langle k_1,k_2 \rangle}(x_i, i, y_{i-1}, y_i) = \begin{cases} 1 & \text{if } y_{i-1} = k_1, y_i = k_2 \text{ and } x_i \in \mathcal{B}; \\ 0 & \text{otherwise.} \end{cases} \tag{6.9}$$

where $\mathcal{B}$ is output from the alignment algorithm. In this way, we can combine both the local information and long-range interactions for better prediction. Table 6.7 lists the results of our refined method compared with other approaches. From the results, we can see that our algorithm considerably helps the prediction for β-sheets, especially in sensitivity with around 6% improvement.

6.5 Thesis work: Kernel CRFs for Secondary Structure Prediction

Our previous results have demonstrated that CRFs can effectively handle the long-range interactions in β-sheets for score combination. In all of the work, however, conditional random fields are based on explicit feature representations. Since it is still unclear how the sequences encode the evolutionary information to determine the structures and functions, it would help to improve the predictions if we can explore this implicit information from the multiple sequence alignment. In this section, an extension of conditional random fields, kernel conditional random fields (kCRFs), is used to permit the use of implicit features spaces through kernels (Lafferty et al., 2004).

6.5.1 Kernel Conditional Random Fields

Similar to CRFs, kernel CRFs defines the conditional probability as the following form:

$$P(\mathbf{Y}|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \exp f^*(\mathbf{x}, c, Y_c), \tag{6.10}$$

where $f(\cdot)$ is the kernel basis function, i.e. $f(\cdot) = K(\cdot, (\mathbf{x}, c, y_c))$. One way to define the kernels over a structured graph is a factorization of a kernel over the observations and an indicator function over the labels, i.e. $K((\mathbf{x}, c, y_c), (\mathbf{x}', c', y'_c)) = K((\mathbf{x}, c), (\mathbf{x}', c'))\delta(y_c, y'_c)$. By representer theorem, the minimizer of the regularized negative log-loss

$$R_\phi(f^*) = \sum_l \sum_c f^*(\mathbf{x}_l, c, y_c) + \frac{\eta}{2}||f||_K^2$$

has the form

$$f^*(\cdot) = \sum_{j=1}^{L} \sum_{c \in \mathcal{C}_{G(j)}} \sum_{y_c \in \mathcal{Y}^{|c|}} \lambda_{y_c}^{(j)} K(\cdot, (\mathbf{x}^{(j)}, c, y_c)). \tag{6.11}$$

Notice that the dual parameters λ depend on all the clique label assignments, not limited to the true label, which results in an extremely large number of parameters. Therefore a greedy clique selection algorithm is proposed to incrementally select cliques that reduce the regularized risk. The algorithm maintains an *active set* of cliques with labels, where each candidate clique can be represented by a basis function $h(\cdot) = K(\cdot, (\mathbf{x}_l, c, y_c)) \in H_K$. To evaluate a candidate h, one strategy is to compute the *gain* $\sup_\lambda R_\phi(f) -$

$R_\phi(f + \lambda h)$, and choose the candidate h having the largest gain. This presents an apparent difficulty, since the optimal parameter λ cannot be computed in closed form, and must be evaluated numerically. For sequence models this would involve forward-backward calculations for each candidate h, the cost of which is prohibitive. Therefore the functional gradient descent approach is adopted, which evaluates a small change to the current function. For a given candidate h, consider adding h to the current model with small weight ε; thus $f \mapsto f+\varepsilon h$. we have $R_\phi(f+\varepsilon h) = R_\phi(f)+\varepsilon dR_\phi(f,h)+O(\varepsilon^2)$, where the functional derivative of R_ϕ at f in the direction h is computed as

$$dR_\phi(f,h) = E_f[h] - \widetilde{E}[h] + \eta\langle f,h\rangle_K \tag{6.12}$$

where $\widetilde{E}[h] = \sum_j \sum_c h(\mathbf{x}^{(j)}, c, y_c)$ is the empirical expectation and $E_f[h]$ is the model expectation conditioned on x. The idea is that in directions h where the functional gradient $dR_\phi(f,h)$ is large, the model is mismatched with the labeled data; this direction should be added to the model to make a correction. An alternative to the greedy functional gradient descent algorithm above is to estimate parameters α_h for each candidate using mean field approximation. A quasi-Newton method can be used to estimate the parameters to $\sup_\lambda \Delta R_\phi(f,h)$.

6.5.2 Experiment Results

The expensive computational costs of kCRFs prevent us from large scale evaluation. Therefore in our experiment, we use the RS126 dataset with a subset of 5 and 10 sequences respectively as the training data and the rest as testing data. For each size we perform 10 trials where the training sequences are randomly sampled from the whole set. The input features to kCRFs are PSI-BLAST profiles and 300 cliques are selected using greedy clique selection algorithm. We compare the results with other state-of-art methods using window-based method with SVM classifier. All methods use the same RBF kernel and the results are shown in Table 6.8. From the results, we can see that kCRFs achieve slight improvement than SVM in overall prediction accuracy.

Further information can be obtained by studying the transition boundaries, for example, the transition from “coil” to “sheet.” From the point of view of structural biology, these transition boundaries may provide important information about how proteins fold in three dimension and those are the positions where most secondary structure prediction systems will fail. The transition boundary is defined as a pair of adjacent positions $(i, i+1)$

	5 protein set		10 protein set	
Method	Accuracy	std	Accuracy	std
kCRF (v)	0.6625	0.0224	0.6933	0.0276
kCRF (v+e)	0.6562	0.0202	0.6933	0.0272
SVM	0.6509	0.0307	0.6875	0.0235

Table 6.8: Per-residue accuracy of different methods for secondary structure prediction, with the RBF kernel. kCRFs (v) uses vertex cliques only; KCRF (v+e) uses vertex and edge cliques.

	5 protein set		10 protein set	
Method	Accuracy	std	Accuracy	std
KCRF (v)	0.1097	0.0271	0.1462	0.0235
KCRF (v+e)	0.1114	0.0250	0.1522	0.0214
SVM	0.0667	0.0313	0.1066	0.0311

Table 6.9: Transition accuracy with different methods.

whose true labels differ. We have a hard boundary definition, i.e. it is classified correctly only if both labels are correct. This is a very hard problem, as can be seen in Table 6.8, Table 6.9, and kCRFs are able to achieve a considerable improvement over SVM.

6.6 Summary

By now we have studied the use of conditional graphical models for protein secondary structure prediction from three perspectives, including score combination, β-sheet prediction and allowing kernels to explore the evolutionary information within the sequence. The experiment results demonstrate improvement over the state-of-art methods and therefore confirm our hypothesis of graphical models for protein structure prediction.

As we know, protein secondary structure prediction has been extensively studied for decades (Cuff & Barton, 1999; Rost, 2001) and every breakthrough is directly associated with the advances of sequence analysis and the accumulation of more structural data. The current prediction accuracy is still around 80% and far from the predicted upper-bound of 85-90% (Rost, 2001). The solutions are many-folds: one direction is to collect all the possible arrangements of protein folds, which are believed to be a very limited number, and then search against these folds when given a new protein sequence; another direction is to provide a deeper understanding of the protein folding process and discover new informative biological features. We believe the graphical models, combined with advances in these directions, will bring a new breakthrough in this area.

Chapter 7

Protein Tertiary Structure Prediction

It is widely believed that protein structures reveal important information about the function, activity, stability and subcellular localization of the proteins, and the mechanisms of protein-protein interactions in cells. An important issue in inferring tertiary structures from amino-acid sequences is how to accurately identify supersecondary structures (also termed "protein folds") arising from typical spatial arrangements of well-defined secondary structures. *In silico* protein super-secondary structure prediction (or protein fold recognition) seeks to predict whether a given protein sequence contains a putative structural fold (usually represented by a training set of instances of this fold) and if so, locate its exact position within the sequence.

Traditional approaches for protein fold prediction either search the sequences in the database that are similar to the training sequences, such as PSI-BLAST (Altschul et al., 1997), or match against an HMM profile built from sequences with the same fold, such as SAM or HMMER (Krogh et al., 1994; Durbin et al., 1998; Karplus et al., 1998). To date, there has been significant progress in predicting certain types of well-defined supersecondary structures, such as $\alpha\alpha$-hairpins and β-turns, using sequence similarity based approach. However, these methods work well for simple folds with strong sequence conservations, however, fail when the sequence similarity across proteins is poor and/or there exist long-range interactions between elements in the folds such as those containing β-sheets. These cases necessitate a more expressive model, which is able to capture the structure-revealing features (e.g. the long range interactions) shared by all proteins with the same fold.

7.1 Materials and Evaluation Measure

In this chapter of the thesis, we are trying to solve the problem of tertiary fold (motif) recognition. Specifically, our task starts with a target fold F that the biologists are interested in. There are no constraints about F, which can be either a supersecondary structure with a few number of secondary structure elements, or a large complex fold occupying the whole domain. All the proteins with resolved structures deposited in the PDB can be classified into two groups, i.e. those take the target fold F and those not. These proteins together with the labels can be used as training data. Our goal is to predict whether a testing protein, without resolved structures, takes the fold F in nature or not; if they do, locate the starting and ending positions of the subsequence that takes the fold.

Our task involves two sub-tasks: one is the classification problem, that is, given a set of training sequences $X_1, X_2, \ldots, X_N$ and their labels $y_1, y_2, \ldots, y_N$ ($y_i = 0, 1$), predict the label of a new testing sequence X_{new}; the other subtask is not that straightforward to describe in mathematical settings. We can think of the target fold as some patterns (or motifs in bioinformatics terminology). Given a set of instances of the pattern, including both the positive examples (subsequences with the pattern F) and the negative examples (sequences without the pattern F), we want to predict whether the pattern appears in any subsequence of the testing proteins. The first question can be answered easily if we can solve the second one successfully. A key problem in the second task is how we can represent the descriptive patterns (or motifs) using mathematical notations.

This task falls within the general studies in protein fold (or motif) classification, but differs in two aspects: first, the target fold comes directly from the focused study and experiments by the biologists (in our case, the collaborators that we worked with have been studying a particular fold for a long time), rather than from the databases of common folds. Usually the positive proteins with resolved structures are quite limited, although the fold is believed to be common in nature. Second, the problem we aim to address is much more difficult than the common fold classification because we do not have as many positive examples and they do not share high sequence similarities. In other words, the patterns that we are trying to identify have not been represented clearly in the training data. This is the main motivation why we want to develop a richer graphical model, rather than a simple classifier. Notice that our models can be used in the traditional fold recognition or threading setting, however, its complexities can be paid off best in predicting those difficult protein folds.

To testify the effectiveness of different recognition models, we choose the right-handed β-helix and leucine-rich repeats as examples in our experiments:

The right-handed parallel β-helix fold is an elongated helix-like structure with a series of progressive stranded coilings (called *rungs*), each of which is composed of three parallel β-strands to form a triangular prism shape (Yoder et al., 1993). The typical 3-D structure of a β-helix is shown in Figure 7.3(A-B). As we can see, each basic structural unit, i.e. a rung, has three β-strands of various lengths, ranging from 3 to 5 residues. The strands are connected to each other by loops with distinctive features. One loop is a unique two-residue turn which forms an angle of approximately $120^\diamond$ between two parallel β-strands (called *T-2 turn*). The other two loops vary in size and conformation, which might contain helix or even β-sheets. The β-helix structures are significant in that they include pectate lyases, which are secreted by pathogens and initiate bacterial infection of plants; the phage P22 tailspike adhesion that binds the O-antigen of Salmonella typhimurium; and the P.69 pertactin toxin from Bordetella pertussis, the cause of Whooping Cough. Therefore it would be very interesting if we can accurately predict other unidentified β-helix structure proteins.

The leucine-rich repeats are solenoid-like regular arrangement of β-strand and α-helix, connected by coils. They are believed to be prevalent in proteins and can involve in a wide spectrum of cellular and biochemical activities, such as various protein-protein interaction processes (Kobe & Deisenhofer, 1994). There are 41 LLR proteins with known structure in PDB, covering 2 super-families and 11 families in SCOP. The LLR fold is relatively easy to detect due to its conserved motif with many leucines in the sequence and short insertions. Therefore it would be more interesting to discover new LLR proteins with much less sequence identity to previous known proteins.

7.2 Thesis work: Segmentation CRFs for General Protein Fold Recognition

Protein folds or super-secondary structures are frequent arrangement patterns of several secondary structural components: some components are quite conserved in sequences or prefer a specific length, and some might

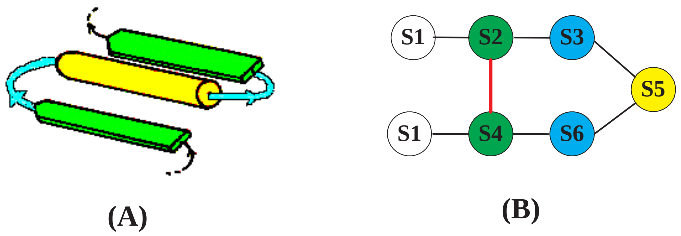


Figure 7.1: Graph structure of β-α-β motif (A) 3-D structure (B) Protein structure graph: node: Green=β-strand, yellow=α-helix, cyan=coil, white=non-β-α-β (I-node); edge: $E_1 = \{\text{black edges}\}$ and $E_2 = \{\text{red edges}\}$.

form non-covalent bonds with each other, such as two β-strands in a parallel β-sheet. To model the protein fold better, we define the models based on the protein structural graph, in which the nodes represent secondary structure modules of fixed or various length (instead of individual residues) and the edges between nodes indicate the interactions of the corresponding secondary structure elements in 3-D. A segmentation conditional random fields can be used to define a probability distribution over all possible structural configurations (i.e., segmentations and functional labeling of the delineated segments) underlying a given protein sequence. Given a protein sequence, we can search for the best segmentation defined by the graph and determine if the protein has the fold.

7.2.1 Segmentation Conditional Random Fields

Before delving into the details of the model, we first define the protein structural graph, which is an annotated graph $G = \{V, E\}$, where V is the set of nodes corresponding to the specificities of structural units such as motifs, insertions or the regions outside the fold (which are unobserved and to be inferred), and the amino acid residues at each position (which are observed and to be conditioned on). E represents the set of edges denoting dependencies between the objects represented by the nodes, such as location constraints (e.g. state transitions between adjacent nodes in the sequence order), or long-range interactions between non-neighboring motifs and/or insertions (e.g. hydrogen bonding between two component β-strands). The latter type of dependencies is unique to the protein structural graph for complex folds and causes much of the difficulties in solving such graphical models. Figure 7.1 shows an example of β-α-β motif.

In practice, one protein fold might correspond to several reasonable structural graphs given different semantics for one node. There is always a tradeoff between the graph complexity, fidelity of model and the real computational costs. Therefore a good graph is the most expressive one that captures the properties of the protein folds while retaining as much simplicity as possible. There are several ways to simplify the graph, for example we can combine multiple nodes with similar properties into one, or remove some edges that are less important or less interesting to us (notice that currently all the protein structural graphs are constructed manually based on domain knowledge, although automatic generation is possible).

The random variables corresponding to the nodes in PSG are as follows: M denotes the number of nodes in PSG. Notice that M can be either a constant or a variable taking values from a discrete sets $\{1, \ldots, m_{\max}\}$, where $m_{\max}$ is the maximal number of nodes allowed (usually defined by the biologists). $W_i = \{p_i, q_i, s_i\}$ is the label for the i^{th} node, where p_i, q_i, s_i are the starting position, ending positions and the state assignment in the sequence, which completely determine the node according to its semantics defined in the PSG. Under this setup, a value instantiation of $W = \{M, \{W_i\}\}$ defines a unique segmentation and annotation of the observed protein sequence $\mathbf{x}$. A probabilistic distribution on a protein structural graph can be postulated using the potential functions defined on the *cliques* of nodes induced by the edges in the graph (Hammersley & Clifford, 1971). The conditional probability of W given the observation $\mathbf{x}$ is defined as

$$P(W|\mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}_G} \exp(\sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, W_c)), \qquad (7.1)$$

where f_k is the k^{th} feature defined over the cliques c, such as the secondary structure assignment or the segment length. Note that $\mathcal{C}_G$ can be a huge set, and each W_c can also include a large number of nodes due to various levels of dependencies. Designing features for such cliques is non-trivial because one has to consider all the joint configurations of all the nodes in a clique.

Usually, the spatial ordering of most protein folds is known *a priori*, which leads to a deterministic state dependency between W_i and W_{i+1}. This leads to a simplification of the "effective" clique sets (those need to be parameterized) and the relevant feature design. As a result, only pairs of segment-specific cliques that are coupled needs to be considered (e.g., those connected by the undirected "red" arc in Figure 7.1, which leads to the following formulation:

$$P(W|\mathbf{x}) = \frac{1}{Z}\prod_{i=1}^{M}\exp(\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, W_i, W_{\pi_i})), \qquad (7.2)$$

where W_{π_i} denotes the spatial predecessor (i.e., with small position index) of W_i determined by a "long-range interaction arc". Technically, neighboring nodes must satisfy the constraints on the location indexes, i.e. $q_{i-1}+1 = p_i$. We omit it here for presentation clarity.

7.2.2 Efficient Inferences via Belief Propagation

Similar to CRFs, we estimate the parameters λ_k by minimizing the regularized negative loss:

$$R_\Phi(\lambda) = \sum_{i=1}^{M}\sum_{k=1}^{K}\lambda_k f_k(\mathbf{x}, w_i, w_{\pi_i}) - \log Z + \frac{\eta\|\lambda\|^2}{2}.$$

To perform the optimization, we need to seek the zero of the first derivative, i.e.

$$\frac{\partial R}{\partial \lambda_k} = \sum_{i=1}^{M}(f_k(\mathbf{x}, w_i, w_{\pi_i}) - \mathrm{E}_{P(W|x)}[f_k(\mathbf{x}, W_i, W_{\pi_i})]) + \eta\lambda_k, \qquad (7.3)$$

where $\mathrm{E}_{P(s|\mathbf{x})}[f_k(\mathbf{x}, W_i, W_{\pi_i})]$ is the expectation of feature $f_k(\mathbf{x}, W_i, W_{\pi_i})$ over the model. The convexity property guarantees that the root corresponds to the optimal solution. Since there is no closed-form solution to (7.3), iterative searching algorithms have to be applied.

Similar to CRFs, we still have an efficient inference algorithm as long as the graphs do not have crossing edges. We redefine the forward probability $\alpha_{<l,y_l>}(r, y_r)$ as the conditional probability that a segment of state y_r ends at position r given the observation $x_{l+1}\ldots x_r$ and a segment of state y_l ends at position l. Let "→" be the operator to get the predecessor state and "←" for successor state (the value is known if the state transition is deterministic). The recursive step can be written as:

$$\alpha_{<l,y_l>}(r, y_r) = \sum_{p,\, p',\, q'}\alpha_{<l,y_l>}(q', y')\alpha_{<q',y>}(p-1, \overleftarrow{y_r})\exp(\sum_{k}\lambda_k f_k(\mathbf{x}, w, w_\pi)),$$

where w is the ending segment from position p to r with state y_r and w_π is the spatial predecessor segment determined by a "long-range interaction arc" from p' to q' with state y'. The range over the summation is

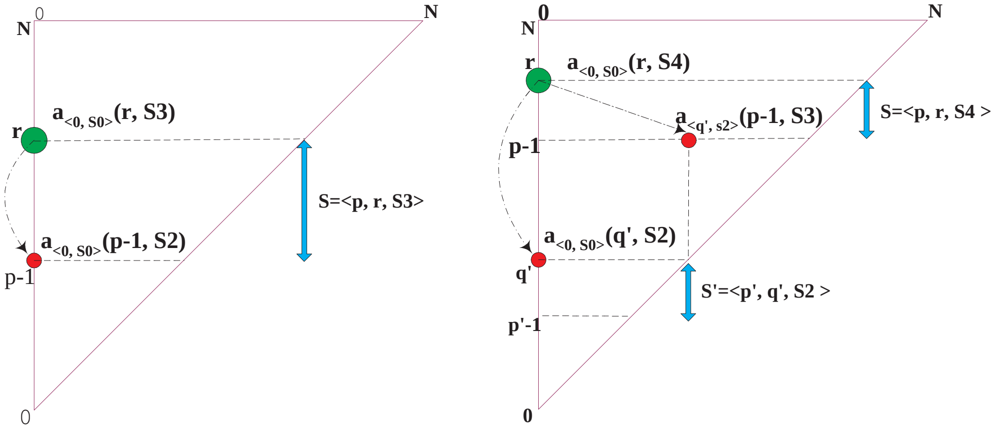


Figure 7.2: An example of forward algorithm for the graph defined in Figure 7.1 (B). x/y-axis: index of starting/end residue position; green circle: target value; red circle: intermediate value. (Left) calculation for $\alpha_{<0,S_0>}(r, S3)$ for segment S_3 with no direct forward neighbor; (right) calculation for $\alpha_{<0,S_0>}(r, S4)$ for segment S_4 with direct forward neighbor S_2

$\sum_{p=r-\ell_1+1}^{r-\ell_2+1} \sum_{q'=l+\ell'_1-1}^{p-1} \sum_{p'=l}^{q'-\ell'_1+1}$, where $\ell_1 = \max \text{length}(y)$, $\ell_2 = \min \text{length}(y)$. Then the normalizer $Z = \alpha_{<0,y_{\texttt{start}}>}(N, y_{\texttt{end}})$. Figure 7.2 shows a toy example on how to calculate the forward probability.

Similarly, we can define the backward probability $\beta_{<r,y_r>}(l, y_l)$ as the probability that a segment of state y_l ends at l given $x_{l+1} \ldots x_r$ and a segment of state y_r ends at r. Then we have

$$\beta_{<r,y_r>}(l, y_l) = \sum_{q',\, p,\, q} \beta_{<r,y_r>}(p{-}1, \overleftarrow{y})\beta_{<p'-1,\overleftarrow{y}>}(q', \overrightarrow{y_l}) \exp(\sum_k \lambda_k f_k(\mathbf{x}, w, w_\pi),$$

where w_π is the starting segment from l+1 to q' with state $\overrightarrow{y_l}$ and w is the spatial successor segment from p to q at state y. Given the backward and forward algorithm, we can compute the expectation of each feature f_k in (7.3) accordingly. For a test sequence, we search for the segmentation that maximizes the conditional likelihood $P(W|x)$. Define $\delta_{<l,y_l>}(r, y_r)$ as the best score over all possible segmentation configurations of $x_{l+1} \ldots x_r$ that ends at state y_r, then we have

$$\delta_{<l,y_l>}(r, y_r) = \max_{p,\, p',\, q'} \delta_{<l,y_l>}(q', y')\delta_{<q',y>}(p-1, \overleftarrow{y_r}) \exp(\sum_k \lambda_k f_k(\mathbf{x}, w, w_\pi)).$$

The best segmentation can be traced back from $\delta_{<0,y_{\texttt{start}}>}(N, y_{\texttt{end}})$, where N is the number of residues in the sequence.

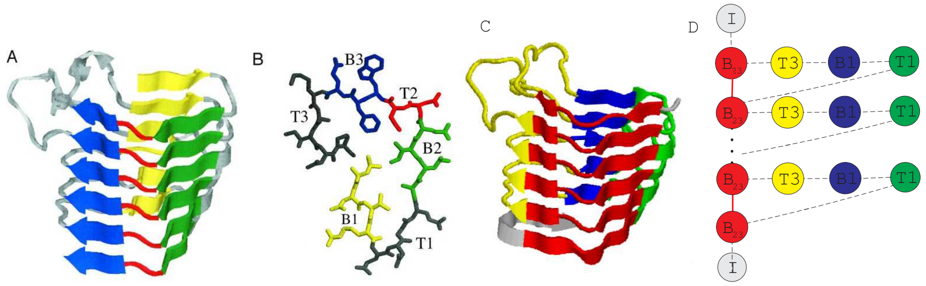


Figure 7.3: 3-D structures and side-chain patterns of β-helices; (A) Side view (B) top view of one rung (C) Segmentation of 3-D structures (D) protein structural graph. E1 = {black edge} and E2 = {red edge} (Figure (A) and (B) are adapted from (Bradley et al., 2001))

In general, the computational cost of SCRFs for the forward-backward and Viterbi algorithm will be polynomial to the length of the sequence N. In most real applications of protein fold prediction, we can define the graph so that the number of possible residues in each node is much smaller than N or fixed. Therefore the final complexity can be reduced to approximately $O(N^2)$.

7.2.3 Experiment Results

Protein structural graph for β-helix fold Currently there exist 14 protein sequences with β-helix whose crystal structures have been known. Those proteins belong to 9 different SCOP families (Murzin et al., 1995) (see Table 7.1). Computationally, it is very difficult to detect the β-helix fold because proteins with this fold share less than 25% similarity in sequence identity, which is the "twilight zone" for sequence-based methods, such as PSI-BLAST or HMMs. Traditional methods for protein family classification, such as threading, PSI-BLAST and HMMs, fail to solve the β-helix recognition problem across different families (Bradley et al., 2001). Recently, a computational method called BetaWrap, has been proposed to predict the β-helix specifically (Bradley et al., 2001). The algorithm "wraps" the unknown sequences in all plausible ways and check the scores to see if any wrap makes sense. The cross-validation results in the protein data bank (PDB) seem promising. However, the BetaWrap algorithm might suffer from hand-coding many biological heuristic rules so that it is prone to over-fit the known β-helix proteins and hard to generalize for other prediction tasks.

From previous literature on β-helix, there are two facts important for accurate prediction: 1) the β-strands of each rung have patterns of pleating and hydrogen bonding that are well conserved across the superfamily; 2) the interaction of the strand side-chains in the buried core are critical determinants of the fold (Yoder & Jurnak, 1995; Kreisberg et al., 2000). Therefore we define the protein structural graph of β-helix as in Figure 7.3 (D).

There are 5 states in the graph altogether, i.e. s-B23, s-T3, s-B1, s-T1 and s-I. The state s-B23 is a union of B2, T2 and B3 because these three segments are all highly conserved in pleating patterns and a combination of conserved evidence is generally much easier to detect. We fix the length of S-B23 and S-B1 as 8 and 3 respectively for two reasons: first, these are the number of residues shared by all known β-helices; second, it helps to limit the search space and reduce the computational costs. The states s-T3 and s-T1 are used to connect s-B23 and s-B1. It is known that the β-helix structures will break if the insertion is too long. Therefore we set the length of s-T3 and s-T1 so that it varies from 1 to 80. s-I is the non-β-helix state, which refers to all those regions outside the β-helix structures. The red edge between s-B23 is used to model the long-range interaction between adjacent β-strand pairs. For a protein without any β-helix structures, we define the protein structural graph as a single node of state s-I.

To determine whether a protein sequence has the β-helix fold, we define the score ρ as the log ratio of the probability of the best segmentation to the probability of the whole sequence as one state s-I, i.e. $\rho = \log \frac{\max_s P(S|x)}{P(<1,N,s-I>|x)}$. The higher the score ρ, the more likely that the sequence has a β-helix fold. We did not explicitly model the long-range interactions between B1 strands since the effect is relatively weak given only 3 residues in s-B1 segments while adding it in makes the graph much more complicated. However, we do use the B1 interactions as a filter in Viterbi algorithm: specifically, $\delta_t(y)$ will be the highest value whose corresponding segmentation also have alignment scores for B1 higher than some threshold set using cross-validation.

Feature extraction SCRFs provide an expressive framework to handle long-range interactions for protein fold prediction. However, the choice of feature function f_k plays a key role in accurate predictions. We define two types of features for β-helix prediction, i.e. *node features* and *pairwise features.*

Node features cover the properties of an individual segment, including: **a)** Regular expression template: Based on the side-chain alternating patterns in B23 region, BetaWrap generates a regular expression template to detect β-helices, i.e. $\Phi X \Phi X X \Psi X \Phi X$, where Φ matches any of the hydropho-

bic residues as {A, F, I, L, M, V, W, Y}, Ψ matches any amino acids except ionisable residues as {D, E, R, K} and X matches any amino acid (Bradley et al., 2001). Following similar idea, we define the feature function $f_{RST}(x, S)$ equal to 1 if the segment S matches the template, and 0 otherwise.
b) Probabilistic HMM profiles: The regular expression template as above is straightforward and easy to implement. However, sometimes it is hard to make a clear distinction between a true motif and a false alarm. Therefore we built a probabilistic motif profile using HMMER (Durbin et al., 1998) for the s-B23 and s-B1 segments respectively. We define the feature function $f_{HMM1}(x, S)$ and $f_{HMM2}(x, s)$ as the alignment scores of S against the s-B23 and s-B1 profiles.
c) Secondary structure prediction scores: Secondary structures reveal significant information on how a protein folds in three dimension. The state-of-art prediction method can achieve an average accuracy of 76 - 78% on soluble proteins. We can get fairly good prediction on alpha-helix and coils, which can help us locate the s-T1 and s-T3 segments. Therefore we define the feature function $f_{ssH}(x, S)$, $f_{ssE}(x, S)$ and $f_{ssC}(x, S)$ as the average of the predicted scores over all residues in segment S, for helix, sheet and coil respectively by PSIPRED (Jones, 1999).
d) Segment length: It is interesting to notice that the β-helix structure has strong preferences for insertions within certain length ranges. To consider this preference in the model, we did parametric density estimation. Several common functions are explored, including Poisson distribution, negative-binomial distribution and asymmetric exponential distribution, which consists for two exponential functions meeting at one point. We use the latter one since it provides a better estimator than the other two. Then we define the feature function $f_{L1}(x, S)$ and $f_{L3}(x, S)$ as the estimated probability of the length of segment S as s-T1 and s-T3 respectively.

Pairwise features capture long-range interactions between adjacent β-strand pairs, including:
a) Side chain alignment scores: BetaWrap calculates the alignment scores of residue pairs depending on whether the side chains are buried or exposed. In this method, the conditional probability that a residue of type X will align with residue Y, given their orientation relative to the core, is estimated from a β-structure database developed from the whole PDB (Bradley et al., 2001). Following similar idea, we define the feature function $f_{SAS}(x, S, S')$ as the weighted sum of the side chain alignment scores for S given S' if both are s-B23 segments, where a weight of 1 is given to inward pairs and 0.5 to

the outward pairs.
b) Parallel β-sheet alignment scores: In addition to the side chain position, another aspect is to study the different preferences for parallel and anti-parallel β-sheets. Steward & Thornton derived the "pairwise information values" (V) for a residue of type X given the residue Y on the pairing parallel (or anti-parallel) strand and the offsets of Y from the paired residue Y' of X (Steward & Thornton, 2002). The alignment score for two segments $x = X_1 \ldots X_m$ and $y = Y_1 \ldots Y_m$ is defined as

$$score(x, y) = \sum_i \sum_j \{(V(X_i|Y_j, i-j) + V(Y_i|X_j, i-j))\}.$$

Compared with the side chain alignment scores, this score also takes into account the effect of neighboring residues on the paired strand. We define the feature function $f_{PAS}(x, S, S') = score(S, S')$ if both S and S' are s-B23 and 0 otherwise.
c) Distance between adjacent s-B23 segments: There are also different preferences for the distance between adjacent s-B23 segments. It is difficult to get an good estimation of this distribution since the range is too large. Therefore we simply define the feature function as the normalized length, i.e. $f_{DIS}(x, S, S') = \frac{dis(S,S')-\mu}{\sigma}$, where μ is the mean and σ^2 is the variance.

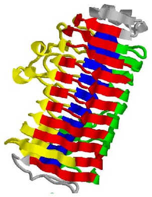
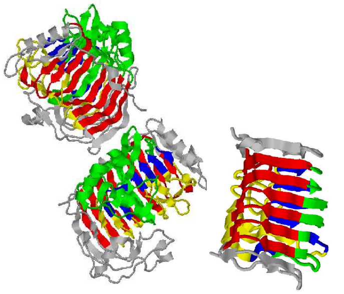
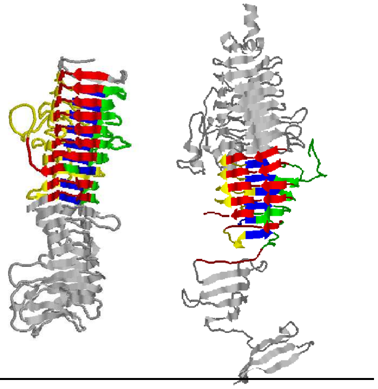
It is interesting to notice that some features defined above are quite general, not limited to predicting β-helices only. For example, an important aspect to discriminate a specific protein fold with others is to build HMM profiles or identify regular expression templates for conserved regions if they exist; the secondary structure assignments are essential in locating the elements within a protein fold; if some segments have strong preferences for certain length range, then the lengths are also informative. For pairwise features, the β-sheet alignment scores are useful for folds in β-family while hydrophobicity is important for α- or $\alpha\beta$-family.

Experiment results We followed the experiment setup described in (Bradley et al., 2001): a PDB-minus dataset was constructed from the PDB protein sequences (July 2004 version) (Berman et al., 2000) with less than 25% similarity to each other and no less than 40 residues in length. Then the β-helix proteins are removed from the dataset, resulting in 2094 sequences in total. The proteins in PDB-minus dataset will serve as negative examples in the cross-family validation and discovery of new β-helix proteins. Since negative data dominate the training set, we subsample 15 negative sequences that are most similar to the positive examples in sequence identity so that SCRFs can learn a better decision boundary than randomly sampling.

Table 7.1: Scores and rank for the known right-handed β-helices by HMMER, BetaWrap and SCRFs. 1: the scores and rank from BetaWrap are taken from [3] except 1ktw and 1ea0; 2: the bit scores in HMMER are not directly comparable

SCOP family	PDB-id	Struct-based	HMMs	Seq-based	HMMs	BetaWrap[1]		SCRFs	
		Bit score[2]	Rank	Bit score[2]	Rank	Score	Rank	ρ-score	Rank
P.69 pertactin	1dab	-73.6	3	-163.4	75	-17.84	1	10.17	1
Chondroitinase B	1dbg	-64.6	5	- 171.0	55	-19.55	1	13.15	1
Glutamate synthase	1ea0	-85.7	65	-109.1	72	-24.87	N/A	6.21	1
Pectin methylesterase	1qjv	-72.8	11	-123.3	146	-20.74	1	6.12	1
P22 tailspike	1tyu	-78.8	30	-154.7	15	-20.46	1	6.71	1
Iota-carrageenase	1ktw	-81.9	17	- 173.3	121	-23.4	N/A	8.07	1
Pectate lyase	1air	-37.1	2	-133.6	35	-16.02	1	16.64	1
	1bn8	180.3	1	-133.7	37	-18.42	3	13.28	2
	1ee6	-170.8	852	-219.4	880	-16.44	2	10.84	3
Pectin lyase	1idj	-78.1	14	-178.1	257	-17.99	2	15.01	2
	1qcx	-83.5	28	-181.2	263	-17.09	1	16.43	1
Galacturonase	1bhe	-91.5	18	-183.4	108	-18.80	1	20.11	3
	1czf	-98.4	43	-188.1	130	-19.32	2	40.37	1
	1rmg	-78.3	3	-212.2	270	-20.12	3	23.93	2

Table 7.2: Groups of segmentation results for the known right-handed β-helix

Group	Perfect match	Good match	OK match
Missing rungs	0	1-2	3 or more
PDB-ID	**1czf**	1air, 1bhe, 1bn8, 1dbg, **1ee6**(right), 1idj, **1ktw**(left), 1qcx, 1qjv, 1rmg	**1dab**(left), 1ea0, **1tyu**(right)
			

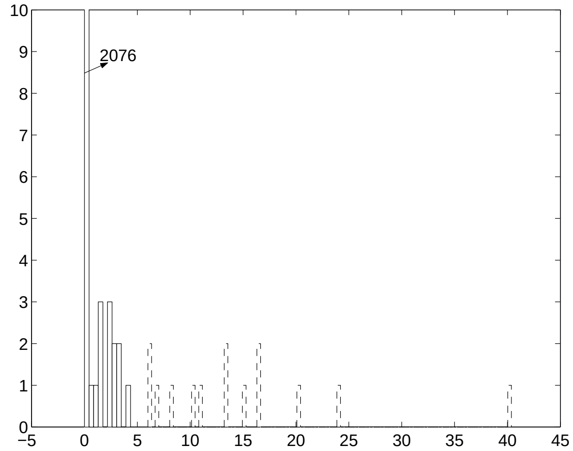


Figure 7.4: Histograms of protein scores of known β-helix proteins against PDB-minus dataset. Blue bar: PDB-minus dataset; green bar: known β-helix proteins. 2076 out of 2094 protein sequences in PDB-minus have a log ratio score ρ of 0, which means that the best segmentation is a single segment in non-β-helix state

A leave-family-out cross-validation was performed on the nine β-helix families of closely related proteins in the SCOP database (Murzin et al., 1995). For each cross, proteins in the one β-helix family are placed in the test set while the remainder are placed in the training set as positive examples. Similarly, the PDB-minus was also randomly partitioned into nine subsets, one of which are placed in the test set while the rest serve as the negative training examples. We compare our results with BetaWrap, a state-of-art algorithm for predicting β-helices, and HMMER, a general motif detection algorithm based on a simple graphical model, i.e. HMMs. The input to HMMER is a multiple sequence alignment. The best multiple alignments are typically generated using 3-D structural information, although this is not strictly sequence-based method. Therefore we generated two kinds of alignments for comparison: one is the multiple structural alignments using CE-MC (Guda et al., 2004), the other is purely sequence-based alignments by CLUSTALW(Thompson et al., 1994).

Table 7.1 shows the output scores by different methods and the relative rank for the β-helix proteins in the cross-family validation. From the results, we can see that the SCRFs model can successfully score all known β-helices higher than non β-helices in PDB. On the other hand, there are two proteins (i.e. 1ktw and 1ea0) in our validation sets that are crystallized recently and thus are not included in the BetaWrap system. We test these two sequences on BetaWrap and get a score of -23.4 for 1ktw and -24.87 for 1ea0. These values are significantly lower than the scores of other β-helices and some of the non β-helix proteins, which indicates that the BetaWrap might be overtrained. As expected, HMMER did worse than SCRFs and BetaWrap even using the structural alignments.

Figure 7.4 plots the score histogram for known β-helix sequences against the PDB-minus dataset. Compared with the histograms in similar experiment by BetaWrap (Bradley et al., 2001), our log ratio score ρ indicates a clearer separation of β-helix proteins v.s. non β-helix proteins. Only 18 out of 2094 proteins has a score higher than 0. Among these 18 proteins, 13 proteins belong to the β-class and 5 proteins belong to the alpha-beta class in CATH database (Orengo et al., 1997). In Table 7.2 we also cluster the proteins into three different groups according to the segmentation results and show examples of the predicted segmentation in each group. From the results, we can see our algorithm demonstrates success in locating each rung in the known β-helix proteins, in addition to predicting membership of β-helix motif.

7.3 Thesis work: Chain Graph Model for Predicting Protein Fold with Structural Repeats

Our experiment results demonstrate that SCRFs are an effective model for general protein fold recognition. However, the computational cost for the forward-backward probabilities and the Viterbi algorithm in SCRFs is at least $O(N^2)$ with an averaged size of N at around 500. This complexity is acceptable for small-scale applications, but is prohibitively expensive for an iterative search algorithm with thousands of iterations. In addition, it will increase (exponentially) with the size of the cliques. When the dependencies between the labels of immediately adjacent segments are not deterministic, for example β-sandwiches or β-trefoils, larger cliques will be induced and thus make SCRFs infeasible for genome-wide applications.

To alleviate the problem, we focus on a special class of the complex protein folds — those with structural repeats, such as the β-helices or the leucine rich repeats (LLR) (Figure 7.5). They are defined as repetitive secondary or supersecondary structural units or *motifs*, such as α-helices, β-strands, β-sheets (colored regions is Fig 7.2), connected by *insertions* of variable lengths, which are mostly short loops and sometimes α-helices or/and β-sheets (gray regions in Fig 7.5). These folds are believed to be prevalent

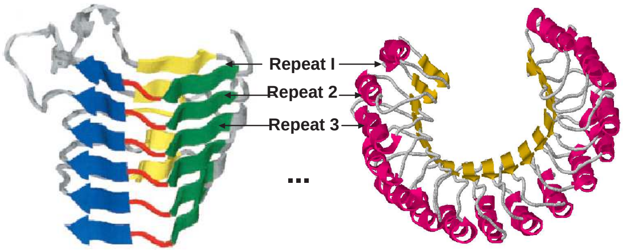


Figure 7.5: Typical 3-D structure of proteins with β-helices (left) and leucine-rich repeats (right). In β-helices, there are three strands: B1 (green), B2 (blue) and B3 (yellow) and the conserved T2 turn (red). In LLR, there is one strand (yellow) and insertions with helices (red).

in proteins and can involve in a wide spectrum of cellular and biochemical activities, such as the initiation of bacterial infection and various protein-protein interaction processes (Yoder et al., 1993; Kobe & Deisenhofer, 1994).

The major challenges in computationally predicting these folds include the long-range interactions between their structural repeats due to unknown number of spacers (i.e., amino acid insertions), low sequence similarities between recurring structural repeats within the same protein and also across multiple proteins, and poor conservation of the insertions across different proteins. Therefore it is desirable to devise a model that contains some sequence motif modules reflecting structural conservation, and at the same time considers the long-range interactions between such structural elements of each repeat (as captured in the SCRF model) and even higher-order dependencies between recurring repeats. Note that a naive SCRFs formalism would be prohibitively expensive due to such higher-order dependencies across repeats, and it also lacks the device to incorporate sequence motifs. Here we propose a chain graph model that makes use of both the undirected SCRFs and the directed sequence motif models as building blocks, and integrates them via a directed network, which captures dependencies between structural repeats without computing a global normalizer required in a naive SCRF formalism.

7.3.1 Chain Graph Model

A *chain graph* is a graph consisting of both directed and undirected arcs associated with probabilistic semantics. It leads to a probabilistic distribution bearing properties of both the Markov random fields (i.e., allowing potential-

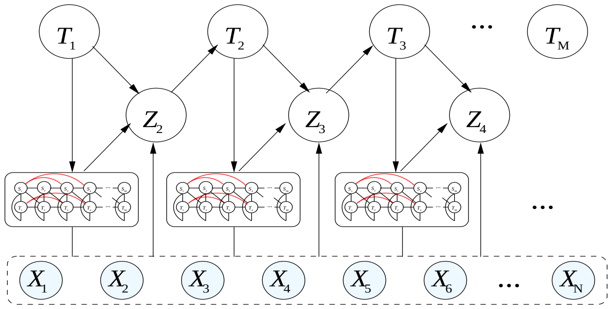


Figure 7.6: The chain graph model for protein folds with structural repeats. The directed edges denote conditional dependencies of the child node on the parental nodes. Note that each of the round-cornered boxes represents a repeat-specific component as SCRFs. An edge from the box denote dependencies on the joint configuration of all nodes within the box.

based local marginals that encode constraints rather than causal dependencies) and the Bayesian networks (i.e., not having a hard-to-compute global partition function for normalization and allowing causal integration of subgraphs that can be either directed or undirected) (Lauritzen & Wermuth, 1989). A chain graph can be represented as a hierarchical combination of conditional networks. Formally, a chain graph over the variable set $\mathbf{V}$ that forms multiple subgraphs $\mathcal{U}$ can be represented by the following factored form: $P(\mathbf{V}) = \prod_{u \in \mathcal{U}} P(u|\text{parents}(u))$, where parents$(u)$ denotes the union of the parents of every variable in u. $P(u|parents(u))$ can be defined as a conditional directed or undirected graph (Buntine, 1995), which needs to be locally normalized only.

In the protein structure graph, we define an *envelop*, as a subgraph that corresponds to one repeat containing both motifs and insertions or the regions outside the protein fold (which we term *null regions*). It can be viewed as a mega node in a chain graph defined over the entire protein sequence and its segmentation. Let M denote the number of envelops in the sequence, $\mathbf{T} = \{T_1, \ldots, T_M\}$ where $T_i \in \{\text{repeat, null region}\}$ denote the structural label of the i^{th} envelop. Recall that the detailed configuration of one repeat or null region can be modeled by a plain SCRF model, therefore we define each envelop as a single SCRF and let $W_{(i)}$ denote all the hidden nodes in envelop i, i.e. $W_{(i)} = \{M_{(i)}, \mathbf{Y}_{(i)}\}$. Following the notational convention in the previous section, we use $W_{(i),j}$ to represent a segment-specific clique within envelop i that completely determines the configuration of the j^{th} segment in the i^{th} envelop. To define a *hierarchical segmentation* of a protein sequence, our chain graph employs a directed graph on the top layer to determine the labels of the envelops and then models the conditional segmentation of the envelops by an undirected SCRFs model. Putting everything together, we arrive at a chain graph depicted in Figure 7.6.

Given a sequence $\mathbf{x}$, the node value initiation of $W = \{M, \{W_{(i)}\}, \mathbf{T}\}$ in the chain graph G defines a hierarchical segmentation of the sequence as follows:

$$P(W|\mathbf{x}) = P(M, \{W_{(i)}\}, \mathbf{T}|\mathbf{x}) = P(M)\prod_{i=1}^{M} P(T_i|\mathbf{x}, T_{i-1}, W_{(i-1)})P(W_{(i)}|\mathbf{x}, T_i, T_{i-1}, W_{(i-1)}). \tag{7.4}$$

$P(M)$ is the prior distribution of the number of repeats in one protein and for simplicity a uniform prior is assumed. $P(T_i|\mathbf{x}, T_{i-1}, W_{(i-1)})$ is the state transition probability and we use the structural motif as an indicator for the existence of a new repeat, i.e.:

$$P(T_i|\mathbf{x}, T_{i-1}, W_{(i-1)}) = \sum_{Q_i=0,1} P(T_i|Q_i)P(Q_i|\mathbf{x}, T_{i-1}, W_{(i-1)}), \tag{7.5}$$

where Q_i is a random variable denoting whether there exists a motif in the i^{th} envelop and $P(Q_i|\mathbf{x}, T_{i-1}, W_{(i-1)})$ is computed using a profile mixture model described in Section 7.3.2. For the third term, we define the conditional probability using SCRFs, i.e.

$$P(W_{(i)}|\mathbf{x}, T_i, T_{i-1}, W_{(i-1)}) = \frac{1}{Z_{(i)}} \exp(\sum_{j=1}^{M_{(i)}} \sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, W_{(i),j}, W_{\pi_{(i),j}})), \tag{7.6}$$

where $Z_{(i)}$ is the local normalizer over all the configurations of $W_{(i)}$, and $W_{\pi_{(i),j}}$ is the spatial predecessor of $W_{(i),j}$ defined by long-range interactions. Similar to SCRFs, the parameters λ can be estimated using the regularized negative log-loss,

$$R_\Phi(\lambda) = \sum_{i=1}^{M}[\sum_{j=1}^{M_{(i)}} \sum_{k=1}^{K} \lambda_k f_k(\mathbf{x}, w_{(i),j}, w_{\pi_{(i),j}}) - \log Z_{(i)}] + \frac{\eta\|\lambda\|^2}{2}, \tag{7.7}$$

where the last term is a Gaussian prior over the parameters as a smoothing term. To perform the optimization, we need to compute the first derivative and set it to zero, i.e.

$$\frac{\partial R}{\partial \lambda_k} = \sum_{i=1}^{M} \sum_{j=1}^{M_{(i)}} \{f_k(\mathbf{x}, w_{(i),j}, w_{\pi_{(i),j}}) - \mathrm{E}_{P(W_{(i)}|\mathbf{x})}[f_k(\mathbf{x}, W_{(i),j}, w_{\pi_{(i),j}})]\} + \eta\lambda_k, \tag{7.8}$$

where $\mathrm{E}_{P(W_{(i)}|\mathbf{x})}[f_k(\mathbf{x}, W_{(i),j}, w_{\pi_{(i),j}})]$ is the expectation of feature $f_k(\mathbf{x}, W_{(i),j}, w_{\pi_{(i),j}})$ over all possible segmentation assignment of the i^{th} envelop.

Given a test sequence, we need to find the segmentation with the highest conditional probability. One naive way is to compute the probability for all possible segmentations, which is computationally too expensive. To solve the problem, we use a greedy search algorithm: define $\delta(r, t)$ as the highest score that the last envelop has state t given the observation $x_1 x_2 \ldots x_r$, and $\varphi(r, t) = \{m, \mathbf{y}\}$ is the corresponding "argmax" segmentation of envelop i. Then the recursive step is

$$\delta(r, t) = \max_{r', t', w} \delta(r', t') P(T = t|\mathbf{x}, t', \varphi(r', t')) P(W = w|\mathbf{x}, t, t', \varphi(r', t')). \tag{7.9}$$

To summarize, using a chain graph model, we can effectively identify motifs based on their structural conservation and at the same time take into account the long-range interactions between repeat units. In addition, a chain graph also reduces the computational costs by using local normalization. Since the side-chain interactions take effect only within a small range in 3-D space, our model can be seen as a reasonable approximation for a global optimal model. For most protein folds, where the number of possible residues in motif or insertions is much smaller than N or fixed, the complexity of our algorithm can be bounded by $O(N)$.

7.3.2 Mixture Profile Model for Structural Motif Detection

A commonly adopted representation for motif-finding is the position weight matrix (PWM), which records the relative frequency (or a related score) of each amino acid type at the positions of a motif (Bailey & Elkan, 1994). Statistically, a PWM defines a product of multinomial model for the observed instances of a motif, which assumes that the positions within the motif are independent of each other.

One important observation about the repetitive structural motif is that motif instances close to each other in 3-D are more similar than the instances from distance locations or on different sequences due to the side-chain interaction constraints. In addition, for motifs in the β-class, the positions with the side-chain pointing to the core are more conserved than the ones pointing outward. To capture these properties of structural motifs, a mixture profile model is applied. Given a multi-alignment of the structural motif from the i-th protein, $A_i = A_{i1} A_{i2} \ldots A_{iH}$ where H is the length of the motif, we assume that it is generated from a mixture of a motif model shared by all the proteins ($\theta^{(1)}$) and a sequence specific background model

($\theta_i^{(0)}$). Let $\theta^{(1)}$ parameterizes a product of multinomial models and $\theta_i^{(0)}$ be a simple multinomial vector. Suppose there exist position-specific features of the motif $\mathbf{f_d}$, for example the side-chain pointing directions (inward or outward) for each position, we define hidden variables $\mathbf{Q} = \{Q_{ij}\}$, for which $Q_{ij} = 1$ means that the j^{th} position in the i^{th} protein is generated by model $\theta^{(1)}$ and $Q_{ij} = 0$ means that it is from model $\theta_i^{(0)}$. We assume the prior distribution of Q_{ij} is Bernoulli with parameter τ. Using the EM algorithm, we can estimate τ, $\theta^{(1)}$ and $\theta_i^{(0)}$. To calculate $P(Q_i|\mathbf{x}, T_{i-1}, W_{(i-1)})$, we do an online updating of $\theta^{(0)}$ and τ using the motif defined by $W_{(i-1)}$.

7.3.3 Experiment Results

In our experiments, we test our algorithm on two important protein folds in β-class, including the right-handed β-helices and the leucine-rich repeats.

We followed the setup described in Section 7.2.3: a PDB-minus dataset was constructed from the PDB protein sequences and a leave-family-out cross-validation was performed. Since the ratio of negative examples to positive examples is very large, we subsample only 15 negative sequences that are most similar to the positive examples in sequence identity in order to find a better decision boundary than randomly sampling. Two types of features are defined: one is *node feature*, which covers the properties of an individual segment, including pattern matching templates and HMM profiles for conserved motifs, secondary structure prediction scores from PSIPRED (Jones, 1999) and the segment length; the other is *pairwise feature*, which captures the long-range interactions between adjacent β-strand pairs, including alignment scores of residue pairs in terms of the buried or exposed side chains (Bradley et al., 2001) and preferences for parallel or anti-parallel β-sheets (Steward & Thornton, 2002) (see Section 7.2.3 for detail).

To determine whether a protein sequence has a particular fold, we define the score ρ as the normalized log ratio of the probability for the best segmentation to the probability of the whole sequence in a null state (non-β-helix or non-LLR). We compare our results with BetaWrap, the state-of-art algorithm for predicting β-helices, THREADER, a threading algorithm and HMMER, a general motif detection algorithm using HMMs. The input to HMMER can be the structural alignments using CE-MC (Guda et al., 2004) or purely sequence-based alignments by CLUSTALW(Thompson et al., 1994).

β-helices fold Table 7.3 shows the output scores by different methods and the relative rank for the β-helix proteins in the cross-family validation.

Table 7.3: Scores and rank for the known right-handed β-helices by HMMER, Threader, BetaWrap, SCRFs and chain graph model(CGM). 1: the scores and rank from BetaWrap are taken from [3] except 1ktw and 1ea0; The result of sequence-based HMMs is shown in Section 7.2.3

SCOP Family	PDB-ID	Struct-based HMMs		Threader	BetaWrap[1]		SCRFs		CGM	
		Bit score	Rank	Rank	Wrap-score	Rank	ρ-score	Rank	ρ-score	Rank
P.69 pertactin	1DAB	-73.6	3	24	-17.84	1	10.17	1	31.69	1
Chondroitinase B	1DBG	-64.6	5	47	-19.55	1	13.15	1	34.89	1
Glutamate synthase	1EA0	-85.7	65	N/A	-24.87	N/A	6.21	1	29.04	1
Pectin methylesterase	1QJV	-72.8	11	266	-20.74	1	6.12	1	22.69	1
P22 tailspike	1TYU	-78.8	30	2	-20.46	1	6.71	1	20.59	1
Iota-carrageenase	1KTW	-81.9	17	10	-23.4	N/A	8.07	1	16.06	1
Pectate lyase	1AIR	-37.1	2	45	-16.02	1	16.64	1	22.87	2
	1BN8	180.3	1	76	-18.42	3	13.28	2	28.98	1
	1EE6	-170.8	852	228	-16.44	2	10.84	3	15.16	3
Pectin lyase	1IDj	-78.1	14	6	-17.99	2	15.01	2	17.50	2
	1QCX	-83.5	28	6	-17.09	1	16.43	1	20.67	1
Galacturonase	1BHE	-91.5	18	18	-18.80	1	20.11	3	28.98	1
	1CZF	-98.4	43	5	-19.32	2	40.37	1	24.68	3
	1RMG	-78.3	3	27	-20.12	3	23.93	2	27.37	2

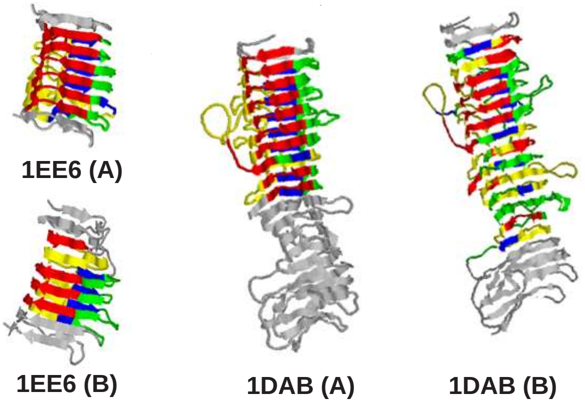


Figure 7.7: Segmentation for protein 1EE6 and 1DAB by SCRFs(A) and chain graph model (B). Red: B2-T2-B3 motif; blue: B1 motif; green and yellow: insertions.

From the results, we can see that the both SCRFs and chain graph model can successfully score all known β-helices higher than non β-helices in PDB, significantly better than Threader, HMMER and BetaWrap, the stat-of-art method for predicting the β-helices fold.

Our algorithm also demonstrates success in locating each repeat in the known β-helix proteins. Fig.7.7 shows the segmentation results for 1EE6 and 1DAB respectively. From the results, we can see: for 1EE6 SCRFs can locate two more repeats accurately than the chain graph model; however, our model is able to span the repeats over the whole area of the true fold for 1DAB while SCRFs can only locate part of them. We can see that there are strength and weakness for both methods in terms of segmentation results. On the other hand, since the computational complexity for chain graph model is only $O(N)$, the real running time of our model (approx. 2.5h) is more than 50 times faster than that of SCRFs (approximately 140h). Therefore the chain graph model achieves a good approximation to SCRF with much less training time.

leucine-rich repeats Based on the conservation level, we define the *motif* for LLR as the β-strand and short loops on two sides, resulting 14 residues in total. The length of the *insertions* varies from 6 to 29. There are 41 LLR proteins with known structure in PDB, covering 2 super-families and 11 families in SCOP. The LLR fold is relatively easy to detect due to its conserved motif with many leucines in the sequence and short insertions. Therefore it would be more interesting to discover new LLR proteins with

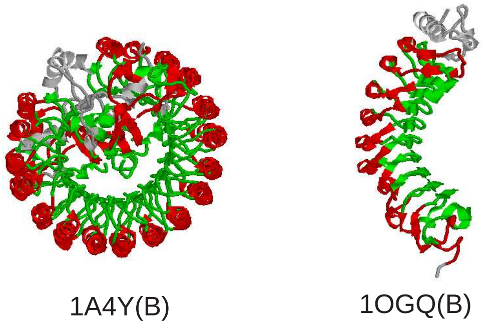


Figure 7.8: Segmentation for protein 1OGQ and 1A4Y by chain graph model. Green: motif; red: insertions.

much less sequence identity to previous known proteins. We select one protein in each family as representative and see if our model can identify LLR proteins across families.

Table 7.4 lists the output scores by different methods and the rank for the LLR proteins. In general, LLR is easier to identify than the β-helices. Again, the chain graph model performs much better than other methods by ranking all LLR proteins higher than non-LLR proteins. In addition, the predicted segmentation by our model is close to prefect match for most LLR proteins (some examples are shown in Figure 7.8).

7.4 Summary

In this section, we propose the segmentation conditional random fields for general protein fold recognition and a chain graph model to detect the protein folds with structural repeats specifically. Both methods demonstrate successes in the protein fold (or motif) recognition in our experiments, which confirmed our hypothesis of applying conditional graphical models for protein structure prediction. In addition, they are one of the first probabilistic models that explicitly consider the long-range interactions in predicting protein super-secondary structures from sequences.

The chain graph model, as a localized version of SCRFs, solves the problem of huge computational costs and achieves good approximation to the original SCRFs. Although the current model is developed for a special kind of protein folds, its divide-and-conquer idea under the chain graph framework can be derived for other complex proteins accordingly.

Table 7.4: Scores and rank for the known right-handed Leucine-rich repeats (LLR) by HMMER, Threader and chain graph model (CGM). For CGM, ρ-score = 0 for all non-LLR proteins.

SCOP Family	PDB-ID	ClustalW+HMMs		Struct-based HMMs		Threader	CGM	
		Bit score	Rank	Bit Score	Rank	Rank	ρ-score	Rank
28-residue LRR	1A4Y	-125.5	4	-76.7	1	457	127.8	1
Rna1p (RanGAP1)	1YRG	-95.4	1	-81.1	1	181	64.3	1
Cyclin A/CDK2-associated p19	1FQV	-163.3	89	-111.4	10	398	77.1	1
Internalin LRR domain	1O6V	-62.8	1	-0.7	1	306	116.5	1
Leucine rich effector	1JL5	-86.7	1	-26.5	1	46	187.5	1
Ngr ectodomain-like	1P9A	-120.0	9	-68.6	1	16	105.0	1
Polygalacturonase inhibiting protein	1OGQ	-155.0	32	-18.2	1	284	66.4	1
Rab geranylgeranyltransferase alpha-subunit	1DCE	-145.4	16	-59.7	1	35	17.4	1
mRNA export factor	1KOH	-153.9	42	-91.7	1	177	37.1	1
U2A'-like	1A9N	- 280.9	861	-151.4	478	62	55.1	1
L domain	1IGR	-150.0	46	-107.1	249	67	8.2	1

Chapter 8

Quaternary Structure Prediction

In previous chapters, we study the tasks of protein secondary structure prediction and tertiary fold recognition. These tasks are both important and difficult, which undoubtedly attracts extensive studies by many researchers in different domains as reviewed in Chapter 2. However, the study of protein quaternary structures, which consist of *multiple* protein chains that form chemical bonds among the side chains of sequence-distant residues to reach a structurally stable domain [1], have been left far behind: on one hand, the current understanding of quaternary structures are quite limited due to the difficulty of resolving the structures of the large complexes. On the other hand, these structures play very important roles in protein functions, some examples include enzymes, hemoglobin, DNA polymerase, and ion channels. They also contribute significantly to evolutionary stability in that the changes of the quaternary structures can occur through each individual chain or through the reorientation relative to each other. Most importantly, recent studies in virus proteins indicate the common existence of quaternary structures in viruses, such as adenovirus and reovirus, as well as HIV-protease. Furthermore, a deeper knowledge about how the protein folds into quaternary structures will inevitably help uncover the complicated folding processes in nature.

Quaternary structures are stabilized mainly by the same non-covalent interactions as tertiary structures, such as hydrogen bonding, van der Walls interactions and ionic bonding. Unfortunately, previous work on fold recog-

[1] In comparison, the stable three-dimension structure held by a single protein is called the tertiary structure.

nition for single chains is not directly applicable because the complexity is greatly increased both biologically and computationally, when moving to quaternary multi-chain structures. First, the averaged size of the quaternary fold is much larger than that of single proteins, which makes it difficult for lab experiments to resolve their structures. As a result, there are only one or two positive examples with structure annotation for most quaternary folds. The unavailability of training data will render useless many machine learning approaches. From an evolutionary point of view, the functional sites on the complexes are more apt to change in order to adapt to the environment (especially true for virus proteins), while the general structural skeleton remains stable. Reflected in the protein sequences, we observe that a large number of proteins share the same fold without sequence similarity, which violates the assumptions of homology (sequence similarity-based) methods. On the other hand, threading algorithms based on physical forces rely strictly on the estimation of free-energies. To find the best conformation, we need to consider the conformation of all the protein chains jointly since every chain contributes to the stability of the structures. Given the enormous search spaces in quaternary structures, it is difficult to find an accurate estimate of the energies, not mention problems posed by the abundant local optima for computational solutions.

Motivated by its biological importance and corresponding computational challenges, we develop the linked SCRF model, another extension of the generalized conditional graphical model, for protein quaternary fold recognition. The major advantage of our model is the use of discriminative objective functions, which make it easy to incorporate any biological features, instead of the free-energy functions with particular assumptions on physical forces and requiring complex free-energy minimization methods. It provides the feasibility to capture the long-range dependencies of different subunits within one chain and between chains under one model gracefully. In addition, efficient approximation algorithms we used are able to find optimal or near-optimal solutions, which can be directly transferred to the free-energy minimization settings.

8.1 Materials and Evaluation Measure

In this section, we give a brief overview of current work in protein quaternary structure prediction, then introduce the protein quaternary fold recognition tasks and evaluation measures.

8.1.1 Protein Quaternary Structures

The *quaternary structure* is the stable association of multiple polypeptide chains via non-covalent bonds, resulting in a stable unit. To date, there has been significant progress in protein tertiary fold recognition and alignment, ranging from sequence similarity matching (Altschul et al., 1997; Durbin et al., 1998), to threading algorithms based on physical forces (Jones et al., 1992) and to machine learning methods (Cheng & Baldi, 2006; Ding & Dubchak, 2000). However, few studies have addressed the problem of predicting *quaternary structures*.

Recent pursuit of computational methods to determine the quaternary structures can summarized in three research directions. One direction is the simple classification problem: given a protein primary sequence, whether it takes a tertiary structure of a single chain or a quaternary structures with other proteins. Most work along this direction focuses on examining the sequence evolution information in terms of PSI-BLAST profiles or different propensities of amino acids in these two structure types (Garian, 2001; Zhang et al., 2003; Chou & Cai, 2003). Then the information is used as input features for a classifier, such as support vector machines or naive Bayes. The overall prediction accuracy is around 60-70%. The second direction is the study of domain-domain docking or interaction type in the protein complexes (Kim & Ison, 2005; Chen & Zhou, 2005). In this approach, the docking or interaction type are examined based on the protein structures deposited in the PDB. The methodology is generalizing the association mechanisms of multiple proteins in the complexes to the quaternary structures in general. It is observed that the overall prediction success rate across the genome-wide study is poor. However, the performance can be improved significantly if only those proteins that have informative (or related) proteins in the training set are consider. The third direction seeks the geometric regularities and constraints to reduce the huge searching spaces of quaternary structures (Inbar et al., 2005).

8.1.2 Quaternary Fold Recognition

In this chapter of the thesis, we are trying to solve the problem of quaternary fold recognition. Specifically, our task starts with a target fold F that the biologists are interested in. There are no constraints about F except that it has to be a quaternary fold, with multiple number of participating protein chains (either different or identical). Then all the proteins with resolved structures deposited in the PDB can be classified into two groups, i.e. those take the target fold F and those not. These proteins together with the labels can be used as training data. Our goal is to predict whether a testing protein, without resolved structures, takes the fold F in nature or not; if they do, locate the starting and ending positions of the subsequence that adopts the fold.

It can be seen that our task involves two sub-tasks: one is the classification problem, that is, given a set of training sequences $X_1, X_2, \ldots, X_N$ and their labels $y_1, y_2, \ldots, y_N$ $(y_i = 0, 1)$, predict the label of a new testing sequence X_{new}; the other subtask is not that straightforward to describe in mathematical settings. We can think of the target fold as some patterns (or motifs in bioinformatics terminology). Given a set of instances of the pattern, including both the positive examples (subsequences with the pattern F) and the negative examples (sequences without the pattern F), we want to predict whether the pattern appears in any subsequence of the testing proteins. It is easy to answer the questions in the first subtask if we can solve the second one successfully, which is our focus in the rest of the chapter. A key problem in the second task is how we can represent the descriptive patterns (or motifs) using mathematical notations. The linked segmentation conditional random fields, as described in the next section, makes very natural use of the graphical model representations and successfully solve the problem.

After introducing the define of our task, we want to stress again its strong biological motivation and wide applications. This task falls within the general blueprint of previous studies in quaternary structures, but differs in two aspects: first, the problem comes directly from the needs of biologists in their experiments or studies. In our case, the collaborators that we worked with have been studying a particular fold for a long time. The positive proteins with resolved structures are quite limited, although they believe it is a common fold in nature. By identifying more examples in the unresolved proteins in sequence databases, such as Swiss-Prot or UniProt, we can help the biologists to verify their hypothesis about the fold. Second, the problem we are trying to address is much more difficult than the common fold classification because we do not have as many positive examples and they do not share high sequence similarities. In other words, the patterns that we are trying to identify have not been represented clearly in the training data. This is the main motivation why we want to develop a relatively complex model, rather than a simple classifier. Notice that our models can be used in the traditional fold recognition or threading setting, however, its advantage can be demonstrated best in cases for predicting those difficult protein folds.

8.1.3 Evaluation Measures

Our goal is to identify the possible positive proteins from the whole collection of protein sequences without resolved structures. It is similar to the information retrieval tasks, where given some key words (or patterns described in words) we want to retrieve the documents that contain similar contents as the key words. Therefore our evaluation measure is to see if we can rank the positive proteins higher than the negative ones in cross-validation.

To construct negative examples in the training set, we build the PDB-minus dataset as described in the previous chapter. It consists of all PDB protein sequences (July 2006 version) (Berman et al., 2000) with less than 25% similarity to each other and no less than 40 residues in length, resulting in 2810 chains with 430927 residues. Since we aim to search proteins sharing similar structures without sequence similarity, a leave-family-out cross-validation was performed to avoid overfitting. For each cross, positive proteins from the same protein family are placed in the test set while the remainder are placed in the training set. Similarly, the PDB-minus was also randomly partitioned into subsets, one of which are placed in the test set while the rest serve as the negative training examples.

To demonstrate the effectiveness of different recognition models, we choose the triple β-spirals and double-barrel trimer as examples in our experiments.

The triple β-Spiral fold is a processive homotrimer consisting of three identical interacting protein chains. It was first identified by Mark J. van Raaij and collaborators in 1999 (van Raaij et al., 1999). The fold serves as a fibrous connector from the main virus capsid to a C-terminal knob that binds to host cell-surface receptor proteins (see Figure 8.3). Up to now there are three crystallized structures with this fold deposited in the Protein Data Bank (PDB) (Berman et al., 2000), one is the adenovirus protein (DNA virus, PDB ID: 1qiu), another is reovirus (RNA virus, PDB ID: 1kke) and the other is PRD1 (PDB ID: 1yq8). The common existence in both DNA and RNA viruses reveals important evolution relationships in the viral proteins, which also indicates that the triple beta-spirals might be a common fold in nature. The detailed description of the TBS fold can be found in Appendix B.3.

The double-barrel trimer is a potential protein fold, which has been found in the coat proteins from several kinds of viruses. It consist of two eight-stranded jelly rolls, or β-barrels. As seen in Figure 8.5, the component

β-strands are labeled as B, C, D, E, F, G, H and I respectively. The first strand is named as B because one example of the β-barrels, the tomato bushy stunt virus, has an extra initial strand. The fold has been found in the major coat proteins of bacteriophage PRD1, that of human adenovirus, Paramecium bursaria chlorella virus (PBCV) and archaeal virus STIV. This amazing phenomenon raised the unexpected possibility that viruses infecting different kinds of species are related by evolution. It has been suggested that the occurrence of a double-barrel trimer is common all icosahedral dsDNA viruses with large facets, irrespective of its host, and furthermore an indicator of common ancestor in a lineage of viruses (Benson et al., 2004). The detailed description of the double-barrel trimer can be found in Appendix B.4.

8.2 Thesis work: Linked Segmentation Conditional Random Fields

In the previous section, we have identified the key issues for quaternary fold recognition, that is, how to represent the patterns exhibited by the fold using mathematical models. In structural biology, the conventional representation of a protein fold is the use of a graph (Westhead et al., 1999), in which nodes represent the secondary structural components and the edges indicate the inter- and intra-chain interactions between the components in their 3-D structures. This intuitive representation motivates us to use graphical models, which is an elegant combination of graph theory and probability theory. Specifically we base the work on SCRFs for single-chained (tertiary) fold recognition problems. Its successful applications to the β-helixes and leucine-rich repeats (LLR) encourages us to pursue similar directions for quaternary fold (or motif) recognition, albeit requiring fundamental changes: representing and inferencing over multiple cross-chain bonds, and resolving a graphical structure of much greater complexity, which demands entirely new estimation methods. Therefore we propose the linked segmentation conditional random fields.

Before covering the algorithm in detail, we first review the protein structural graph described in previous chapters. Given a protein fold, a structural graph is defined as $G =< \mathcal{V}, \mathcal{E} >$, where $\mathcal{V} = \mathcal{U} \bigcup \{\mathcal{I}\}$, $\mathcal{U}$ is the set of nodes corresponding to the secondary structure elements within the fold and I is the node to represent the elements outside the fold. $\mathcal{E}$ is the set of edges between neighboring elements in primary sequences or edges indicating the potential long-range interactions between elements in tertiary structures.

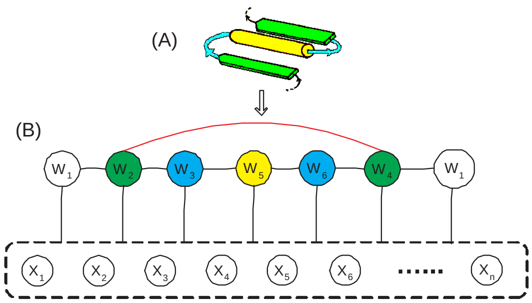


Figure 8.1: (A) 3-D structure of β-α-β motif (B) PSG of β-α-β motif. Node: Green=β-strand, yellow=α-helix, cyan=coil, white=non-β-α-β (I-node)

Figure 8.1 shows an example of the structural graph for β-α-β motif. The PSG for a quaternary fold can be derived similarly: first construct a PSG for each component protein or a component monomeric PSG for homo-multimers, and then add edges between the nodes from different chains if there are chemical bonds, forming a more complex but similarly-structured quaternary PSG.

Given a structural graph G defined on one chain and a protein sequence $\mathbf{x} = x_1 x_2 \ldots x_N$, we can have a possible segmentation of the sequence, i.e. $\mathbf{y} = \{M, \mathbf{w}\}$, where M is the number of segments and $\mathbf{w}_j = \{s_j, p_j, q_j\}$, in which s_j, p_j and q_j are the state, starting position and ending position index of the j^{th} segment, The conditional probability of a segmentation $\mathbf{y}$ given the observation $\mathbf{x}$ can be computed as follows:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_0} \prod_{c \in \mathcal{C}^G} \exp(\sum_k \lambda_k f_k(\mathbf{x}_c, \mathbf{y}_c)),$$

where Z_0 is the normalization factor based on all possible configurations. The graphical model representation is shown in Figure 8.1-(B).

More general, given a quaternary structure graph G with C chains, i.e. $\{\mathbf{x}_i | i = 1 \ldots C\}$, we have a segmentation initiation of each chain $\mathbf{y}_i = (M_i, \mathbf{w}_i)$ defined by the protein structural graph, where M_i is the number of segments in the i^{th} chain, and $\mathbf{w}_{i,j} = (s_{i,j}, p_{i,j}, q_{i,j})$, $s_{i,j}$, $p_{i,j}$ and $q_{i,j}$ are the state, starting position and ending position of the j^{th} segment. Following similar idea as the CRFs model, we have

$$P(\mathbf{y}_1, \ldots, \mathbf{y}_C | \mathbf{x}_1, \ldots, \mathbf{x}_C) = \frac{1}{Z} \prod_{\mathcal{C} \in G} \Phi(\mathbf{y}_{\mathcal{C}}, \mathbf{x}), \tag{8.1}$$

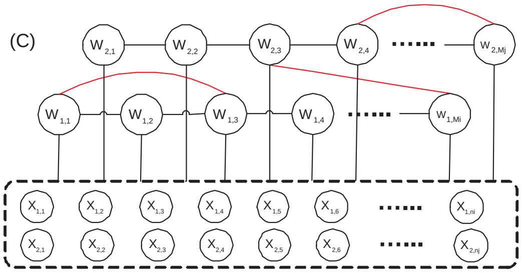


Figure 8.2: Graphical Model Representation of l-SCRFs model with multiple chains. Notice that there are long-range interactions (represented by red edges) within a chain and between chains

where Z is the normalizer over all possible segmentation configures on all the sequences (see Figure 8.2 for its graphical model representation.)

We decompose the potential function over the cliques $\Phi(\mathbf{y}_{\mathcal{C}}, \mathbf{x})$ as a product of unary and pairwise potentials, i.e.

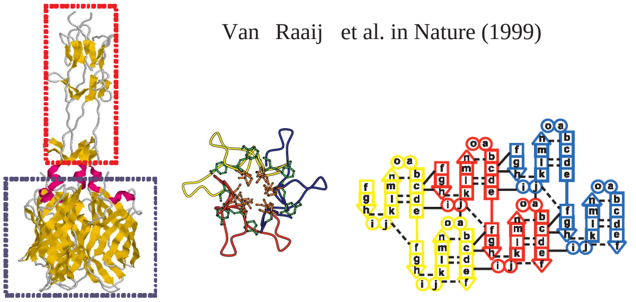
$$
\begin{aligned}
&P(\mathbf{y}_1, \ldots, \mathbf{y}_C | \mathbf{x}_1, \ldots, \mathbf{x}_C) \\
&\quad = \frac{1}{Z} \prod_{\mathbf{w}_{i,j} \in \mathcal{V}_G} \Phi(\mathbf{x}_i, \mathbf{w}_{i,j}) \prod_{\langle \mathbf{w}_{a,p}, \mathbf{w}_{b,q} \rangle \in \mathcal{E}_G} \Phi(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}_{a,p}, \mathbf{w}_{b,q}) \\
&\quad = \frac{1}{Z} \exp\Big(\sum_{\mathbf{w}_{i,j} \in \mathcal{V}_G} \sum_{k=1}^{K1} \theta_{1,k} f_k(\mathbf{x}_i, \mathbf{w}_{i,j}) + \sum_{\langle \mathbf{w}_{a,p}, \mathbf{w}_{b,q} \rangle \in \mathcal{E}_G} \sum_{k=1}^{K2} \theta_{2,k} g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}_{a,p}, \mathbf{w}_{b,q})\Big),
\end{aligned}
$$

where f_k and g_k are features, $\theta_{1,k}$ and $\theta_{2,k}$ are the corresponding weights for the features. Specifically, we factorize the features as the following way,

$$
\begin{aligned}
&f_k(\mathbf{x}_i, \mathbf{w}_{i,j}) = f'_k(\mathbf{x}_i, p_{i,j}, q_{i,j}) \delta(\mathbf{w}_{i,j}) \\
&\quad = \begin{cases} f'_k(\mathbf{x}_i, p_{i,j}, q_{i,j}) & \text{if } (s_{i,j} = s) \& (q_{i,j} - p_{i,j} \in \text{length range}(s)) \\ 0 & \text{otherwise,} \end{cases}
\end{aligned}
$$

$$
\begin{aligned}
&g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{w}_{a,u}, \mathbf{w}_{b,v}) \\
&= \begin{cases} g'_k(\mathbf{x}_a, \mathbf{x}_b, p_{a,u}, q_{a,u}, p_{b,v}, q_{b,v}) & \\ & \text{if } (s_{a,u} = s) \& (s_{b,v} = s'), \\ & q_{a,u} - p_{a,u} \in \text{length range } (s), \\ & q_{b,v} - p_{b,v} \in \text{length range } (s') \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}
$$

Given the definition of the protein structure graph, our next question is how to automatically build the graph for a particular fold. The answer



Van Raaij et al. in Nature (1999)

Figure 8.3: Demonstration graph of triple β-spirals. (left) 3-D structures view. Red block: shaft region (target fold), black block: knob region. (middle) top view. (right) maps of hydrogen bonds within a chain and between chains.

depends on the type of protein folds of concern and how much knowledge we can bring to bear. If it is a fold that biologists have studied over the years and accumulated some basic knowledge of their properties (for example β-α-β motif), the topology of this graph can be constructed easily by communicating with the experts. If it is a fold whose structure is totally new to the biologists, we can follow a general procedure with the following steps: first, construct a multiple structure alignment of all the positive proteins (among themselves); second, segment the alignment into disjoint parts based on the secondary structures of the majority proteins; third, draw a graph with nodes denoting the resulting secondary structure elements and then add edges between neighboring nodes. Finally, add the long-range interaction edge between two nodes if the average distance between all the involved residues is below some threshold $\kappa^{\min}$ specified by the user. We skip detailed discussion of the latter case as it is a separate line of research and assume that we are given a reasonably good graph over which we perform our learning. Below are two examples of how to construct the graphs given some prior knowledge of the target folds.

8.2.1 L-SCRFs for Triple-β Spirals

To provide a better protein structural graph for the linked SCRFs model, we notice the following structural characteristics in the triple β-spirals: the fold consists of the three identical protein chains with a series of repeated structural elements (see Figure 8.3). Each of these structural elements is composed of: 1. a β-strand that runs parallel to the fiber axis 2. a long

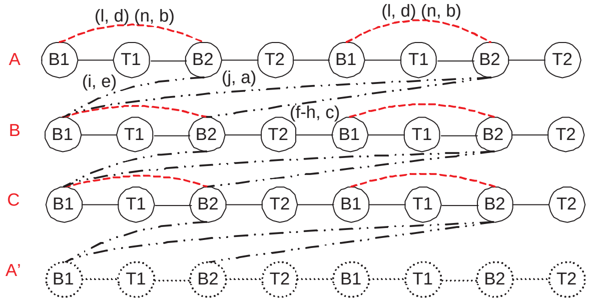


Figure 8.4: Protein Structural Graph of the Triple β-spirals. Chain C' is a mirror of chain C for better visual effects. Dotted line: inter-chain interactions; solid line: intra-chain interactions. The pairs of characters on the edge indicate the hydrogen bonding between the residues denoted by the characters.

solvent-exposed loop of variable lengths, 3. a second β-strand that forms antiparallel β-sheets with the first one, and slightly skewed to the fiber axis, 4. successive structural elements along the same chain are connected together by a tight β-turn (Scanlon, 2004; Weigele et al., 2003). Among those four components, the two β-strands are quite conserved in sequences and Green et al. characterize them by labeling each position using character ‘a’ to ‘o’. Specifically, i-o for the first strand and a-h for the second strand (see Figure 8.3).

Based on the discussion above, we define the protein structural graph of the triple β-spirals as in Figure 8.4. There are 5 states in the graph altogether, i.e. B1, T1, B2 and T2, which correspond to the four components of each repeated structural element, and the null state I, which refers to the non-triple β-spiral region. We fix the length of B1 and B2 as 7 and 8 respectively due to their sequence conservation. In addition, we set the length of T1 and T2 in the range of $[0, 15]$ and $[0, 8]$ individually since longer insertions will bring instability to the structures. It is easy to notice that the transitions between different states are deterministic as long as the number of rungs (repeated structural elements) is given. The pairs of interaction residues are marked on the edges, which will be used to define the pairwise features in section 8.4.

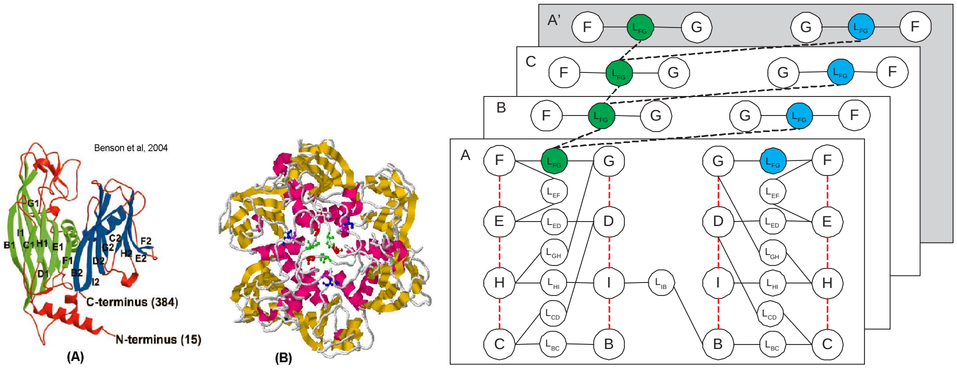


Figure 8.5: (Left) (A) 3-D structure of the coat protein in bacteriophage PRD1 (PDB id: 1CJD). (B) 3-D structure of PRD1 in trimers with the inter-chain and intra-chain interactions in the FG loop. Color notation: In FG1, green: residue #133, red: residue #135, purple: residue #142; In FG2, blue: residue #335. (Right) PSG of double-barrel trimer. The within-chain interacting pairs are shown in red dash line, and the inter-chain ones are shown in black dash line. Green node: FG1; Blue node: FG2.

8.2.2 L-SCRF for Double-barrel Trimer

For the double-barrel trimer fold, it is not straightforward, or even seemingly impossible, to uncover the structural conservation through sequences since there are only four positive proteins and none of them share sequence similarities. There are some general descriptive observations we can make: (1) the lengths of the eight β-strands varies, ranging from 4 to 16 residues, but the layout of the strands is fixed. The separation (insertions) between the strands is relatively short (4- 10 residues), however, it is interesting to notice some exceptions, for example the long insertions between the F and G strand (20 - 202 residues); another long loops between D-E strand (9 - 250 residues); the short β-turn between E and F. (2) The chemical bonds that stabilize the trimers are located between the FG loops. However, the bonding type and specific locations remain unclear, which poses a major challenge. We denote the FG loop in the first double-barrel trimer as FG1, and that in the second one as FG2.

Based on the discussion above, we define the protein structural graph of the double-barrel trimer as shown in Figure 8.5. There are 17 states in the graph altogether, i.e. B, C, D, E, F, G, H, I as the eight β-strands in the β-barrels, l_{BC}, l_{CD}, l_{DE}, l_{EF}, l_{FG}, l_{GH}, l_{HI}, l_{IB} as the loops between the β-strands. The length of the beta-strands are in the range of $[3, 16]$. The range of the loops l_{BC}, l_{CD}, and l_{EF} are $[4, 10]$; that of l_{DE} and l_{FG} are

$[10, 250]$; that of l_{GH}, l_{HI}, l_{IB} are $[1, 30]$. The within-chain interacting pairs are shown in red dash line, and the inter-chain interacting pairs are shown in black dash line.

8.3 Efficient Inference and Learning

The feature weights $\{\theta_{1,k}\}$ and $\{\theta_{2,k}\}$ are the model parameters. In the training phase, we estimate their values by maximizing the regularized joint conditional probability of the training data, i.e

$$\{\hat{\theta}_1, \hat{\theta}_2\} = \text{argmax} \sum_{n=1}^{N} \log P(\mathbf{y}_1^{(n)}, .., \mathbf{y}_C^{(n)} | \mathbf{x}_1^{(n)}, .., \mathbf{x}_C^{(n)}) + \frac{\|\theta_1\|^2}{2\sigma_1^2} + \frac{\|\theta_2\|^2}{2\sigma_2^2}.$$

There is no closed form solution to the equation above, therefore we apply an iterative searching algorithm. Taking the first derivative of the log likelihood $\mathcal{L}(\theta_1, \theta_2)$, we have

$$\frac{\partial \mathcal{L}}{\partial \theta_{1,k}} = \sum_{n=1}^{N} \sum_{\mathbf{y}_{i,j}^{(n)} \in \mathcal{V}_G} (f_k(\mathbf{x}_i^{(n)}, \mathbf{y}_{i,j}^{(n)} - E_{P(\mathbf{Y}^{(n)})}[f_k(\mathbf{x}_i^{(n)}, \mathbf{Y}_{i,j}^{(n)}]) + \frac{\theta_{1,k}}{\sigma_1^2}, \quad (8.2)$$

$$\frac{\partial \mathcal{L}}{\partial \theta_{2,k}} = \sum_{n=1}^{N} \sum_{\langle \mathbf{y}_{a,p}, \mathbf{y}_{b,q} \rangle \in \mathcal{E}_G} (g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{y}_{a,p}, \mathbf{y}_{b,q}) - E_{P(\mathbf{Y}^{(n)})}[g_k(\mathbf{x}_a, \mathbf{x}_b, \mathbf{Y}_{a,p}, \mathbf{Y}_{b,q})]) + \frac{\theta_{2,k}}{\sigma_2^2} (8.3)$$

Since PSG is a complex graph with loops and multiple chains, we explored efficient approximation methods to estimate the whole summation terms on the right-hand side of eq (8.2) and eq (8.3), which are referred to $\nabla\theta_{1,k}$ and $\nabla\theta_{1,k}$ respectively later in the paper.

8.3.1 Approximate Inference via Contrastive Divergence

There are three major approximation approaches in graphical models: sampling, variational methods and loopy belief propagation. It is not straightforward to use the latter two due to the semi-Markov property in our L-SCRF model (the labels are assigned to subsequences instead of individual amino acid), and more importantly the unique property of PSG that allows the number of nodes to be a variable (for example, the triple β-spirals have different number of repeats for each example). Sampling techniques have been

```
    ALGORITHM-1: Description of Contrastive Divergence
    Input: $\theta_1$ and $\theta_2$; Output: $\nabla\theta_1$ and $\nabla\theta_2$
1.  Sample a data vector $\mathbf{y}^0$ from the empirical distribution $P^0$;
2.  Iterate over T times:
    Sample a value for each latent variable $\{\mathbf{y}_i = \{M_i, \mathbf{w}_i\}\}$ $(i = 1, \ldots, C)$ from its posterior
    probability defined in eq(8.1). The value is represented as $\hat{\mathbf{y}}^t$.
4.  Calculate the contrastive divergence as $\nabla\theta_1 = E_{\mathbf{y}^0}[f_k] - E_{\hat{\mathbf{y}}}[f_k]$,
    $\nabla\theta_2 = E_{\mathbf{y}^0}[g_k] - E_{\hat{\mathbf{y}}}[g_k]$.
```

widely used in the statistics community, however, there are two main problems, i.e. inefficiency due to the long "burn-in" periods and large variance in the final estimation. To avoid the problem, we use contrastive divergence (CD) proposed in (Welling & Hinton, 2002). It is similar to Gibbs sampling, except that, instead of running Gibbs sampling until the equilibrium distribution is reached, it runs the sampler up to only a few iterations and uses the resulting distribution to approximate the true model distribution. The algorithm is described in ALGORITHM-1.

Notice that there is a problem if we use the naive Gibbs sampling in step (2) since the variables $\mathbf{y}_i = \{M_i, \mathbf{w}_i\}$ may be of different dimensions in each sampling iteration, depending on the value of M_i (M is a variable if the fold has a variable number of structural repeats, e.g. the TBS fold). We use the reversible jump MCMC algorithm (Green, 1995), which has achieved success in various applications, such as mixture models, hidden Markov models for sequence segmentation and phylogenetic trees.

8.3.2 Reversible Jump Markov Chain Monte Carlo

Given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, our goal is propose a new move $\mathbf{y}_i^*$. To satisfy the detailed balance defined by the MCMC algorithm, auxiliary random variables v and v^* have to be introduced. The definitions for v and v^* should guarantee the *dimension-matching requirement*, i.e. $\dim(y_i) + \dim(v) = \dim(y_i^*) + \dim(v')$ and there is a one-to-one mapping from (y_i, v) to (y_i^*, v'), i.e. there exists a function Ψ so that $\Psi(y_i, v) = (y_i^*, v')$ and $\Psi^{-1}(y_i^*, v') = (y_i, v)$. As a special case, we can add appropriate auxiliary variables v only to the sample spaces with a lower dimension. We define four types of Metropolis operators to construct a Markov chain on the sequence of segmentations:

1. *State switching*: given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, select a segment

j uniformly from $[1, M]$, and a state value s' uniformly from state set $\mathcal{S}$. Set $\mathbf{y}_i^* = \mathbf{y}_i$ except that $s_{i,j}^* = s'$.

2. *Position Switching*: given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, select the segment j uniformly from $[1, M]$ and a position assignment $d' \sim \mathrm{U}[q_{i,j-1}+1, q_{i,j+1} - 1]$. Set $\mathbf{y}_i^* = \mathbf{y}_i$ except that $q_{i,j}^* = d'$.

3. *Segment split*: given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, propose $\mathbf{y}_i^* = (M_i^*, \mathbf{w_i}^*)$ with $M_i^* = M_i + 1$ segments by splitting the j^{th} segment, where j is randomly sampled from $\mathrm{U}[1, M]$. Set $\mathbf{w}_{i,k}^* = \mathbf{w}_{i,k}$ for $k = 1, \ldots, j-1$, and $\mathbf{w}_{i,k+1}^* = \mathbf{w}_{i,k}$ for $k = j+1, \ldots, M_i$. Sample a value assignment of $v \sim P(v)$, compute w_i^*, w_{i+1}^* via $(w_{i,j}^*, w_{i,j+1}^*, v') = \Psi(w_{i,j}, v)$.

4. *Segment merge*: given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, propose $M_i* = M_i - 1$ by merging the j^{th} segment and $j + 1^{th}$ segment, where j is sampled uniformly from $[1, M-1]$. Set $\mathbf{w}_{i,k}^* = \mathbf{w}_{i,k}$ for $k = 1, \ldots, j-1$, and $\mathbf{w}_{i,k-1}^* = \mathbf{w}_{i,k}$ for $k = j+1, \ldots, M_i$. Sample a value assignment of $v' \sim P(v')$, compute $w_{i,j}$ via $(w_{i,j}^*, v) = \Psi^{-1}(w_{i,j}, w_{i,j+1}, v')$.

Then the acceptance rate for the proposed transition from y_i to y_i^* is

$$\min\{1, \text{posterior ratio} \times \text{proposal ratio} \times \text{Jacobian}\} =$$
$$\min\{1, \frac{P(\mathbf{y}_1, .., \mathbf{y}_i^*, .., \mathbf{y}_C | \{\mathbf{x}_i\})}{P(\mathbf{y}_1, .., \mathbf{y}_i, .., \mathbf{y}_C | \{\mathbf{x}_i\})} \frac{P(v')}{P(v)} \left| \frac{\partial(\mathbf{y}_i^*, v')}{\partial(\mathbf{y}_i, v)} \right| \},$$

where the last term is the determinant of the Jacobian matrix.

In general, we have regular arrangement of the secondary structure elements in most protein folds so that the state transitions are deterministic or almost deterministic. Therefore the operator for *state transition* can be removed and *segment split or merge* can be greatly simplified. There might be some cases that the inter-chain or intra-chain interactions are also stochastic. Then two additional operators are necessary, including *segment join* (adding an interaction edge in the protein structure graph) and *segment separate* (deleting an interaction edge in the graph). The detailed steps are similar to *state transition*, and we omit the detailed discussion.

8.3.3 Testing Phase

Given a test example with multiple chains $\{\mathbf{x}_1, \ldots, \mathbf{x}_C\}$, we need to estimate the segmentation that yields the highest conditional likelihood. Similar to the training phase, it is an optimization problem involving search in

Algorithm-2: Reversible Jump MCMC Simulated Annealing

```
Input: initial value of $\mathbf{y}_0$, temperature reduction rate $\beta = 0.5$;
Output: predicted value of $\mathbf{y}$.
1. Set $\hat{\mathbf{y}} = \mathbf{y}_0$.
2. For $t \leftarrow 1$ to $\infty$ do :
   2.1  $T \leftarrow \beta t$. If $T = 0$ return $\hat{\mathbf{y}}$
   2.2  Sample a value $\mathbf{y}^{new}$ using the reversible jump MCMC algorithm as
        described in Section 8.3.2. $\nabla E = P(\mathbf{y}^{new}) - P(\hat{\mathbf{y}})$
   2.3  if $\nabla E > 0$, then set $\hat{\mathbf{y}} = \mathbf{y}^{new}$; otherwise set $\hat{\mathbf{y}} = \mathbf{y}^{new}$ with probability
        $\exp(\nabla E/T)$
3. Return $\hat{\mathbf{y}}$
```

multiple-dimensional space. Since it is computationally prohibitive to search over all possible solutions using traditional optimization methods, simulated annealing with reversible jump MCMC is used. It has been shown theoretically and empirically to converges on the global optimum (Andrieu et al., 2000). See Algorithm-2 for details of the method.

8.3.4 An Example of triple β-spirals

It is straightforward to apply the approximate inference algorithms above for predicting triple β-spiral fold except that there is a slight difference in the reversible jump MCMC algorithm: since the state transitions are deterministic given the number of segments, the *state transition* proposal can be skipped. In addition, there is the concept of "rungs" in the triple-beta spirals, i.e. the four parts of a rung must be generated or deleted at the same time for the fidelity of the structures. Therefore in the proposal *segment split*, we need randomly select a rung (instead of one segment) and split it into two rungs, each of which contains the segment B_1, T_1, B_2, T_2. Similarly, in the proposal *segment merge* we randomly select a rung and merge it with the neighboring rung on the right.

Without the loss of generality, we assume that probability of the number of rungs given the data is uniformly distributed. Therefore, with equal probability, we select one of the moves below:

Position Switching given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, randomly select the segment j uniformly at $[1, M]$ and a position assignment $d' \sim \mathrm{U}[q_{i,j-1} + 1, q_{i,j+1} - 1]$. Set $\mathbf{y}_i^* = \mathbf{y}_i$ except that $q_{i,j}^* = d'$. The acceptance rate for the proposal is:

$$\min\{1, \frac{P(\mathbf{y}_1, .., \mathbf{y}_i^*, .., \mathbf{y}_C | \{\mathbf{x}_i\})}{P(\mathbf{y}_1, .., \mathbf{y}_i, .., \mathbf{y}_C | \{\mathbf{x}_i\})}\}.$$

Segment split given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, propose $\mathbf{y}_i^* = (M_i^*, \mathbf{w_i}^*)$ with $M_i^* = M_i + 4$ segments by splitting the j^{th} rung, where j is randomly sampled from uniform distribution over $[0, \llcorner M/4 \lrcorner + 1]$. Set $\mathbf{w}_{i,k}^* = \mathbf{w}_{i,k}$ for $k = 1, \ldots, 4(j-1)+1$, and $\mathbf{w}_{i,k+4}^* = \mathbf{w}_{i,k}$ for $k = 4(j+1)+2, \ldots, M_i$. Since we constrain the length of segment B_1 and B_2 to be fixed, represented as L_{B_1} and L_{B_2}, the starting position of the following T_1 and T_2 can be inferred easily.

If j = 0, we split the starting null segment $w_{i,0}$ into a shorter null segment and a new rung of triple-beta spirals. To fully determine the position of the new rung, we need to provide the value of two variables l_{B1} and l_{B2}, which are the lengths of segment $T1$ and $T2$ respective. Therefore two auxiliary random variables $v_1 \sim U[0,1]$ and $v_2 \sim U[0,1]$ are introduced. Let $L_0 = q_{i,1} - 1$, that is, the length of segment $w_{i,0}$. Then we define

$$\begin{aligned} &l_1 + l_2 + L_{B1} + L_{B2} = v_1 L_0, \\ &\frac{l_1}{l_2} = \frac{1 - v_2}{v_2}. \end{aligned}$$

Solving the two equations above, we have the transformation function Ψ as

$$\begin{aligned} &l_1 = v_2(v_1 L_0 - L_{B1} - L_{B2}) \\ &l_2 = (1 - v_2)(v_1 L_0 - L_{B_1} - L_{B_2}), \end{aligned}$$

and it is straightforward to compute the Jacobian as $J = L_0(v_1 L_0 - L_{B_1} - LB_2)$. The sampling steps are: we first sample a value of v_1, v_2 from $U[0,1]$, then set

$$\begin{aligned} q_{i,1}^* &= q_{i,1} - (l_1 + l_2 + L_{B1} + L_{B_2}) \\ q_{i,2}^* &= q_{i,1} - (l_1 + l_2 + L_{B_2}) \\ q_{i,3}^* &= q_{i,1} - (l_2 + L_{B_2}) \\ q_{i,4}^* &= q_{i,1} - l_2, \end{aligned}$$

accept the proposal $\mathbf{y}^*$ with the acceptance rate

$$\min\{1, \frac{P(\mathbf{y}_1, .., \mathbf{y}_i^*, .., \mathbf{y}_C | \{\mathbf{x}_i\})}{P(\mathbf{y}_1, .., \mathbf{y}_i, .., \mathbf{y}_C | \{\mathbf{x}_i\})} \times L_0(v_1 L_0 - L_{B_1} - L_{B_2})\}.$$

Similar derivation can be developed easily when $j = \llcorner M/4 \lrcorner + 1$, i.e. the ending null segment is selected.

If $1 \leq j \leq \llcorner M/4 \lrcorner$, we split the j^{th} rung into two rungs. In order to fully determine the position of the two new rungs, we need to provide the value of four variables l_1^*, l_2^*, $l_1^{*\prime}$ $l_2^{*\prime}$, which are the length of segment T_1 and T_2 in the two *new* rungs respectively. Let $l_1 = q_{i,4j+3} - q_{i,4j+2}$, the length of segment T_1 in the *old* rung, and $l_2 = q_{i,4j+5} - q_{i,4j+4}$ segment T_3 in the old rung. We introduce three auxiliary variables v_1, v_2, $v_3 \sim U[0,1]$, and define

$$\begin{gathered}\frac{l_1^*}{l_2^*} = \frac{1-v_1}{v_1},\ \frac{l_1^{*\prime}}{l_2^{*\prime}} = \frac{1-v_2}{v_2},\ \frac{l_2^*/v}{l_2^{*\prime}/w} = \frac{1-v_3}{v_3}, \\ l_1^* + l_2^* + l_1^{*\prime} + l_2^{*\prime} = l_1 + l_2 - L_{B_1} - L_{B_2}. \end{gathered} \tag{8.4}$$

In order to achieve the detailed balance, i.e. $P(\mathbf{y}, \mathbf{y}^*) = P(\mathbf{y}^*, \mathbf{y})$, we need to diverge the current topic a little bit to study its reverse proposal, that is, merging the j^{th} and $j+1^{th}$ rung into one rung. The position of the new rung can be fully determined given the value of l_1, the segment length of T_1, and l_2, the segment length of T_1. We introduce one auxiliary variable $v^* \sim U[0,1]$ and define

$$\frac{l_1}{l_2} = \frac{1-v^*}{v^*},\ \ l_1 + l_2 = l_1^* + l_2^* + l_1^{*\prime} + l_2^{*\prime} + L_{B_1} + L_{B_2}.$$

In this way, we satisfy the dimension matching requirement, i.e. dim $(\{l_1, l_2, v_1, v_2, v_3\})$ = dim $(\{l_1^*, l_2^*, l_1^{*\prime}, l_2^{*\prime}, v^*\})$.

Finally, solving the equations in eq (8.4), we have the transformation function Ψ as

$$\begin{aligned} l_1^* &= (1-v_1)(1-v_3)\nabla L \\ l_2^* &= v_1(1-v_3)\nabla L \\ l_1^{*\prime} &= v_3(1-v_2)\nabla L \\ l_2^{*\prime} &= v_3 v_2 \nabla L \\ v^* &= \frac{l_2}{l_1+l_2}, \end{aligned}$$

where $\nabla L = l_1 + l_2 - L_{B_1} - L_{B_2}$. The Jacobian is $J = v_3(1-v_3)\nabla L^3/(l_1+l_2)$. The sampling steps are: we first sample a value of v_1, v_2, v_3 from $U[0,1]$, then set

$$\begin{aligned} &q^*_{i,4j+2} = q_{i,4j+2},\ q^*_{i,4j+3} = q^*_{i,4j+2} + L_{B_1} \\ &q^*_{i,4j+4} = q_{i,4j+3} + l_1^*,\ q^*_{i,4j+5} = q^*_{i,4j+4} + L_{B_2} \\ &q^*_{i,4j+6} = q_{i,4j+5} + l_2^*,\ q^*_{i,4j+7} = q^*_{i,4j+6} + L_{B_1} \\ &q^*_{i,4j+8} = q_{i,4j+7} + l_1^{*\prime},\ q^*_{i,4j+9} = q^*_{i,4j+4} + L_{B_2}, \end{aligned}$$

accept the proposal $\mathbf{y}^*$ with the acceptance rate

$$\min\{1, \frac{P(\mathbf{y}_1, .., \mathbf{y}_i^*, .., \mathbf{y}_C|\{\mathbf{x}_i\})}{P(\mathbf{y}_1, .., \mathbf{y}_i, .., \mathbf{y}_C|\{\mathbf{x}_i\})} \times v_3(1 - v_3)\nabla L^3/(l_1 + l_2)\}.$$

Segment merge given a segmentation $\mathbf{y}_i = (M_i, \mathbf{w_i})$, propose $M_i* = M_i - 4$ by merging the j^{th} rung and $j+1^{th}$ rung, where j is randomly sampled from U[0, $\llcorner M/4 \lrcorner$]. Set $\mathbf{w}^*_{i,k} = \mathbf{w}_{i,k}$ for $k = 1, \ldots, 4(j-1)+1$, and $\mathbf{w}^*_{i,k-4} = \mathbf{w}_{i,k}$ for $k = 4(j+1)+2, \ldots, M_i$. If $j = 0$ or $j = \llcorner M/4 \lrcorner$, the new segment will become a null segment, otherwise it will be a new triple-beta spiral rung. The detail can be derived easily based on our discussion in the *segment split* proposal.

One might notice that in the description above we did not consider the length constraints of T_1 and T_2, which will affect the sampling space of the variables, such as v and v^*. We intentionally omit the constraints for comprehensiveness. In practice those variables are sampled from a more stringent range. Again, the algorithm described above is only one implementation of the general reversible jump MCMC approaches and there are many other alternatives.

8.4 Feature Extraction

The linked SCRF model provide an expressive framework to capture the structural properties of quaternary folds characterized by both inter-chain and intra-chain interactions. Similar as the SCRF model, link SCRFs retain all the feasibility so that we can use any type of informative features, either overlapping or long-range correlations. Again, the choice of feature function f_k plays an essential role in accurate predictions. Two types of features can be defined, i.e. *node features*, which cover the properties of an individual segment, and *pairwise features*, which tries to model the chemical-bonding between the pairs of segments that are close in three-dimensional spaces.

Another view of the feature space is via *common features*, which are can be shared for all kinds of fold recognition, and *signal features*, which are unique to the target fold and but require domain expertise. Our experiments and studies show that the signal features are usually the most discriminative of the target fold and given higher weights in the learnt model. On the other hand, it is time-consuming to get those signal features: generally it takes years for the biologists to accumulate the required knowledge. Sometimes, the current understanding of the target fold is not enough to summarize any reliable signal patterns, in which case the common features could be a reasonable backup. Table 8.1 summarizes the features we used for predicting the TBS and DBT folds.

8.4.1 Common Features for Quaternary Fold Recognition

In general, the common features of quaternary fold recognition are similar to those for tertiary folds. Some features, such as hydrophobicity and iconic propensity, seem to get higher weights since the quaternary complexes usually form a hydrophobic core. The node features that we use in our prediction include:
Secondary structure prediction scores Secondary structures reveal significant information on how a protein folds in three dimension. Therefore we define the feature function $f_{ssH}(\mathbf{x}, q_i, q_{i+1})$, $f_{ssE}(\mathbf{x}, q_i, q_{i+1})$ and $f_{ssC}(\mathbf{x}, q_i, q_{i+1})$ as the average of the predicted scores by PSIPRED (Jones, 1999) over all residues in the segment, for helix, sheet and coil respectively. Similarly, we also define the feature function using the maximal and minimal scores.
Segment length In most cases, each state has strong preferences to a particular range of lengths. Therefore we define the feature function $f_l(\mathbf{x}, q_i, q_{i+1}) = q_{i+1} - q_i$.
Physicochemical properties Some physicochemical properties of the amino acids might be informative. We use the Kyte-Doolittle hydrophobicity score, solvent accessibility and ionizable scores in our methods. The feature functions can easily be derived accordingly.

The pairwise features we found useful for β-sheet related folds include:
Side chain alignment scores For β-sheets, it is observed that the amino acids have different propensities to form a hydrogen bond depending on whether the side-chains are buried and exposed (Bradley et al., 2002). An alignment score of interacting residue pairs can be computed accordingly. In the methods described in (Bradley et al., 2002), the conditional probability that a residue of type X will align with residue Y, given their orientation relative to the core (buried or exposed), is estimated from a β-structure database developed from the PDB database. The feature function $f_{SAS}^{(\phi,\psi)}(\mathbf{x}_a, \mathbf{x}_b, q_{a,p}, q_{a,p+1}, q_{b,q}, q_{b,q+1})$ can be defined as the weighted side-chain alignment scores for the ϕ^{th} residue in segment (a, p) given the ψ^{th} residue in segment (b, q), where (ϕ, ψ) are the positions of interacting pairs marked in Figure 8.4, and a weight of 1 is given to inward pairs and 0.5 to the outward pairs.

Parallel β-sheet alignment scores In addition to the side chain position, another aspect is the different preferences of each amino acid to form parallel and anti-parallel β-sheets. Steward & Thornton derived the "pairwise information values" (V) for a residue of type X given the residue Y on the pairing parallel (or anti-parallel) strand and the offsets of Y from the paired residue Y' of X (Steward & Thornton, 2002). The alignment score for two segments $x = X_1 \dots X_m$ and $y = Y_1 \dots Y_m$ is defined as

$$score(x, y) = \sum_i \sum_j (V(X_i|Y_j, i - j) + V(Y_i|X_j, i - j)).$$

Notice that this score also takes into account the effect of neighboring residues on the paired strand. We define the feature function

$$f_{PAS}^{(\phi,\psi)}(\mathbf{x}_a, \mathbf{x}_b, q_{a,p}, q_{a,p+1}, q_{b,q}, q_{b,q+1}) = score(x_{q_{a,p}+\phi}, y_{q_{b,q}+\psi}),$$

where (ϕ, ψ) are the positions of interacting pairs marked as in the protein structural graph (for example Figure 8.4 and 8.5).

Distance between interacting pairs Mostly there is a distance constraint between the interacting pairs of states since too long an insertion will collapse the structure stabilities. To enforce such constraints, we define feature function $f_{dis}(\mathbf{x}_a, \mathbf{x}_b, q_{a,p}, q_{a,p+1}, q_{b,q}, q_{b,q+1}) = 1$ if $q_{a,p+1} - q_{a,p}$ falls in some range, and 0 otherwise.

8.4.2 Specific Features for Triple-β Spirals

It is quite hard to predict the triple-β spiral fold given the very limited number of positive examples. Fortunately, there exists some identifiable sequence repeat patterns for both B1 and B2 states, which greatly helps to boost the prediction accuracy. We use the regular expression template and profile hidden Markov model to capture those patterns:

Regular expression template Based on the alternating patterns of conserved hydrophobic core and peripheral patches in the B1 and B2 strands, we define the following regular expression templates: XΥΦXΨXX for B1 strand and XXΦXΦXΨX for B2 strand, where Υ is the conserved tight turn that only matches residues in {P, G, A, F, S, L}, Φ is the hydrophobic core that matches any amino acid in {L, I, M, V, T, S, F, A}, Φ is the peripheral patches which matches any amino acid *except* {C, E, H, P, Q, W}, and X can match any amino acid. We define the feature function $f_{RST}(\mathbf{x}, q_i, q_{i+1})$ equal to 1 if the segment matches the template, and 0 otherwise.

Table 8.1: Feature definition for segment $w_i = \{s_i, p_i, q_i\}$ and $w_j = \{s_j, p_j, q_j\}$. Notation: $\ell = q_i - p_i$, $\Upsilon \in$ {P, G, A, F, S, L}, $\Phi \in$ {L, I, M, V, T, S, F, A}, $\Phi \notin$ {C, E, H, P, Q, W}, X match any amino acid. "=∼" indicates that the string matches the regular expression.

Feature Type		Semantics	Examples
Common Features	Node Features	Max predicted 2$^{\text{nd}}$ structure scores	$\max_{t \in [p_i, p_{i+1}]} \mathrm{P}_{\beta-\text{sheet}}(x_t)$
		Min predicted 2$^{\text{nd}}$ structure scores	$\min_{t \in [p_i, p_{i+1}]} \mathrm{P}_{\beta-\text{sheet}}(x_t)$
		Avg predicted 2$^{\text{nd}}$ structure scores	$\sum_{t \in [p_i, p_{i+1}]} \mathrm{P}_{\beta-\text{sheet}}(x_t)/(p_{i+1} - p_i)$
		segment length	$p_{i+1} - p_i$
		physicochemical properties (hydrophobicity, solvent accessibility, ionizable)	$\sum_{t \in [p_i, p_{i+1}]} S_{\text{ionic}}(x_t)/(p_{i+1} - p_i)$
	Pairwise Features	side-chain alignment scores (buried or exposed (Bradley et al., 2002))	$\sum_{t \in [0, \ell]} I(x_i = \text{buried}) S_B(x_{t+p_i}, x_{t+p_j}) + I(x_i = \text{exposed}) S_E(x_{t+p_i}, x_{t+p_j})$
		parallel/anti-parallel β-sheet alignment score (Steward & Thornton, 2002)	$\sum_{t \in [0, \ell]} S_{\text{parallel}}(x_{t+p_i}, x_{t+p_j})$
Signal Features	TBS fold	B1-strand pattern expression matching	$x_{p_i} \ldots x_{p_{i+1}}$ =∼ XϒΦXΨXX
		B2-strand pattern expression matching	$x_{p_i} \ldots x_{p_{i+1}}$ =∼ XXΦXΦXΨX
		B1 (B2) alignment profile matching	$P_{\text{HMMER-B1}}(x_{p_i} \ldots x_{p_{i+1}})$
	DBT fold	max β-turn score (6 type: I, II, VIII, I', II',VIa, VIb, and IV) (Fuchs & Alix, 2005)	$\max_{t \in [p_i, p_{i+1}]} S_{\text{type I } \beta\text{-turn}}(x_t)$

Probabilistic HMM profiles Sometimes the regular expression template is not preferred since it is hard to make a clear cutoff between a true motif and a false alarm. Therefore profile HMM using probabilistic estimation is a better resort. Initially we used the alignments of all the positive examples for B1 and B2 state, but fail to get reasonable results as expected since the sequence similarity is too low to generalize a good profile. Later we observe that the alignments share more similar patterns in sequence if we separate the alignments into groups based on the type of amino acid on conserved β-turn position, that is, position 'j' in Green's labeling scheme (see Figure 8.3). Therefore we built six HMM profiles (one for each amino acid type at position 'j') using HMMER (Durbin et al., 1998) for B1 and B2 respectively. Then we define the feature functions $f_{HMM}(\mathbf{x}, q_i, q_{i+1})$ as the alignment scores of the segment against those B1 and B2 profiles.

8.4.3 Specific Features for Double-barrel Trimer

The double-barrel trimer is a relatively new protein fold which attracts biologists' attention recently, due to their common existence in the coat proteins of viruses infecting different kinds of species. It is claimed that the layouts of the β-barrels are quite unique to virus proteins, but there is no significant sequence conservation either in the β-strand components or the loops or turns connecting the β-strands. The only interesting observation we made after careful study is this short β-turns between strand E and F. It has strong structural conservations without sequence similarities. Therefore we define β-turn features as follows:

β-turn scores There has been extensive research on how to reliably predict the β-turns in the protein sequence. Up to now, the commonly accepted nomenclature divides the β-turns into six types, i.e. type I, II, VIII, I', II', VIa, VIb, and IV, as defined by Hutchinson and Thornton (Hutchinson & Thornton, 1994). In (Fuchs & Alix, 2005), the propensity scores of different amino acids in those six type of β-turns are calculated. In particular, the experiments show that a weighted propensity score using the PSI-BLAST profile performs much better than using the amino acid type that only appear in the protein sequences. Therefore we define the feature function $f_{\beta-turn}(\mathbf{x}, q_i, q_{i+1})$ as the maximal (and minimal) score of the β-turn propensity of each type over the subsequence q_i to q_{i+1}.

Maximal alignment scores The pairwise features of β-sheet alignment scores are defined similarly as described in Section 8.4.1 except that the lengths of the β-strand pair is not necessarily the same. This causes a problem when we try to compute the alignment score since we do not know the interacting pairs any more. To solve the problem, we compute all possible alignments by shifting the starting position of the longer segment and use the highest alignment scores as the features.

Pointwise alignment scores Another challenges in predicting the double-barrel trimer is the incomplete understanding of the inter-chain interactions. It is suggested that the interactions happen within the FG-loop of the two β-barrels, but the specific location as well as the type of chemical bonding remains unclear. Following the idea of natural selection of hydrogen bonds, we compute all the possible pairs of side-chain interactions, and use the highest score as features. In other words, we try to model the possibility of forming hydrogen between the current pairs of segments.

8.5 Experiments

In the experiments, we test our hypothesis by examining whether the linked SCRFs can score the positive examples higher than the negative ones by using the positive sequences from different protein families in the training set. Here the score is defined as the log ratio of the probability of the best segmentation of the sequences to the probability of the whole sequence as one segment in a null state s-I. Since negative data, the PDB-minus set, dominates the training set, we subsample 10 negative sequences that are most similar to the positive examples in sequence identity so that the model can learn a better decision boundary than randomly sampling.

We compare our results with PSI-BLAST (Altschul et al., 1997), Pfam (Bateman et al., 2004), HMMER (Durbin et al., 1998), Threader (Jones et al., 1992) and RADAR (Heger & Holm, 2000). For PSI-BLAST, we use the positive examples in training set as query and search against the PDB database to see if the testing positive protein in the hit list. The threshold is set as 0.001 with 10 repeated iterations. The results are shown in significant score. Pfam is a large collection of protein multiple sequence alignments and profile hidden Markov models. We use the alignments of the training sequences from Pfam and build a HMM profile. HMMER is a general motif detection algorithm based on hidden markov model. The input to HMMER is a multiple sequence alignment generated by CLUSTALW(Thompson et al., 1994). Since there are sequence repeats in the TBS fold proteins, we also examine the sequence repeat detector Radar, an unsupervised learning algorithm to detect sequence repeats. Therefore the results may not be directly comparable with the rest methods, but it would be interesting to explore if the repeats can be easily identified via such approaches. For the l-SCRFs model, we stop the iterative searching algorithm when the differences of loglikelihood is less than 1e-3 or the iterations are larger than 5000; the number of sampling steps T in the contrastive divergence is set to 5; and the number of iterations in the simulated annealing algorithm is set to 500.

8.5.1 Triple β-Spirals

Table 8.2 and 8.3 list the comparison results of different approaches for recognizing the triple-β spirals. From the results we can see that the sequence similarity based methods, such as PSI-BLAST and Pfam performs poorly. The structure-based algorithms, such as HMMER based on structural alignment and threading algorithm, fail to gain improvement even given additional information. It can be seen that the task we are trying to tackle is dramatically difficult than the common fold classification tasks: the fold involve very complex structures, yet there are only three positive examples without sequence conservation. However, our methods not only can score all the known triple beta-spirals higher than the negative sequences, but also is able to recover most of the repeats from the segmentation (see Table 8.4 and Figure 8.6).

Figure 8.7 shows the histograms of the log-ratio score of the TBS proteins and the PDB-minus dataset. We can see that there is a relatively clear separation between the positive and negative examples. Of all the proteins scored higher than 0 in the PDB-minus set, there are 58 proteins from α class, 45 from β class, 51 from α/β class, 72 from $\alpha + \beta$ class , 4 from α and β class, 6 from membrane class. The false-alarm proteins with the highest scores (most confusing to L-SCRFs) are listed in Table 8.5. We also hypothesize potential TBS proteins from the Swiss-Prot using L-SCRFs. The whole list can be accessed at http://www.cs.cmu.edu/∼yanliu/swissprot_list.xls.

8.5.2 Double-barrel Trimer

From Table 8.6 and 8.7, we can see that it is extremely difficult in predicting the DBT fold. However, our method is able to give higher ranks for 3 of the 4 known DBT proteins, although we are unable to reach a clear separation between the DBT proteins and the rest. The results are within our expectation because the lack of signal features and unclear understanding about the inter-chain interactions makes the prediction significantly harder. We believe more improvement can be achieved by combining the results from multiple algorithms. Figure 8.8 shows the histograms of the log-ratio score

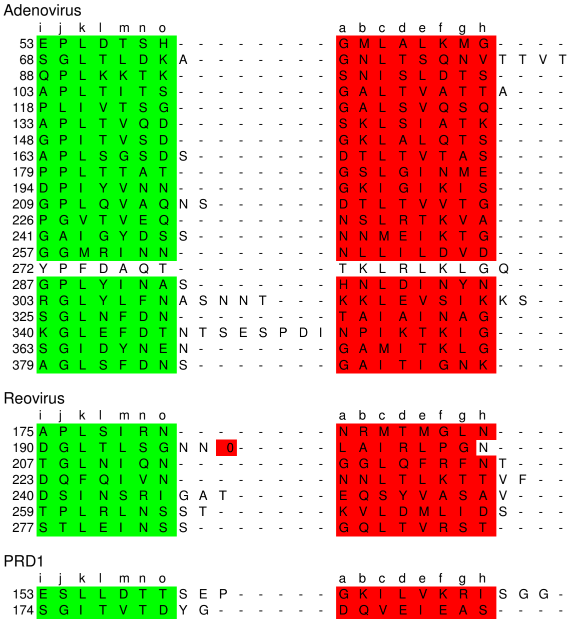

Adenovirus
i j k l m n o a b c d e f g h
53 E P L D T S H - - - - - - - - G M L A L K M G - - - -
68 S G L T L D K A - - - - - - - G N L T S Q N V T T V T
88 Q P L K K T K - - - - - - - - S N I S L D T S - - - -
103 A P L T I T S - - - - - - - - G A L T V A T T A - - -
118 P L I V T S G - - - - - - - - G A L S V Q S Q - - - -
133 A P L T V Q D - - - - - - - - S K L S I A T K - - - -
148 G P I T V S D - - - - - - - - G K L A L Q T S - - - -
163 A P L S G S D S - - - - - - - D T L T V T A S - - - -
179 P P L T T A T - - - - - - - - G S L G I N M E - - - -
194 D P I Y V N N - - - - - - - - G K I G I K I S - - - -
209 G P L Q V A Q N S - - - - - - D T L T V V T G - - - -
226 P G V T V E Q - - - - - - - - N S L R T K V A - - - -
241 G A I G Y D S S - - - - - - - N N M E I K T G - - - -
257 G G M R I N N - - - - - - - - N L L I L D V D - - - -
272 Y P F D A Q T - - - - - - - - T K L R L K L G Q - - -
287 G P L Y I N A S - - - - - - - H N L D I N Y N - - - -
303 R G L Y L F N A S N N T - - - K K L E V S I K K S - -
325 S G L N F D N - - - - - - - - T A I A I N A G - - - -
340 K G L E F D T N T S E S P D I N P I K T K I G - - - -
363 S G I D Y N E N - - - - - - - G A M I T K L G - - - -
379 A G L S F D N S - - - - - - - G A I T I G N K - - - -
Reovirus
i j k l m n o a b c d e f g h
175 A P L S I R N - - - - - - - - N R M T M G L N - - - -
190 D G L T L S G N N 0 - - - - - L A I R L P G N - - - -
207 T G L N I Q N - - - - - - - - G G L Q F R F N T - - -
223 D Q F Q I V N - - - - - - - - N N L T L K T T V F - -
240 D S I N S R I G A T - - - - - E Q S Y V A S A V - - -
259 T P L R L N S S T - - - - - - K V L D M L I D S - - -
277 S T L E I N S S - - - - - - - G Q L T V R S T - - - -
PRD1
i j k l m n o a b c d e f g h
153 E S L L D T T S E P - - - - - G K I L V K R I S G G -
174 S G I T V T D Y G - - - - - - D Q V E I E A S - - - -

Figure 8.6: Segmentation results of the known triple β-spirals by SCRFs. Yellow bar: B1 strand; red bar: B2 strand

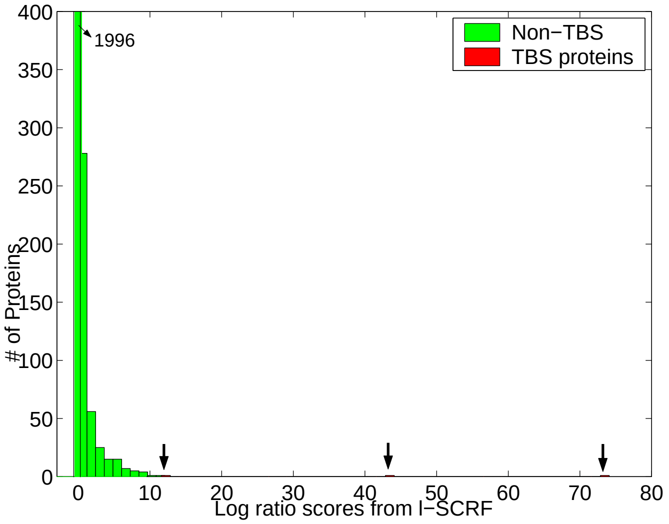


Figure 8.7: Histograms of l-SCRF scores on positive triple-beta spirals (red bar with arrow indicator) and negative set PDB-select (green bars).

Table 8.2: Results of PSI-blast search on triple-β spirals. "x" denotes that the testing protein appears in the result hit list of the query sequence; "-" denotes no hit.

Query Sequence	Adenovirus	Reovirus	PRD1
Adenovirus	x	x	-
Reovirus	x	x	-
PRD1	-	-	x

Table 8.3: Cross-family validation results of the known triple β-spirals by PFAM, HMMER using structural alignment, Threader, RADAR and l-SCRFs. Notice that the scores from the HMMER and Pfam are not directly comparable on different proteins.

SCOP family	Pfam		HMMER		Threader	l-SCRFs	
	score	rank	score	rank	rank	score	rank
Adenovirus	-343.9	**11**	-225.5	**7**	**26**	74.1	**1**
Reovirus	7.9	**1**	-294.3	**2**	**242**	11.6	**1**
PRD1	-6.7	**7**	-399.4	**194**	**928**	43.4	**1**

Table 8.4: # of repeats corrected predicted by different methods for the triple β-spiral proteins

SCOP family	Swiss-Prot ID	PDB ID	# of Correct Repeats		
			Truth	RADAR	L-SCRF
Adenovirus	FIBP_ADE02	1qiu	22	3	21
Reovirus	COA5_BPPRD	1kke	8	2	7
PRD1	VSI1_REOVD	1yq8	2	0	2

of the double-barrel trimer proteins and the PDB-minus dataset. Of all the proteins scored higher than 0 in the PDB-minus set, there are 45 proteins from α class, 37 from β class, 88 from α/β class, 28 from $\alpha + \beta$ class , 14 from α and β class, 7 from membrane class. The most confusing proteins are listed in Table 8.8.

8.6 Summary

In this chapter, we develop linked segmentation conditional random fields (l-SCRFs), for predicting complex protein folds involving multiple chains. Following the framework of conditional graphical models, a protein struc-

Table 8.5: Examples of high-scored proteins in the PDB-select dataset

PDB id	L-SCRF score	SCOP Cluster #	Description
1z7a	12.09	-	hypothetical Pseudomonas aeruginosa PAO1
1zmb	9.68	-	Putative Acetylxylan Esterase from Clostridium acetobutylicum
1xa7	9.64	e.3.1.1	beta-lactamase/transpeptidase-like
1tm0	9.55	d.21.1.3	Diaminopimelate epimerase-like
1kgs	8.83	a.4.6.1	C-terminal effector domain of the bipartite response regulators
1m2w	8.77	a.100.1.9	6-phosphogluconate dehydrogenase C-terminal domain-like
1p16	8.00	b.40.4.6	Nucleic acid-binding proteins
1td5	7.94	d.110.2.2	GAF domain-like
1yox	7.88	-	hypothetical protein PA3696 from Pseudomonas aeruginosa

tural graph is defined, in which the nodes represent secondary structural components of unknown lengths and the edges indicate the inter- or intra-chain long range interactions in the fold. As a discriminative model, l-SCRFs have the flexibility to include any types of features, such as overlapping or long-range interaction features. Due to the complexity of the model, exact inferences are computationally prohibitive. Therefore we propose to use the reversible jump Markov chain Monte Carlo for inferences and optimization. Our model is applied to predict two protein folds and the cross-family validation shows that our method outperforms other state-of-the-art algorithms. For future work, it would be interesting to combine the l-SCRFs model with active learning, in which we can automatically bootstrap negative features from the motif databases (e.g. Pfam or PROSITE) using false positive examples in the previous iterations.

Table 8.6: Results of PSI-blast search on double-barrel trimer proteins (3-iterations with cutoff score of 0.05)

	Adenovirus	PRD1	PCBV-1	STIV
Adenovirus	x	-	-	-
PRD1	-	x	-	-
PCBV-1	-	-	x	-
STIV	-	-	-	x

Table 8.7: Cross-family validation results of the known double-barrel trimer by HMMER (profile HMM) using sequence alignment (seq-HMMER) and structural alignment (struct-HMMER), Threader and l-SCRFs. Notice that the scores from different methods are not directly comparable on different proteins.

SCOP family	Seq-HMMER		Struct-HMMER		Threader	l-SCRFs	
	score	rank	score	rank	rank	score	rank
Adenovirus	-196.4	**12**	-165.1	**14**	$>$ **385**	38.6	**87**
PRD1	-457.8	**84**	-381.3	**107**	**323**	75.5	**8**
PBCV	-295.3	**92**	-344.3	**8**	**321**	94.0	**3**
STIV	-520.4	**218**	-390.4	**70**	**93**	123.6	**2**

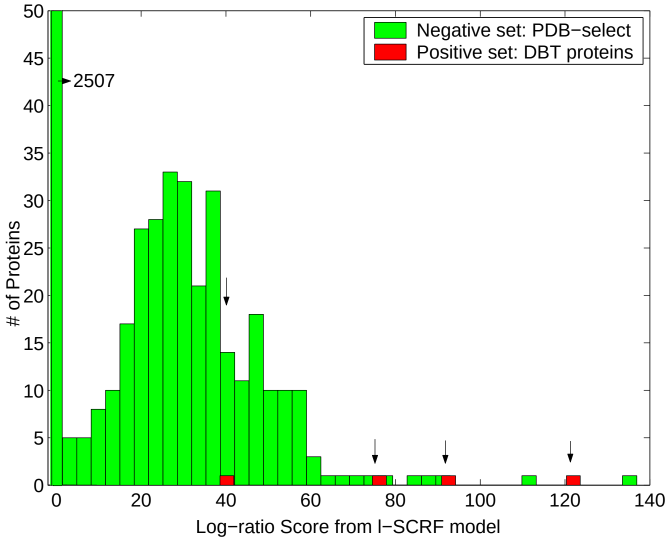


Figure 8.8: Histograms of L-SCRF scores on positive double-barrel trimer (red bar with arrow indicator) and negative set PDB-select (green bars).

Table 8.8: Examples of high-scored proteins in the PDB-select dataset

PDB id	L-SCRF score	SCOP Cluster #	Description
1qcr	136.97	f.32.1.1	a subunit of cytochrome bc1 complex
1qle	113.00	f.24.1.1	Cytochrome c oxidase subunit I-like
1g55	89.92	c.66.1.26	Enigmatic DNA methyltransferase homolog
1t1u	87.96	-	Choline Acetyltransferase
1ln6	83.07	f.13.1.2	G protein-coupled receptor-like
1he8	78.80	a.118.1.6	ARM repeat
1xtj	75.75	-	human UAP56 in complex with ADP
4cts	71.69	a.103.1.1	Complex of citrate synthase
1kpl	66.17	f.20.1.1	Clc chloride channel
1ofq	65.67	c.1.10.4	Aldolase in complexes with manganese

Table 8.9: L-SCRF scores of potential double-barrel trimers suggested in (Benson et al., 2004)

Swiss-Prot ID	Description	L-SCRF score (prob)
P11795	Tomato bushy stunt virus (TBSV)	29.60 (40.1%)
P22776	(p72) African swine fever virus (ASFV)	7.92 (0.00%)
Q05815	(MCP) Chilo iridescent virus (CIV)	32.09 (89.0%)
Q5UQL7	Probable capsid protein 1 - Mimivirus	41.90 (99.9%)
Q6X3V1	Bacillus thuringiensis bacteriophage Bam35c	33.53 (97.2%)
Q8JJU8	(Fragment) Trichoplusia ni ascovirus 2a	42.86 (99.9%)
Q8QN59	Ectocarpus siliculosus virus 1	44.12 (99.9%)
Q9YW23	Poxvirus	42.88 (99.9%)

Chapter 9

Conclusion and Future Work

In this thesis, we develop a framework of conditional graphical models for protein structure prediction. We focus on predicting the general structural topology of proteins, rather than specific 3-D coordinates of each atom. Based on the structural characteristics at each level in the protein structure hierarchy, we can develop a corresponding conditional graphical model to capture the interactions between structural components, which correspond to the chemical bonding essential to the stability of the protein structures. To our best knowledge, this approach is one of the first probabilistic models to capture the long-range interactions directly for protein structure prediction.

In our exploration, we have demonstrated the effectiveness of conditional graphical models for protein secondary structure prediction, tertiary motif recognition with two example motifs, i.e. right-handed β-helix and leucine-rich repeats, and quaternary motif recognition on two specific examples, i.e. triple β-spirals and double-barrel trimer. We confirm our thesis statement that conditional graphical models are theoretically justified and empirically effective for protein structure prediction, independent of the structure hierarchies of target outputs.

9.1 Contribution and Limitation

The contribution of this thesis work involves two aspects. From computational perspective,

1. We propose a series of conditional graphical models under a unified framework. They enrich current graphical models for structured prediction, in particular for handling the long-range interactions common

in various applications, such as information extraction and machine translation. They furthermore relax the iid assumption about data with inherent structures theoretically.

2. With millions of sequences in the protein databank (Swiss-Prot or UniProt), efficient structure prediction algorithms are required. Therefore for each graphical model in the framework, we develop a corresponding inference and learning algorithm. Our large scale applications have demonstrated the efficiency of these inference algorithms and the possibility of applying graphical models in other genome-wide or large-scale applications.

3. In protein structure prediction, we have to handle the data with characteristics well beyond the classical learning scenario: there are very few positive examples for most of the motifs we work on; the labels and features are quite noisy; the application needs massive data processing (millions of sequences) while computational resources are limited. We are able to resolve most of these challenges by incorporating prior knowledge into graphical models. This serves as a good example to demonstrate how domain knowledge can compensate the lack of reliable data. Although our discussions are focused on applications in computational biology, the methodologies are easily transferrable to other applications.

From biological perspective,

1. We use CRFs and kernel CRFs for protein secondary structure prediction and achieve some improvement over the state-of-art methods using window-based approaches.

2. We develop SCRFs, one of the first probabilistic models to capture the long-range interactions globally for protein fold recognition. It has been proven to be effective at identifying examples of very complex motifs where most traditional approaches fail. We also hypothesize potential membership proteins of the β-helix motif. We hope that the results will provide useful guidance to the biologists in related areas for their experiments.

3. We develop linked SCRFs, one of the first probabilistic models specifically for quaternary motif recognition. It is also one of the early models to successfully make predictions for viral motifs.

4. In general, our work helps to provide a better understanding on the mapping from protein sequences to their structures. We hope that our prediction results will shed light on the functions of some protein folds and aid drug design.

Until now, we have verified our thesis statement, i.e. conditional graphical models are an effective solution for protein structure prediction. At the same time, there are also several limitations:

1. The models provide the convenience to use any types of informative features. However, there is no guideline on how to extract those meaningful features from protein sequences automatically. Most of our features come from domain experts, who devote tremendous time and efforts on the study of our target motifs. We have not found an efficient way to generate the features for all the motifs and structures. We make an early efforts to examine if we can bootstrap features automatically from the motif databases (e.g. Pfam or PROSITE), but fail to achieve further improvement. More elaborated extensions will be part of the future work.

2. Obtaining the ground truth, i.e. the true structures of concerned proteins, requires lab experiments with long waiting time (1-5 years or more). As a result, many of our prediction results cannot be verified in the near future although we do get encouraging results for several proteins whose structures have been resolved recently.

3. Another limitation of our work is the computational complexities. Our models have a much higher complexity than the similarity-based approaches (both are polynomial $O(n^d)$, but $d >= 3$ for the former and d=1 or 2 for the latter). The advantage of our model is a better performance (in terms of prediction accuracy and sensitivity) for the most difficult target folds (motifs). A natural solution for a genome-wide application is to use sequence-based methods on simple folds, and apply our model on the more complex and challenging folds.

9.2 Future Work

For future work, we would like to examine multiple directions, including:

Efficient inference algorithms In the thesis, we have examined several inference algorithms, such as belief propagation and MCMC sampling.

There are some recent advances in the research of efficient inference algorithms for graphical models. For example, pre-conditioner approximation and structured variational approximation. We have not fully examined these alternatives since the main theme of the thesis is to develop efficient models and solve important biological problems. As future work, it will be interesting to examine the efficiency and effectiveness of different approximation algorithms. On one hand, we can find the best algorithm for the graphical models we develop here; on the other hand, our experiment settings (complex graphs and noisy, unbalanced training data) provide an excellent testing case for a thorough comparison of different inference algorithms.

Viral protein structure prediction Viruses are a noncellular biological entity that can reproduce only within a host cell. Most viruses consist of nucleic acids covered by proteins while some animal viruses are also surrounded by membranes. Inside the infected cell, the virus uses the synthetic capability of the host to produce progeny virus and attack the host cell. There are many types of viruses, either species-dependent or species-independent. For example, some famous viruses are known to be unique to human beings, such as human immunodeficiency virus (HIV), tumor virus, sudden acute respiratory syndrome (SARS) virus.

The structures of viral proteins are very important for studying the infection processes and designing drugs to stop the infection. However, it is extremely difficult to acquire their structures by lab experiments since the genes of viruses mutate rapidly and therefore the structures of the proteins change accordingly. Give the limited number of training instances, there are very few computational methods that can successfully predict the structures of viral proteins.

Many examples we use in our experiments are viral proteins, such as the right-handed β-helices, triple β-spirals and double-barrel trimer. Our successes in these proteins show strong indication that our model might be useful for sequence analysis and structure prediction of other viral proteins. Therefore it would be interesting to examine this direction and verify the generality of our model in this exciting area.

Protein function prediction It is widely believed that protein structures reveal important information about their functions, but it is not straightforward to map the sequences to specific functions since most functional sites or active binding sites consist of only a few residues, which may be quite distant in sequence order. Previous work on protein function prediction can be summarized into two approaches. One approach is to collect and combine the information from multiple resources, such as the database of functional motifs, microarray data, protein dynamics and so on. The other approach is motivated by the structural properties of functional sites. For example, TRILOGY is a system that searches all the possible triplets (a combination of three residues) in protein sequences and selects only a subset as seeds to search for longer patterns (Bradley et al., 2002). Both approaches have achieved some successes, but the current prediction results are still far from practical use.

In the thesis, we have studied some protein families with structural repeats, such as the leucine-rich repeats and TIM barrel fold. These structures provide a stable frame so that the active sites can perform their functions. By segmenting the protein sequences against these motifs, we manage to know the locations of the structural frame and the active sites. Along the direction, we choose the ankyrin repeats, one of the most common motifs in protein-protein interactions, as a study case. Ankyrin repeats are tandem modules of about 33 amino acids: each repeat folds into a helix-loop-helix structure with a β-hairpin (or loop region) projecting out from the helices at a 90 degree angle. The repeats stack together and form an L-shaped structure. The ankyrin repeat has been found in proteins with diverse function such as transcriptional initiators, cell-cycle regulators, cytoskeletal, ion transporters and signal transducers. Our future plan is to apply our model to the motifs from one or two subfamilies in the ankyrin repeats. This information, combined with other features indicative of the functions, such as the location information and the results from the mircoarray data analysis, may provide a reasonable solution for function identification.

Other applications In addition to applications in biology, there are many other tasks involving sequential observations with long-range interactions, such as information extraction and video segmentation. It would be interesting to apply our conditional graphical models to other applications and testify the generality of the thesis statement. We are now pursuing the idea on information extraction in the medical domain.

Appendix A

Notation and Index

capital letter	constants and random variables
lower-case letter	observed variables
bold letter	vectors

$\mathbf{x} \in \mathcal{X}$	input and input space
$\mathbf{y} \in \mathcal{Y}$	output and output space
N	dimension of input space
M	number of segments of x in segmentation space
K	dimension of feature space
L	training set size
Z	normalizer over all possible configuration of y
W	segmentation and labeling of x
R	loss function to be optimized
G	a graph
V	a set of nodes in graph G
E	a set of edges in graph G
f	feature function
p	starting position of a segment
q	ending position of a segment
s	the state of a segment or a node
$\mathcal{C}$	a set of cliques in a graph
λ	weight for the features
α	forward probability
β	backward probability
δ	indicator function

Index

Appendix B

Details of Example Proteins

In this appendix, we describe the details about the protein folds we select in our experiments, including the right-handed β-helix, leucine-rich repeats, triple β-spirals and double-barrel trimer. These folds are good examples of what most other prediction algorithms fail to predict. They all exhibit complex structures, involve in many important biological functions but have very few positive training data. We collect the domain knowledge about these folds from the literature, domain experts and the online resources. We also make some observations by examining the structures of the proteins ourselves.

B.1 Right-handed β-helix

The right-handed parallel β-helix fold is an elongated helix-like structure with a series of progressive stranded coilings (called *rungs*), each of which is composed of three parallel β-strands to form a triangular prism shape (Yoder et al., 1993). The typical 3-D structure of a β-helix is shown in Fig.B.1(A-B). As we can see, each basic structural unit, i.e. a rung, has three β-strands of various lengths, ranging from 3 to 5 residues. The strands are connected to each other by loops with distinctive features. One loop is a unique two-residue turn which forms an angle of approximately $120^\diamond$ between two parallel β-strands (called *T-2 turn*). The other two loops vary in size and conformation, which might contain helix or even β-sheets.

The β-helix proteins are significant in that they include pectate lyases, which are secreted by pathogens and initiate bacterial infection of plants; the phage P22 tailspike adhesion that binds the O-antigen of Salmonella typhimurium; and the P.69 pertactin toxin from Bordetella pertussis, the cause of Whooping Cough. Therefore it would be very interesting if we can accurately predict other unknown β-helix structure proteins.

Currently there are 14 β-helix proteins whose structures have been determined. Those proteins belong to 9 different SCOP families (Murzin et al., 1995). Computationally, it is very difficult to detect the β-helix fold because the membership proteins do not exhibit strong sequence identity (less than 25%), which is the "twilight zone" for sequence-based methods, such as PSI-BLAST or HMMs. From previous literature on β-helix, there are two properties about the fold essential for accurate prediction: 1) the β-strands of each rung have patterns of pleating and hydrogen bonding that are well conserved across the superfamily; 2) the interaction of the strand side-chains in the buried core are critical determinants of the fold (Yoder & Jurnak, 1995; Kreisberg et al., 2000).

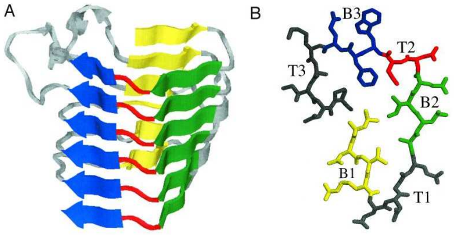


Figure B.1: 3-D structures and side-chain patterns of β-helices; (A) Side view (B) top view of one rung (C) Segmentation of 3-D structures (D) protein structural graph. E1 = {black edge} and E2 = {red edge} (Figure (A) and (B) are adapted from (Bradley et al., 2001))

B.2 Leucine-rich Repeats

The leucine-rich repeats are solenoid-like regular arrangement of β-strand and α-helix, connected by coils (Fig.B.2). Based on the conservation level, we define the *motif* for LLR as the β-strand and short loops on two sides, resulting 14 residues in total. The length of the *insertions* varies from 6 to 29. There are 41 LLR proteins with known structure in PDB, covering 2 super-families and 11 families in SCOP. The LLR fold is relatively easy to detect thanks to its sequence conservation with many leucines and short insertions. Therefore it would be more interesting to discover new LLR proteins less similar to the previously known ones.

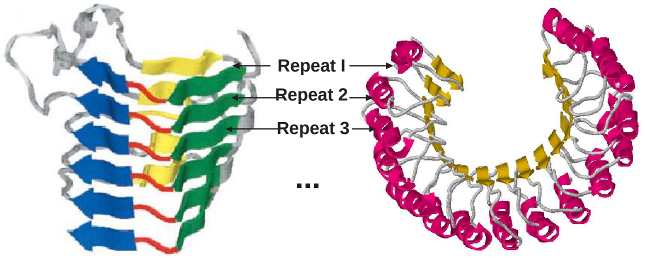


Figure B.2: (Left): beta helices; (Right) Leucine-rich repeats

B.3 Triple β-spirals

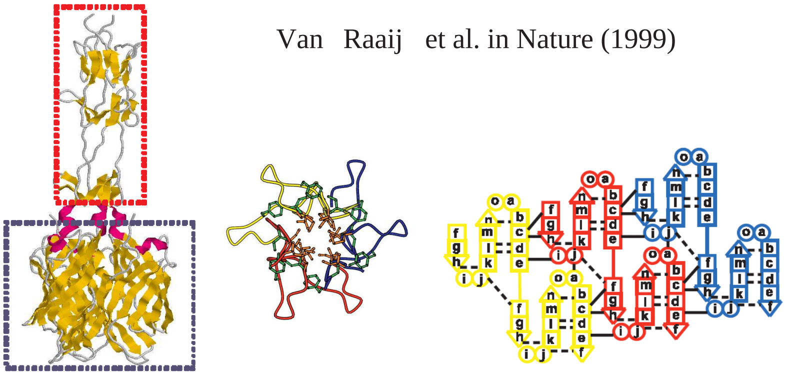


Figure B.3: Demonstration graph of triple β-spirals. (left) 3-D structures view. Red block: shaft region (target fold), black block: knob region. (middle) top view. (right) maps of hydrogen bonds within a chain and between chains.

The triple β-Spiral fold is a processive homotrimer consisting of three identical interacting protein chains. It was first identified by Mark J. van Raaij and collaborators in 1999 (van Raaij et al., 1999). The fold serves as a fibrous connector from the main virus capsid to a C-terminal knob that binds to host cell-surface receptor proteins (see Figure 8.3). Up to now there are three crystallized structures with this fold deposited in the Protein Data Bank (PDB) (Berman et al., 2000), one is the adenovirus protein (DNA virus, PDB ID: 1qiu), another is reovirus (RNA virus, PDB ID: 1kke) and the other is PRD1 (PDB ID: 1yq8). The common existence in both DNA and RNA viruses reveals important evolution relationships in the viral proteins, which also indicates that the triple beta-spirals might be a common fold in nature.

The triple-β spiral fold has several structural characteristics that distinguish itself from others: it consists of three identical protein chains with a series of repeated structural elements, which is referred as "rung" in our later discussion (see Figure 8.3). Each of these structural elements is composed of: 1. A β-strand that runs parallel to the fiber axis 2. A long solvent-exposed loop of variable lengths, 3. A second β-strand that forms antiparallel β-sheets with the first one, and slightly skewed to the fiber axis, 4. successive structural elements along the same chain are connected together by a tight β-turn (Scanlon, 2004; Weigele et al., 2003). Among those four components, the two beta-strands are quite conserved in sequences and Green et al. characterize them by labeling each position using character 'a' to 'o' (Green, 1995). Specifically, i-o for the first strand and a-h for the second strand (see Figure 8.3).

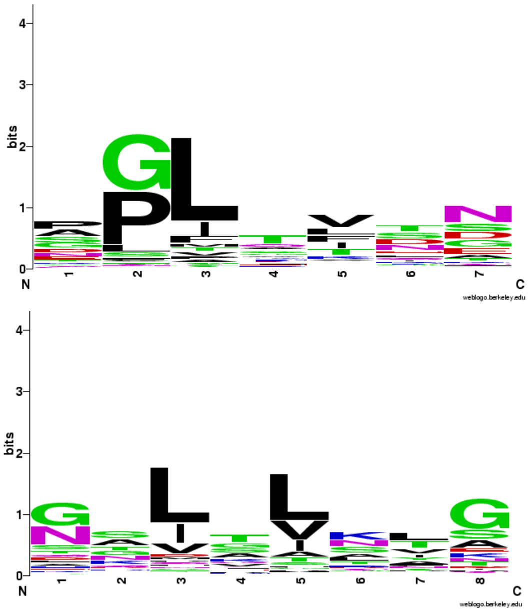


Figure B.4: Sequence alignment logo of the first β-strand and second β-strand.

It is extremely challenging to predict the triple-β spirals with only three positive examples. Fortunately, the structural repetitive patterns have been partially reflected in the sequences conservation. The sequence alignment logo is shown in Figure B.4. More careful study suggests that we can get

```
PAFTVSN
GAIGYDS
GALGFDS
SALIMSG

PFTTTNE
PFNVVNN
PFVTPPF
PFVTPPF
PFITPPF
PFITPPF
GFPPPFF

PLIVTSG
PLTVSNN
PLTVSNN
PLSILKN
PLMVAGN
PLTTTDE
PLNVVNN
PLALQDH
GLQISNN
PLQFQGN
GLLNVRL
KLSTGPG
QLLLGSG
KLKTGGG
TLGVGRG
TLAFGGG

PSLHLEE
ESMQVTE
ESMQVTE
ESMQVTE
DSMQVTN
QSLDVED
SSVAAFT
SSPGTLA
SSPGTLA
SSPGTLA
DSGKANT
ESLLDTT

NGALTLK
LGAIKLS
LGAIKLS
LGAIKLS
DGTGKLT
DGTGKLT
EGTGNLT
SGITVTD
```

```
SGLTLDK
SGLTVDT
GGLTIDD
GGLTVDD
GGLTVDD
GGLTLQE
GGLTLQD
GGLTLQD
GGLTLQE
EGVDLDD
NGLSLDE
DGIKLNA
NGIKLNS
TGLNIDE
TGLNIDK
AGLILKE
AGLILKE
AGLILKE
PGLTLNE
SGLSVNA
GGLTADA
RGIRINP
GGLQLSG
EGLEDES
EGLEDES
EGLEDES
DGLEDEN
EGLSVDH
PGLSNSE
RGLVITN
RGLVITN
NGLQIEQ
PGLGTNE
AGLGTDE
AGLGTNE
SGLGLSG
QGLQVND
SGITVDA
AGLQNTD
PGLRMLN
PGLGTDN
SGLTTDG
GGMRVDG
GGMRVDG
RGLHVTT
AGLAVQD
AGLAVQD
AGLAVQD
AGLAVQS
QGFQVVA
AGLSIQG
AGLSIQG
```

```
AGLSIQG
AGLSVQN
SGLAVTE
PGMWVDQ
PGVTVEQ
PGVTINN
NGLQVSG
NGLQVSG
EGIQVKE
DGLTFDN
DGLRFDN
DGLALGG
DGLALGG
SGLRVSG
SGLRVSG
SGLRVSG
SGLRISG
NGLAVTE
GGMRINN
GGLRIDS
GGMRTSG
GGNETLR
PGLQMSN
PGLQMSN
PGLSHIN
PGLSHIN
PGLSHIN
PGLSHQN
NGIKVDE
NGLEFSN
TGNFVSS
RGLYLFN
KGLYLFT
SGLNFDN
KGLMFDA
KGLMFSG
KGLTFSG
SGLIMSG
NGLTLTD
KGLEFDT
DGLEFGS
DGLEVRD
SGIDYNE
HGLEFDS
CGLTFNN
KGLAVEN
KGLAIEN
AGLKFEN
AGLKFEN
AGLKFEN
AGLKFEN
QGLEIAD
```

```
AGLSFDN
SGLQFDS
SGLQFDN
GGLSFND
GGLSFNN
GGLSFNN
GGLSFNE
SGLSFDS
TGLSFDS
AGLIFDS
AGHTFSS
NGLRFDS
NGLRFDS
NGLTIRD
NGLTIRD
NGLTIRD
NGLTIRD
YGFHAHR
DGLQFDS
NGTLTLK
NGSLTLK
NGALTLK

SPLYLDS
PPITVES
APLVSGS
GPLFINS
DPLTVNS
KPLALRS
TPLAVSS
DPLMVSS
EPLPHTS
APLTITS
EPLTNTS
APLSTTS
APLTVTS
PPLTTAT
PPLTTAT
PPLVFDT
SPLAIET
YPFDAQT
YPFDATT
KPPGVLA
KPPGVLS
KPPGVLS
KPPGVLS
NPIEVNQ
NPLTISQ
APLSVSQ
TPLVVNR
NPLTTAS
SPLYLDS
```

```
PPITVEA
APLAVKA
APLSVKA
QPVTINA
GPLYINA
QPVTVNA
APIAVSA
APIAVSA
SPITLTA
GPLTTTA
APLTVHD
APFDVID
APFDVID
APFDVID
APLQIND
YPLVKND
APLTVQD
LPLQYRD
APLSGSD
DPLAISD
GPITVSD
PPFLITD
PPFTITD
EPLSKTD
GPLTTTD
APLHVTD
APLGLVD
APLGLVD
KPLTFDE
SPLHKIE
SPLHKIE
SPLHKIE
SPLHKNE
GPLTVSE
EPLLETE
EPLLETE
EPLLETE
APFDVIG
APLQFSG
GPFTVSG
GPFTVSG
TPLVKTG
TPLVKTG
TPLVKTG
TPLTTTG
TPLTTTG
TPLTTTG
EPLDTSH
DPITTNK
DPITTNK
PPLKKTK
QPLKKTK
```

```
PPLTFSL
PPLTFSL
PPLTFSL
PPLTFSL
IPLYTKM
DPIAIAN
PPLKIEN
PPLKIEN
DPIVTEN
DPIVTEN
SPITLIN
SPLKVIN
SPITVIN
RPCHTKN
DPLTTKN
DPLTTKN
LPLQYKN
DPITINN
YPLIKNN
YPLIKNN
YPLIKNN
DPIYVNN
YPFDAQN
KPLALQN
EPIYTQN
YPFDASN
TPLTKSN
PPLTNSN
PPLTNSN
EPLVTSN
APLDVSN
PPLKKTN
PPLQKTN
PPLQKTN
PPLQKTN
APITKTN
APITKTN
PPLHLTN
PPLQLTN
PPLQLTN
PPLQLTN
PPLTYTN
DPIAIVN
DPIAIVN
GPLQVAQ
APLSFFQ
```

Figure B.5: Sequence alignment of the first β-strand grouped by the 'j' position.

even more conserved sequence alignment if we group them based on the amino acid type on the 'j' position (see Figure B.5). The identification of this pattern play an essential role in successfully predicting the triple β-spirals. Another interesting observation about the TBS fold is that the three component chains interwind with each other to form a rigid fiber with a hydrophobic core. The accessibility and hydrophobic properties might be indicative for distinguishing TBS from others.

B.4 Double-barrel Trimer

The double-barrel trimer is a potential protein fold, which has been found to in the coat proteins from several kinds of viruses. It consist of two eight-stranded jelly rolls, or β-barrels. As seen in Figure B.6, the component β-strands are labeled as B, C, D, E, F, G, H and I respectively. The first

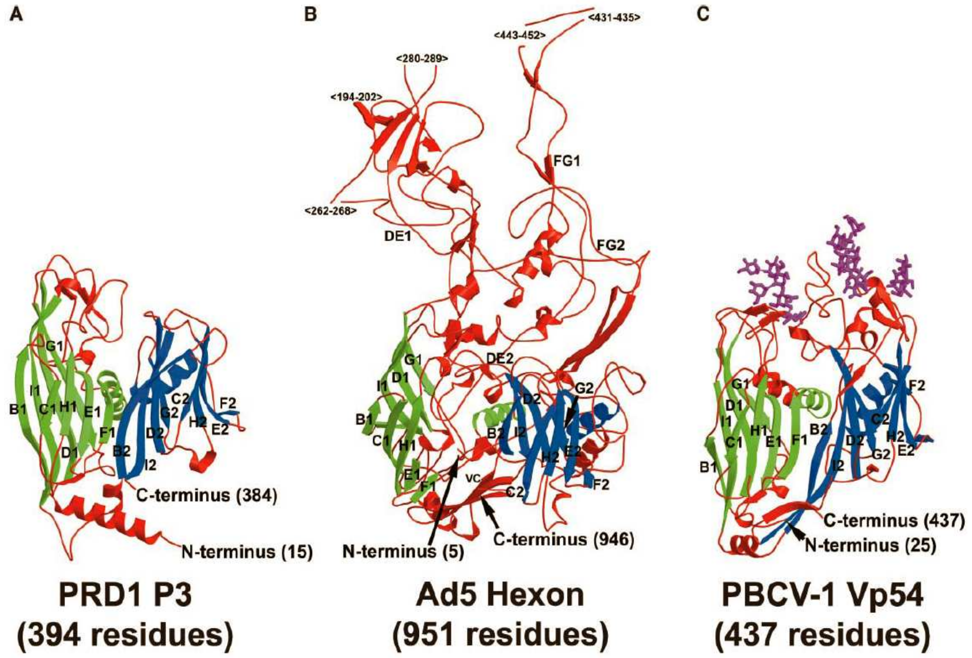


Figure B.6: X-Ray Crystal Structures of Viral Double-Barrel Trimeric Major Coat Proteins (A) P3 of bacteriophage PRD1 (394 residues; PDB code 1hx6; Benson et al., 2002), (B) Hexon of adenovirus type 5 (Ad5; 951 residues; PDB code 1p30; Rux et al., 2003), and (C) Vp54 of Paramecium bursaria chlorella virus 1 (PBCV-1; 436 residues; PDB code 1m3y; Nandhagopal et al., 2002). The eight β strands and a flanking α-helix are displayed for the first (green) and second (blue) jelly rolls, and the individual strands are labeled (B1-I1 and B2-I2, respectively). Graph and caption adapted from Benson et al, 2004.

strand is named as B because one example of the β-barrels, the tomato bushy stunt virus, has an extra initial strand. The folded β-barrel has two sheets, each consisting of four β-strands, i.e. BIDG and CHEF, with hydrogen bonds within a sheet, but not across the edges. Notice that there is no hydrogen bonds between B and C, nor F and G, therefore it is not a true β-barrel like the β/α motif. At its native state, multiple identical protein chains of the double-barrel timer will come together with chemical bonds (mostly hydrogen bonds), forming trimeric hexon protein arranged on the planes and a penton complex at each of the twelve vertices. The rest of our discussion are focused on the trimeric hexons.

The importance of studying the double-barrel trimer is far beyond simple verification of our proposed computational model. Biologically speaking, the fold has been found in the major coat proteins of bacteriophage PRD1, that of human adenovirus, Paramecium bursaria chlorella virus (PBCV) and archaeal virus STIV. This amazing phenomenon raised the unexpected possibility that viruses infecting different kinds of species are related by evolution. It has been suggested that the occurrence of a double-barrel trimer is common to all icosahedral dsDNA viruses with large facets, irrespective of its host, and furthermore an indicator of common ancestor in a lineage of viruses (Benson et al., 2004). Notice that similar observations have been made for the triple β-spirals. If we can find more examples of the double-barrel trimer in other viruses, the statement would be strengthened greatly and bring more significant impact to the biology science via computational methods.

However, it is not straightforward, or even seemingly impossible, to uncover the structural conservation through sequences only. There are 4 double-barrel trimer proteins altogether with resolve structures, including adenovirus (PDB ID: 1P2Z), PRD1 (PDB ID: 1CJD), PBCV (PDB ID: 1M4X) and STIV (PDB ID: 2BBD). The sequence similarity between the positive proteins are around 7-20%, which is significantly lower than other protein folds that we have studied before. Figure B.8 shows the alignment of those 4 proteins according to structures. After careful examination, we find no obvious patterns (such as regular expression templates) to uniquely identify the double-barrel trimer. There are several general descriptive observations we can make: (1) the lengths of the eight β-strands varies, ranging from 4 to 16 residues, but the layout of the strands is fixed. The separation (insertions) between the strands is fairly constant (4- 10 residues), however, it is interesting to notice some exceptions, for example the long insertions between the F and G strand (20 - 202 residues), exactly where the inter-chain interactions (chemical bonding) are located,; another long loops be-

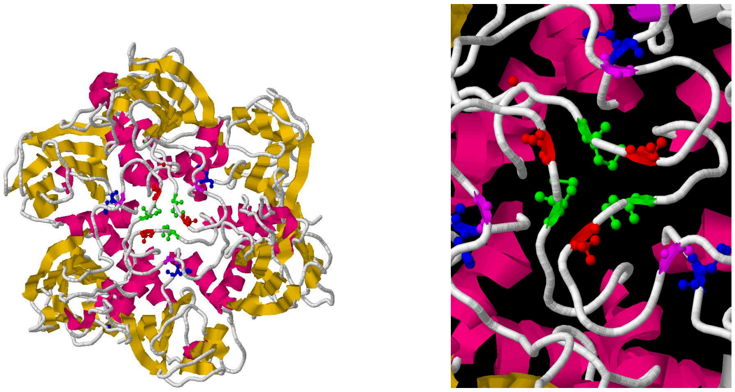


Figure B.7: 3-D structure of the coat protein in bacteriophage PRD1 (PDB id: 1CJD). (left) 3-D structure of PRD1 in trimers. (right) Zoom-in view of the inter-chain and intra-chain interactions in the FG loop. Color notation: green: residue #133; red: residue #135; purple: residue #142; blue: residue #335.

tween D-E strand (9 - 250 residues); the short β-turn between E and F. (2) The chemical bonds that stabilize the trimers are located between the FG loops. We denote the FG loop in the first double-barrel trimer as FG1, and that in the second one as FG2. Figure B.7 shows the side-chain bonding in the FG loop of PRD1 (PDB id: 1CJD). It can be seen that the there are inter-chain interactions (chemical bonding) between some residues in FG1 of different chains and intra-chain interactions between some residues in FG1 and FG2 of the same chain. (3) Most often, the FG loops are buried inside the hydrophobic core. One exception is adenovirus, in which a long α-helix flanking outside the core.

Table B.1: Pairwise sequence similarity between the double-barrel timers

	Adenovirus	PRD1	PCBV-1	STIV
Adenovirus	-	8.40%	8.50%	7.70%
PRD1	-	-	17.00%	15.10%
PCBV-1	-	-	-	7.90%

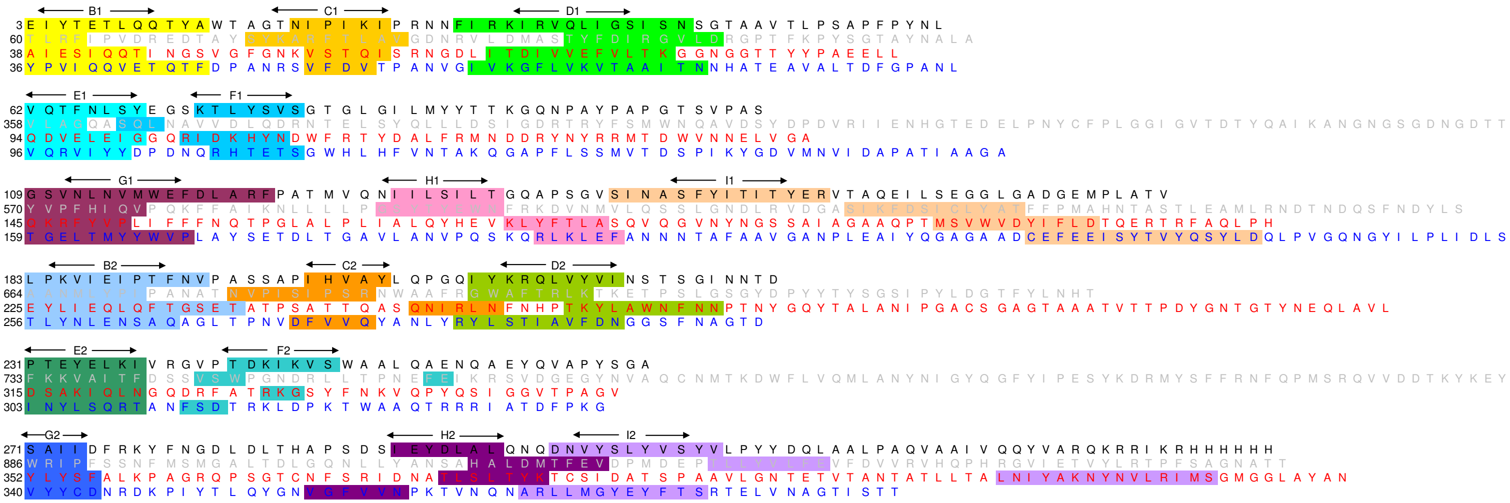

B1 C1 D1
3 E I Y T E T L Q Q T Y A W T A G T N I P I K I P R N N F I R K I R V Q L I G S I S N S G T A A V T L P S A P F P Y N L
60 T L R F I P V D R E D T A Y S Y K A R F T L A V G D N R V L D M A S T Y F D I R G V L D R G P T F K P Y S G T A Y N A L A
38 A I E S I Q Q T I N G S V G F G N K V S T Q I S R N G D L I T D I V V E F V L T K G G N G G T T Y Y P A E E L L
36 Y P V I Q Q V E T Q T F D P A N R S V F D V T P A N V G I V K G F L V K V T A A I T N N H A T E A V A L T D F G P A N L
E1 F1
62 V Q T F N L S Y E G S K T L Y S V S G T G L G I L M Y Y T T K G Q N P A Y P A P G T S V P A S
358 V L A G Q A S Q L N A V V D L Q D R N T E L S Y Q L L L D S I G D R T R Y F S M W N Q A V D S Y D P D V R I I E N H G T E D E L P N Y C F P L G G I G V T D T Y Q A I K A N G N G S G D N G D T T
94 Q D V E L E I G G Q R I D K H Y N D W F R T Y D A L F R M N D D R Y N Y R R M T D W V N N E L V G A
96 V Q R V I Y Y D P D N Q R H T E T S G W H L H F V N T A K Q G A P F L S S M V T D S P I K Y G D V M N V I D A P A T I A A G A
G1 H1 I1
109 G S V N L N V M W E F D L A R F P A T M V Q N I I L S I L T G Q A P S G V S I N A S F Y I T I T Y E R V T A Q E I L S E G G L G A D G E M P L A T V
570 Y V P F H I Q V P Q K F F A I K N L L L L P G S Y Y Y E W N F R K D V N M V L Q S S L G N D L R V D G A S I K F D S I C L Y A T F F P M A H N T A S T L E A M L R N D T N D Q S F N D Y L S
145 Q K R F Y V P L I F F F N Q T P G L A L P L I A L Q Y H E V K L Y F T L A S Q V Q G V N Y N G S S A I A G A A Q P T M S V W V D Y I F L D T Q E R T R F A Q L P H
159 T G E L T M Y Y W V P L A Y S E T D L T G A V L A N V P Q S K Q R L K L E F A N N N T A F A A V G A N P L E A I Y Q G A G A A D C E F E E I S Y T V Y Q S Y L D Q L P V G Q N G Y I L P L I D L S
B2 C2 D2
183 L P K V I E I P T F N V P A S S A P I H V A Y L Q P G Q I Y K R Q L V Y V I N S T S G I N N T D
664 A A N M L Y P I P A N A T N V P I S I P S R N W A A F R G W A F T R L K T K E T P S L G S G Y D P Y Y T Y S G S I P Y L D G T F Y L N H T
225 E Y L I E Q L Q F T G S E T A T P S A T T Q A S Q N I R L N F N H P T K Y L A W N F N N P T N Y G Q Y T A L A N I P G A C S G A G T A A A T V T T P D Y G N T G T Y N E Q L A V L
256 T L Y N L E N S A Q A G L T P N V D F V V Q Y A N L Y R Y L S T I A V F D N G G S F N A G T D
E2 F2
231 P T E Y E L K I V R G V P T D K I K V S W A A L Q A E N Q A E Y Q V A P Y S G A
733 F K K V A I T F D S S V S W P G N D R L L T P N E F E I K R S V D G E G Y N V A Q C N M T K D W F L V Q M L A N Y N I G Y Q G F Y I P E S Y K D R M Y S F F R N F Q P M S R Q V V D D T K Y K E Y
315 D S A K I Q L N G Q D R F A T R K G S Y F N K V Q P Y Q S I G G V T P A G V
303 I N Y L S Q R T A N F S D T R K L D P K T W A A Q T R R R I A T D F P K G
G2 H2 I2
271 S A I I D F R K Y F N G D L D L T H A P S D S I E Y D L A L Q N Q D N V Y S L Y V S Y V L P Y Y D Q L A A L P A Q V A A I V Q Q Y V A R Q K R R I K R H H H H H H
886 W R I P F S S N F M S M G A L T D L G Q N L L Y A N S A H A L D M T F E V D P M D E P T L L Y V L F E V F D V V R V H Q P H R G V I E T V Y L R T P F S A G N A T T
352 Y L Y S F A L K P A G R Q P S G T C N F S R I D N A T L S L T Y K T C S I D A T S P A A V L G N T E T V T A N T A T L L T A L N I Y A K N Y N V L R I M S G M G G L A Y A N
340 V Y Y C D N R D K P I Y T L Q Y G N V G F V V N P K T V N Q N A R L L M G Y E Y F T S R T E L V N A G T I S T T

Figure B.8: Alignment of double-barrel trimer proteins based on structure annotation, including STIV (black), Adenovirus (silver), PRD1 (red), and PBCV (blue). Each β-barrel has eight strands, labeled as B to I and highlighted by different colors respectively.

Appendix C

Feature Extraction

Conditional graphical models provide an expressive framework to capture the structural properties of protein folds characterized by both local interactions, inter-chain and intra-chain interactions. They enjoy the advantages of the original CRF model so that any type of informative features, either overlapping or long-range correlations, can be used conveniently. However, the choice of feature function f_k plays an essential role in accurate predictions.

From the perspective of graph topology, two types of features can be defined, i.e. *node features*, which cover the properties of an individual segment, and *pairwise features*, which tries to model the chemical-bonding between the pairs of segments that are close in three-dimensional spaces. More specifically, for all the models discussed in the thesis, the node features $f(\mathbf{x}, i, \mathbf{w}_i)$ are factorized as follows:

$$f_{(L^\star, S^\star)}(\mathbf{x}, i, \mathbf{w}_i) = f'_k(\mathbf{x}, p_i, q_i)\delta_{(L^\star, S^\star)}(\mathbf{w}_i) = f'_k(\mathbf{x}, p_i, q_i)\delta(q_i - p_i, L^\star)\delta(s_i, S^\star), \tag{C.1}$$

where $L^\star \in [l_{min}, l_{max}]$, $S^\star \in \mathcal{S}$ and $\mathcal{S}$ is the set of state assignments.

The pairwise features $g((\mathbf{x}_a, u, \mathbf{w}_{a,u}), (\mathbf{x}_b, v, \mathbf{w}_{b,v}))$ are factorized as:

$$\begin{aligned} &g_{(L_a^\star, S_a^\star),(L_b^\star, S_b^\star)}((\mathbf{x}_a, u, \mathbf{w}_{a,u}), (\mathbf{x}_b, v, \mathbf{w}_{b,v})) = \\ &\quad g'((\mathbf{x}_a, p_{a,u}, q_{a,u}), (\mathbf{x}_b, p_{b,v}, q_{b,v}))\delta(q_{a,u} - p_{a,u}, L_a^\star)\delta(q_{b,v} - p_{b,v}, L_b^\star)\delta(s_{a,u}, S_a^\star)\delta(s_{b,v}, S_b^\star). \end{aligned}$$

Here δ is the indicator function. In this chapter, we provide a complete list of features (f' and g') used for protein structure prediction.

As described in the thesis, four different types of protein folds are examined to verify the effectiveness of the conditional graphical models, including right-handed β-helix, leucine-rich repeats (LLR), triple β-spirals and double-barrel trimer. The features useful to predict these protein folds can be summarized as two types: *common features*, which can be used for all kinds of fold recognition, and *signal features*, which are unique to the target fold but require domain expertise. Our experiments and studies show that the signal features are usually the most discriminative of the target fold and given higher weights in the learnt model. On the other hand, it is time-consuming to get those signal features: generally it takes years for the biologists to accumulate the required knowledge. Sometimes, the current understanding of the target fold (e.g. the double-barrel trimer) is not enough to summarize any reliable signal patterns, in which case the common features could be a reasonable backup.

C.1 Common Features

The common node features we defined to predict all folds include:

1. **Secondary structure prediction scores** Secondary structures reveal significant information on how a protein folds in three dimension. The state-of-art prediction method can achieve an average accuracy of 76 - 78% on soluble proteins. We can get fairly good prediction on α-helix and coils, which can help us locate many structural components. Therefore we define the feature as the averaged secondary structure score:

   $$f'_{avgH}(\mathbf{x}, q_i, p_i) = \frac{1}{q_i - p_i + 1} \sum_{t=p_i}^{q_i} \text{PSIpred-score}(\mathbf{x}, t, \text{H}),$$

   $$f'_{avgE}(\mathbf{x}, q_i, p_i) = \frac{1}{q_i - p_i + 1} \sum_{t=p_i}^{q_i} \text{PSIpred-score}(\mathbf{x}, t, \text{E}),$$

   and

   $$f'_{avgC}(\mathbf{x}, q_i, p_i) = \frac{1}{q_i - p_i + 1} \sum_{t=p_i}^{q_i} \text{PSIpred-score}(\mathbf{x}, t, \text{C}),$$

   where PSIpred-score$(\mathbf{x}, t, \text{Y})$ is the probability that the t-th residue belongs to type Y predicted by PSIPRED (Jones, 1999). $Y \in \{\text{H, E, C}\}$, which represent α-helix, β-sheet and coil respectively.

   We can also define the feature of maximal secondary structure score as

   $$f'_{maxH}(\mathbf{x}, q_i, p_i) = \max_{t=p_i}^{q_i} \text{PSIpred-score}(\mathbf{x}, t, \text{H}),$$

and minimal secondary structure score as

$$f'_{minH}(\mathbf{x}, q_i, p_i) = \min_{t=p_i}^{q_i} \text{PSIpred-score}(\mathbf{x}, t, \text{H}).$$

Similarly, we can derive the definitions for f_{maxE}, f_{maxC}, f_{minE} and f_{minC}.

2. **Segment length** In many cases, each state has strong preferences to a specific range of the segment length, i.e. the number of residues. Therefore we can define the length feature as

$$f_l(\mathbf{x}, p_i, q_i) = q_i - p_i + 1.$$

Notice that the length information has already been modeled via the indicator function in eq(C.1). Here we duplicate the information only for the sake of completeness.

3. **Physicochemical properties** For some motifs or folds, the physicochemical patterns of member residues are unique to themselves or play an important role in the stability of the structures. Therefore we define the features using Kyte-Doolittle hydrophobicity score, solvent accessibility and ionizable score [1]. Similar to the secondary structure score, we also develop the averaged, maximal and minimal versions for each type of physicochemical properties. The feature functions can easily be derived accordingly, i.e.

$$\begin{aligned}
f'_{avgHydro}(\mathbf{x}, q_i, p_i) &= \frac{1}{q_i - p_i + 1} \sum_{t=p_i}^{q_i} \text{KD-score}(\mathbf{x}, t), \\
f'_{maxHydro}(\mathbf{x}, q_i, p_i) &= \max_{t=p_i}^{q_i} \text{KD-score}(\mathbf{x}, t), \\
f'_{minHydro}(\mathbf{x}, q_i, p_i) &= \min_{t=p_i}^{q_i} \text{KD-score}(\mathbf{x}, t).
\end{aligned}$$

where KD-score is the Kyte-Doolittle hydrophobicity score for the t-th amino acid.

The pairwise features we found useful for β-sheet related motifs or folds include:

[1] The score tables of these properties can be accessed at http://www.cgl.ucsf.edu/chimera/1.2065/docs/UsersGuide/midas/hydrophob.html, http://prowl.rockefeller.edu/aainfo/access.htm.

1. **Side chain alignment scores** For β-sheets, it is observed that the amino acids have different propensities to form a hydrogen bond depending on whether the side-chains are buried and exposed (Bradley et al., 2002). An alignment score of interacting residue pairs can be computed accordingly. In the methods described in (Bradley et al., 2002), the conditional probability that a residue of type X will align with residue Y, given their orientation relative to the core (buried or exposed), is estimated from a β-structure database developed from the PDB database. The feature function $g'_{SAS}((\mathbf{x}_a, p_{a,u}, q_{a,u}), (\mathbf{x}_b, p_{b,v}, q_{b,v}))$ can be defined as

$$g'_{SAS}((\mathbf{x}_a, p_{a,u}, q_{a,u}), (\mathbf{x}_b, p_{b,v}, q_{b,v})) = \\ \delta(q_{a,u} - p_{a,u}, q_{b,v} - p_{b,v}) * \sum_{t=0}^{q_{a,u}-p_{a,u}} \text{SAS-score}((\mathbf{x}_a, p_{a,u} + t), (\mathbf{x}_b, p_{b,v} + t)),$$

   where ASA-score$((\mathbf{x}_a, i), (\mathbf{x}_b, j))$ equals to Inward-score$(x_{a,i}, x_{b,j})$ if the side chains of two residues are pointing inward and 0.5∗Outward-score$(x_{a,i}, x_{b,j})$ if the side chains pointing outwards (the side-chain orientation is determined beforehand via domain knowledge).

2. **Parallel β-sheet alignment scores** In addition to the side chain position, another aspect is the different preferences of each amino acid to form parallel and anti-parallel β-sheets. Steward & Thornton derived the "pairwise information values" (V) for a residue of type X given the residue Y on the pairing parallel (or anti-parallel) strand and the offsets of Y from the paired residue Y' of X (Steward & Thornton, 2002). The alignment score for two segments $x = X_1 \dots X_m$ and $y = Y_1 \dots Y_m$ is defined as

$$\text{para-score}(x, y) = \sum_i \sum_j (V(X_i|Y_j, i-j) + V(Y_i|X_j, i-j)).$$

   Notice that this score also takes into account the effect of neighboring residues on the paired strand. We define the feature function as:

$$g'_{PAS}((\mathbf{x}_a, p_{a,u}, q_{a,u}), (\mathbf{x}_b, p_{b,v}, q_{b,v})) = \\ \delta(q_{a,u} - p_{a,u}, q_{b,v} - p_{b,v}) * \sum_{t=0}^{q_{a,u}-p_{a,u}} para - score((\mathbf{x}_a, p_{a,u} + t), (\mathbf{x}_b, p_{b,v} + t)).$$

3. **Distance between interacting pairs** It is observed that the distance between the interacting pairs of segments can not exceed a specific range since otherwise the long insertions will break the structure stabilities. To enforce such constraints, we define feature function $g'_{Dis}((\mathbf{x}, p_u, q_u), (\mathbf{x}, p_v, q_v)) = 1$ if $|p_v - q_u|$ falls in some range, and 0 otherwise.

C.2 Signal Features for β-Helix

From previous literature on the right-handed β-helix, there are two observations important for accurate prediction: 1) the β-strands of each rung have patterns of pleating and hydrogen bonding that are well conserved across the superfamily; 2) the interaction of the strand side-chains in the buried core are critical determinants of the fold (Yoder & Jurnak, 1995; Kreisberg et al., 2000). To better capture these structural properties, we extract the following node features:

1. **Regular expression template** Based on the side-chain alternating patterns in B2-T2-B3 region, BetaWrap generates a regular expression template to detect β-helices, i.e. $\Phi X \Phi X X \Psi X \Phi X$, where Φ matches any of the hydrophobic residues as {A, F, I, L, M, V, W, Y}, Ψ matches any amino acids except ionizable residues as {D, E, R, K} and X matches any amino acid (Bradley et al., 2001). Following similar idea, we define the feature function $f'_{RST}(\mathbf{x}, i, \mathbf{w}_i)$ equal to 1 if the segment $\mathbf{w}_i$ matches the template, and 0 otherwise.

2. **Probabilistic HMM profiles** The regular expression template as above is straightforward and easy to implement. However, sometimes it is hard to make a clear distinction between a true motif and a false alarm. Therefore we built a probabilistic motif profile using HMMER (Durbin et al., 1998) for the s-B23 and s-B1 segments respectively. We define the feature function $f'_{HMM1}(\mathbf{x}, i, \mathbf{w}_i)$ and $f'_{HMM2}(\mathbf{x}, i, \mathbf{w}_i)$ as the alignment scores of segment $\mathbf{w}_i$ against the s-B23 and s-B1 profiles.

3. **Segment length** It is interesting to notice that the β-helix structure has strong preferences for insertions within certain length ranges. Figure C.1 shows the histogram plots of the segment length for state s-T1 and s-T3 respectively. To take into consideration the different preferences of lengths, we did parametric density estimation, a classical method to model the distribution of a random variable. We explored

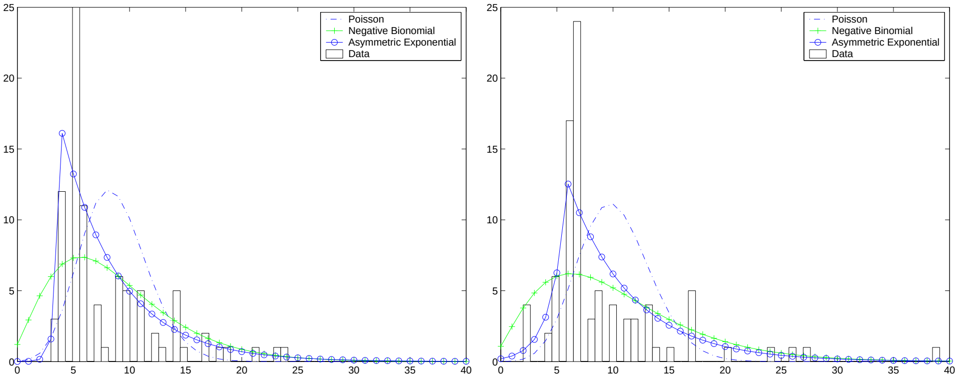


Figure C.1: Histograms for the length of s-T1 (top) and s-T3 (bottom)

several common functions, including Poisson distribution, negative-binomial distribution and asymmetric exponential distribution, which consists for two exponential functions meeting at one point. From the figure, we can see that the asymmetric exponential model is a better estimator than the other two. Therefore we define the feature function $f'_{L1}(\mathbf{x}, i, \mathbf{w}_i) = P(L_{T_1} = q_i - p_i)$ and $f'_{L3}(\mathbf{x}, i, \mathbf{w}_i) = P(L_{T_3} = q_i - p_i)$, where the distribution is estimated via the asymmetric exponential model.

C.3 Signal Features for Leucine-rich Repeats

The leucine-rich repeats are solenoid-like regular arrangement of β-strand and α-helix, connected by coils. The LLR fold is relatively easy to detect due to its conserved motif with many leucines in the sequence and short insertions. We define the following node features:

1. **Regular expression template** The template to identify the LLR is XXXLXXLX[LV]XXXXX, where X matches any amino acid. We define the feature function $f_{RST}(\mathbf{x}, i, \mathbf{w}_i)$ equal to 1 if the subsequence corresponding to w_i matches the template, and 0 otherwise.

2. **Probabilistic HMM profiles** Similar to the β-helix, we also built a probabilistic motif profile using HMMER for the $\beta - \alpha$ segment. We define the feature function $f'_{HMM}(\mathbf{x}, i, \mathbf{w}_i)$ as the alignment scores of w_i against the profiles.

C.4 Signal Features for Triple-β Spirals

In general, the common features of quaternary fold recognition are similar to those for tertiary folds. Some features, such as hydrophobicity and iconic propensity, seem to get higher weights since the quaternary complexes usually form a hydrophobic core. It is quite hard to predict the triple-β spiral fold given the very limited number of positive examples. Fortunately, there exists some identifiable sequence repeat patterns for both B1 and B2 states, which greatly helps to boost the prediction accuracy. We use the regular expression template and profile hidden Markov model to capture those patterns:

1. **Regular expression template** Based on the alternating patterns of conserved hydrophobic core and peripheral patches in the B1 and B2 strands, we define the following regular expression templates: $\mathrm{X}\Upsilon\Phi\mathrm{X}\Psi\mathrm{XX}$ for B1 strand and $\mathrm{XX}\Phi\mathrm{X}\Phi\mathrm{X}\Psi\mathrm{X}$ for B2 strand, where Υ is the conserved tight turn that only matches residues in {P, G, A, F, S, L}, Φ is the hydrophobic core that matches any amino acid in {L, I, M, V, T, S, F, A}, Φ is the peripheral patches which matches any amino acid *except* {C, E, H, P, Q, W}, and X can match any amino acid. We define the feature function $f_{RST}(\mathbf{x}, i, \mathbf{w}_i)$ equal to 1 if the segment matches the template, and 0 otherwise.

2. **Probabilistic HMM profiles** Sometimes the regular expression template is not preferred since it is hard to make a clear cutoff between a true motif and a false alarm. Therefore profile HMM using probabilistic estimation is a better resort. Initially we used the alignments of all the positive examples for B1 and B2 state, but fail to get reasonable results as expected since the sequence similarity is too low to generalize a good profile. Later we observe that the alignments share more similar patterns in sequence if we separate the alignments into groups based on the type of amino acid on conserved β-turn position, that is, position 'j' in Green's labeling scheme (see Figure 8.3). Therefore we built six HMM profiles (one for each amino acid type at position 'j') using HMMER (Durbin et al., 1998) for B1 and B2 respectively. Then we define the feature functions $f_{HMM}(\mathbf{x}, i, \mathbf{w}_i)$ as the alignment scores of the segment against those B1 and B2 profiles.

C.5 Signal Features for Double-barrel Trimer

The double-barrel trimer is a relatively new protein fold which attracts biologists' attention recently, due to their common existence in the coat proteins of viruses infecting different kinds of species. It is claimed that the layouts of the β-barrels are quite unique to virus proteins, but there is no significant sequence conservation either in the β-strand components or the loops or turns connecting the β-strands. The only interesting observation we made after careful study is this short β-turns between strand E and F. It has strong structural conservations without sequence similarities. Therefore we define β-turn features as follows:

1. **β-turn scores** There has been extensive research on how to reliably predict the β-turns in the protein sequence. Up to now, the commonly accepted nomenclature divides the β-turns into six types, i.e. type I, II, VIII, I', II', VIa, VIb, and IV, as defined by Hutchinson and Thornton (Hutchinson & Thornton, 1994). In (Fuchs & Alix, 2005), the propensity scores of different amino acids in those six type of β-turns are calculated. In particular, the experiments show that a weighted propensity score using the PSI-BLAST profile performs much better than using the amino acid type that only appear in the protein sequences. Therefore we define the feature function $f_{\beta-turn}(\mathbf{x}, i, \mathbf{w}_i)$ as the maximal (and minimal) score of the β-turn propensity of each type over the subsequence d_i to d_{i+1}.

2. **Maximal alignment scores** The pairwise features of β-sheet alignment scores are defined similarly as described above except that the lengths of the β-strand pair is not necessarily the same. This causes a problem when we try to compute the alignment score since we do not know which pairs of residues interact with each other. To solve the problem, we compute all possible alignments by shifting the starting position of the longer segment and use the highest alignment scores as the features:

$$g'_{MAS}((\mathbf{x}_a, p_{a,u}, q_{a,u}), (\mathbf{x}_b, p_{b,v}, q_{b,v})) =$$
$$\max_{t'=p_{b,v}}^{p_{b,v}+q_{a,u}-p_{a,u}} \sum_{t=0}^{q_{a,u}-p_{a,u}} \text{SAS-score}((\mathbf{x}_a, p_{a,u}+t), (\mathbf{x}_b, t'+t)),$$

3. **Pointwise alignment scores** Another challenges in predicting the double-barrel trimer is the incomplete understanding of the inter-chain

interactions. It is suggested that the interactions happen within the FG-loop of the two β-barrels, but the specific location as well as the type of chemical bonding remains unclear. Following the idea of natural selection of hydrogen bonds, we compute all the possible pairs of side-chain interactions, and use the highest score as features.

$$g'_{PTAS}((\mathbf{x}_a, p_{a,u}, q_{a,u}), (\mathbf{x}_b, p_{b,v}, q_{b,v})) = \max_{t'=p_{b,v}}^{p_{b,v}} \max_{t=p_{a,u}}^{q_{a,u}} \text{SAS-score}((\mathbf{x}_a, t), (\mathbf{x}_b, t')),$$

In other words, we try to capture the possibility of forming hydrogen bonds between the current pairs of segments.

It is interesting to notice that most of the features defined above are quite general, not limited to predicting the four protein folds only. For example, an important aspect to discriminate a specific protein fold with others is to build HMM profiles or identify regular expression templates for conserved regions if they exist; the secondary structure assignments are essential in locating the elements within a protein fold; if some segments have strong preferences for certain length range, then the lengths are also informative. For pairwise features, the β-sheet alignment scores are useful for folds in β-family while hydrophobicity is important for α- or $\alpha\beta$-family.

Bibliography

Altschul, S., Gish, W., Miller, W., Myers, E., & Lipman, D. (1990). Basic local alignment search tool. *J Mol Biol.*, *215*, 403–10.

Altschul, S., Madden, T., Schaffer, A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. (1997). Gapped BLAST and PSI-blast: a new generation of protein database search programs. *Nucleic Acids Res.*, *25*, 3389–402.

Altun, Y., Hofmann, T., & Smola, A. J. (2004). Gaussian process classification for segmenting and annotating sequences. *ICML '04: Twenty-first international conference on Machine learning.*

Andrieu, C., de Freitas, N., & Doucet, A. (2000). Reversible jump mcmc simulated annealing for neural networks. *Proceedings of UAI-00.*

Bailey, T. L., & Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc. of ISMB'94.*

Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L. L., Studholme, D. J., Yeats, C., & Eddy, S. R. (2004). The pfam protein families database. *Nucleic Acids Research*, *32*, 138–141.

Baxter, J. (2000). A model of inductive bias learning. *Journal of Artificial Intelligence Research*, *12*, 149–198.

Benson, S., Bamford, J., Bamford, D., & Burnett, R. (2004). Does common architecture reveal a viral lineage spanning all three domains of life? *Mol Cell.*, *16*, 673–85.

Berman, H., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., Shindyalov, I., & Bourne, P. (2000). The protein data bank. *Nucleic Acids Research*, *28*, 235–42.

Bockhorst, J., & Craven, M. (2005). Markov networks for detecting overlapping elements in sequence data. *NIPS'05*.

Bourne, P. E., & Weissig, H. (2003). *Structural bioinformatics: Methods of biochemical analysis*. Wiley-Liss.

Boys, R. J., & Henderson, D. A. (2001). A comparison of reversible jump mcmc algorithms for dna sequence segmentation using hidden markov models. *Comp. Sci. and Statist.*, *33*, 35–49.

Bradley, P., Cowen, L., Menke, M., King, J., & Berger, B. (2001). Predicting the beta-helix fold from protein sequence data. *Proceedings of RECOMB'01*.

Bradley, P., Kim, P. S., & Berger, B. (2002). Trilogy: discovery of sequence-structure patterns across diverse proteins. *Proceedings of RECOMB'02*.

Breiman, L., & J, F. (1997). Predicting multivariate responses in multiple linear regression. *J. Royal Stat. Society B*, *59*, 3–37.

Buntine, W. L. (1995). Chain graphs for learning. *Uncertainty in Artificial Intelligence* (pp. 46–54).

Bystroff, C., Thorsson, V., & Baker, D. (2000). HMMSTR: a hidden markov model for local sequence-structure correlations in proteins. *J Mol Biol.*, *301*, 173–90.

Chen, H., & Zhou, H. (2005). Prediction of interface residues in protein-protein complexes by a consensus neural network method: test against nmr data. *Proteins.*, *61*, 21–35.

Cheng, J., & Baldi, P. (2006). A machine learning information retrieval approach to protein fold recognition. *Bioinformatics*, *22*, 1456–63.

Chou, K., & Cai, Y. (2003). Predicting protein quaternary structure by pseudo amino acid composition. *Proteins.*, *53*, 282–9.

Chu, W., Ghahramani, Z., & Wild, D. L. (2004). A graphical model for protein secondary structure prediction. *Proc.of ICML'04*.

Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines*. Cambridge University Press.

Crooks, G., & Brenner, S. (2004). Protein secondary structure: entropy, correlations and prediction. *Bioinformatics.*, *20*, 1603–1611.

Cuff, J., & Barton, G. (1999). Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. *Proteins.*, *34*, 508–519.

Cuff, J., & Barton, G. (2000). Application of multiple sequence alignment profiles to improve protein secondary structure prediction. *Proteins.*, *40*, 502–511.

Dahiyat, B., & Mayo, S. (1997). De novo protein design: fully automated sequence selection. *Science.*, *278*, 82–7.

Delcher, A., Kasif, S., Goldberg, H., & Xsu, W. (1993). Protein secondary-structure modeling with probabilistic networks. *Proceedings of ISMB'93.*

Desmet, J., de Maeyer, M., Hazes, B., & Lasters, I. (1992). The dead-end elimination theorem and its use in protein side-chain positioning. *Nature.*, *356*, 539–542.

Dietterich, T., Ashenfelter, A., & Bulatov, Y. (2004). Training conditional random fields via gradient tree boosting.

Ding, C. H., & Dubchak, I. (2000). Multi-class protein fold recognition using support vector machines and neural networks . *Bioinformatics*, *17*, 349–358.

Do, C. B., Gross, S. S., & Batzoglou, S. (2006a). Contralign: Discriminative training for protein sequence alignment.

Do, C. B., Woods, D. A., & Batzoglou, S. (2006b). Contrafold: Rna secondary structure prediction without physics-based models.

Durbin, R., Eddy, S., Krogh, A., & Mitchison, G. (1998). *Biological sequence analysis: probabilistic models of proteins and nucleic acids.* Cambridge University Press.

Evgeniou, T., Micchelli, C. A., & Pontil, M. (2005). Learning multiple tasks with kernel methods. *J. Mach. Learn. Res.*, *6*, 615–637.

Fuchs, P., & Alix, A. (2005). High accuracy prediction of beta-turns and their types using propensities and multiple alignments. *Proteins.*, *59*, 828–39.

Garian, R. (2001). Prediction of quaternary structure from primary structure. *Bioinformatics.*, *17*, 551–6.

Ghosn, J., & Bengio, Y. (2000). Bias learning, knowledge sharing. *ijcnn*, *01*, 1009.

Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, *82*, 711–732.

Guda, C., Lu, S., Sheeff, E., Bourne, P., & Shindyalov, I. (2004). CE-MC: A multiple protein structure alignment server. *Nucleic Acids Res.*, *In press.*

Guermeur, Y., Geourjon, C., Gallinari, P., & Deleage, G. (1999). Improved performance in protein secondary structure prediction by inhomogeneous score combination. *Bioinformatics.*, *15*, 413–421.

Guo, J., Chen, H., Sun, Z., & Lin, Y. (2004). A novel method for protein secondary structure prediction using dual-layer svm and profiles. *Proteins.*, *54*, 738–743.

Hammersley, J., & Clifford, P. (1971). *Markov fields on finite graphs and lattices.* Unpublished manuscript.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2001). *The elements of statistical learning: data mining, inference, and prediction.* New York: Springer-Verlag.

He, X., Zemel, R., & Carreira-Perpinan, M. (2004). Multiscale conditional random fields for image labelling.

Heger, A., & Holm, L. (2000). Rapid automatic detection and alignment of repeats in protein sequences. *Proteins*, *41*, 224–237.

Heskes, T. (2000). Empirical bayes for learning to learn. *Proc. 17th International Conf. on Machine Learning* (pp. 367–374). Morgan Kaufmann, San Francisco, CA.

Hua, S., & Sun, Z. (2001). A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. *J Mol Biol.*, *308*, 397–407.

Huelsenbeck, J., Larget, B., & Alfaro, M. (2004). Bayesian phylogenetic model selection using reversible jump markov chain monte carlo. *Mol Biol. Evol.*, *6*, 1123–33.

Hutchinson, E., & Thornton, J. (1994). A revised set of potentials for beta-turn formation in proteins. *Protein Sci.*, *3*, 2207–16.

Inbar, Y., Benyamini, H., Nussinov, R., & Wolfson, H. (2005). Prediction of multimolecular assemblies by multiple docking. *J Mol Biol.*, *349*, 435–47.

Jaakkola, T. (2000). Tutorial on variational approximation methods.

Jones, D. (1999). Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol.*, *292*, 195–202.

Jones, D., Taylor, W., & Thornton, J. (1992). A new approach to protein fold recognition. *Nature.*, *358*, 86–9.

Jordan, M. I., Ghahramani, Z., Jaakkola, T., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*, 183–233.

Kabsch, W., & Sander, C. (1983). Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, *22*, 2577–2637.

Karplus, K., Barrett, C., & Hughey, R. (1998). Hidden markov models for detecting remote protein homologies. *Bioinformatics*, *14*, 846–56.

Kim, H., & Park, H. (2003). Protein secondary structure prediction based on an improved support vector machines approach. *Protein Eng.*, *16*, 553–60.

Kim, W., & Ison, J. (2005). Survey of the geometric association of domain-domain interfaces. *Proteins.*, *61*, 1075–88.

King, R., & Sternberg, M. (1996). Identification and application of the concepts important for accurate and reliable protein secondary structure prediction. *Protein Sci.*, *5*, 2298–2310.

Kobe, B., & Deisenhofer, J. (1994). The leucine-rich repeat: a versatile binding motif. *Trends Biochem Sci.*, *10*, 415–21.

Kreisberg, J., Betts, S., & King, J. (2000). Beta-helix core packing within the triple-stranded oligomerization domain of the p22 tailspike. *Protein Sci.*, *9*, 2338–43.

Krogh, A., Brown, M., Mian, I. S., Sjolander, K., & Haussler., D. (1994). Hidden markov models in computational biology: Applications to protein modeling. *J Mol Biol.*, *235*, 1501–31.

Kumar, S., & Hebert, M. (2003). Discriminative random fields: A discriminative framework for contextual interaction in classification. *Proc. of ICCV'03* (pp. 1150–1159).

Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proc. of ICML'01*.

Lafferty, J., Zhu, X., & Liu, Y. (2004). Kernel conditional random fields: representation and clique selection. *Proc.of International Conference on Machine Learning (ICML-04)*.

Lauritzen, S., & Wermuth, N. (1989). Graphical models for associations between variables, some of which are qualitative and some quantitative. *Annals of Statistics*, *17*, 31–57.

Leinonen, R., Diez, F., Binns, D., Fleischmann, W., Lopez, R., & Apweiler, R. (2004). Uniprot archive. *Bioinformatics.*, *20*, 3236–7.

Liu, Y., Carbonell, J., Klein-Seetharaman, J., & Gopalakrishnan, V. (2004). Comparison of probabilistic combination methods for protein secondary structure prediction. *Bioinformatics.*, *20*, 3099–107.

Liu, Y., Carbonell, J., Weigele, P., & Gopalakrishnan, V. (2005). Segmentation conditional random fields (SCRFs): A new approach for protein fold recognition. *Proceedings of RECOMB'05*.

Mamitsuka, H., & Abe, N. (1994). Predicting location and structure of beta-sheet regions using stochastic tree grammars. *Proc Int Conf Intell Syst Mol Biol.*, 276–84.

McCallum, A., Freitag, D., & Pereira, F. C. N. (2000). Maximum entropy markov models for information extraction and segmentation. *Proc.of International Conference on Machine Learning (ICML-00)* (pp. 591–598).

Meiler, J., & Baker, D. (2003). Coupled prediction of protein secondary and tertiary structure. *Proc Natl Acad Sci U S A.*, *100*, 12105–12110.

Minka, T. P. (2001). Algorithms for maximum-likelihood logistic regression. *CMU Statistics Tech Report 758*.

Murphy, K. P., Weiss, Y., & Jordan, M. I. (1999a). Loopy belief propagation for approximate inference: An empirical study. *In Proceedings of Uncertainty in AI* (pp. 467–475).

Murphy, K. P., Weiss, Y., & Jordan, M. I. (1999b). Loopy belief propagation for approximate inference: An empirical study. *UAI'99* (pp. 467–475).

Murray, I., & Ghahramani, Z. (2004). Bayesian learning in undirected graphical models: Approximate mcmc algorithms. *Proceedings of UAI-04* (pp. 392–399).

Murzin, A., Brenner, S., Hubbard, T., & Chothia, C. (1995). SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol.*, *247*, 536–40.

Ng, A., & Jordan, M. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in Neural Information Processing Systems 14*.

Orengo, C., Michie, A., Jones, S., Jones, D., Swindells, M., & Thornton, J. (1997). CATH–a hierarchic classification of protein domain structures. *Structure.*, *5*, 1093–108.

Ouali, M., & King, R. (2000). Cascaded multiple classifiers for secondary structure prediction. *Protein Sci.*, *9*, 1162–1176.

Pinto, D., McCallum, A., Wei, X., & Croft, W. B. (2003). Table extraction using conditional random fields. *Proceedings of the 26th ACM SIGIR conference* (pp. 235–242).

Pollastri, G., Przybylski, D., Rost, B., & Baldi, P. (2002). Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. *Proteins.*, *47*, 228–35.

Qi, Y., Szummer, M., & Minka, T. P. (2005). Bayesian conditional random fields.

Quattoni, A., Collins, M., & Darrel, T. (2005). Conditional random fields for object recognition.

Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*, 257–286.

Rost, B. (2001). Review: protein secondary structure prediction continues to rise. *J Struct Biol.*, *134*, 204–218.

Rost, B., & Sander, C. (1993). Prediction of protein secondary structure at better than 70% accuracy. *J Mol Biol.*, *232*, 584–599.

Rost, B., Sander, C., & Schneider, R. (1994). Redefining the goals of protein secondary structure prediction. *J Mol Biol.*, *235*, 13–26.

Roth, D., & Yih, W. (2005). Integer linear programming inference for conditional random fields. *ICML '05: Proceedings of the 22nd international conference on Machine learning* (pp. 736–743). New York, NY, USA: ACM Press.

Salamov, A., & Solovyev, V. (1995). Prediction of protein secondary structure by combining nearest-neighbor algorithms and multiple sequence alignments. *J Mol Biol.*, *247*, 11–15.

Sarawagi, S., & Cohen, W. W. (2004). Semi-markov conditional random fields for information extraction. *Proc. of NIPS'2004*.

Scanlon, E. L. (2004). Predicting the triple beta-spiral fold from primary sequence data. *Master Thesis, Massachusetts Institute of Technology*.

Schmidler, S., Liu, J., & Brutlag, D. (2000). Bayesian segmentation of protein secondary structure. *Journal of computational biology*, *7*, 233–48.

Selbig, J., Mevissen, T., & Lengauer, T. (1999). Decision tree-based formation of consensus protein secondary structure prediction. *Bioinformatics.*, *15*, 1039–1046.

Sha, F., & Pereira, F. (2003). Shallow parsing with conditional random fields. *Proceedings of Human Language Technology, NAACL 2003*.

Sminchisescu, C., Kanaujia, A., Li, Z., & Metaxas, D. (2005). Conditional models for contextual human motion recognition.

Smith, N. A., & Eisner, J. (2005). Contrastive estimation: Training log-linear models on unlabeled data. *ACL*.

Steward, R., & Thornton, J. (2002). Prediction of strand pairing in antiparallel and parallel beta-sheets using information theory. *Proteins.*, *48*, 178–91.

Sutton, C., & McCallum, A. (2006). An introduction to conditional random fields for relational learning. *Introduction to Statistical Relational Learning*.

Sutton, C. A., Rohanimanesh, K., & McCallum, A. (2004). Dynamic conditional random fields: factorized probabilistic models for labeling and segmenting sequence data. *ICML*.

Taskar, B. (2004). Learning structured prediction models: A large margin approach. Master's thesis, Stanford University.

Taskar, B., Guestrin, C., & Koller, D. (2003). Max-margin markov networks. *Proc. of NIPS'03*.

Taskar, B., & Simon Lacoste-Julien, M. J. (2005). Structured prediction via the extragradient method. *In Proceedings of NIPS*.

Teh, Y., Seeger, M., & Jordan, M. (2005). Semiparametric latent factor models.

Thompson, J., Higgins, D., & Gibson, T. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, *22*, 4673–80.

Torralba, A., Murphy, K. P., & Freeman, W. T. (2004). Contextual models for object detection using boosted random fields.

Tsochantaridis, I., Hofmann, T., Joachims, T., & Altun, Y. (2004). Support vector machine learning for interdependent and structured output spaces.

van Raaij, M., Mitraki, A., Lavigne, G., & Cusack, S. (1999). A triple beta-spiral in the adenovirus fibre shaft reveals a new structural motif for a fibrous protein. *Nature.*, *401*, 935–8.

Vapnik, V. (1995). *The nature of statistical learning theory*. New York: Springer-Verlag.

Venclovas, C., Zemla, A., Fidelis, K., & Moult, J. (2003). Assessment of progress over the casp experiments. *Proteins.*, *53*, 585–95.

Vishwanathan, S. V. N., Schraudolph, N. N., Schmidt, M. W., & Murphy, K. P. (2006). Accelerated training of conditional random fields with stochastic gradient methods. *ICML '06* (pp. 969–976).

Wallach, H. (2002). Efficient training of conditional random fields. Master's thesis, University of Edinburgh.

Wang, S. B., Quattoni, A., Morency, L.-P., & Demirdjian, D. (2006). Hidden conditional random fields for gesture recognition. *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 1521–1527). Washington, DC, USA: IEEE Computer Society.

Weigele, P. R., Scanlon, E., & King, J. (2003). Homotrimeric, β-stranded viral adhesins and tail proteins. *J Bacteriol.*, *185*, 4022–30.

Welling, M., & Hinton, G. E. (2002). A new learning algorithm for mean field boltzmann machines. *ICANN '02: Proceedings of the International Conference on Artificial Neural Networks* (pp. 351–357). London, UK: Springer-Verlag.

Westhead, D., Slidel, T., Flores, T., & Thornton, J. (1999). Protein structural topology: Automated analysis and diagrammatic representation. *Protein Sci.*, *8*, 897–904.

Williams, C. K. I., & Barber, D. (1998). Bayesian classification with gaussian processes. *IEEE Trans. Pattern Anal. Mach. Intell.*, *20*, 1342–1351.

Winn, J., & Shotton, J. (2006). The layout consistent random field for recognizing and segmenting partially occluded objects.

Xing, E., Jordan, M., & Russell, S. (2003). A generalized mean field algorithm for variational inference in exponential families. *Uncertainty in Artificial Intelligence (UAI2003).* Morgan Kaufmann Publishers.

Yanover, C., & Weiss, Y. (2002). Approximate inference and protein-folding. *Neural Information Processing Systems (NIPS'02).*

Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2000). Generalized belief propagation. *NIPS'00* (pp. 689–695).

Yoder, M., & Jurnak, F. (1995). Protein motifs. 3. the parallel beta helix and other coiled folds. *FASEB J.*, *9*, 335–42.

Yoder, M., Keen, N., & Jurnak, F. (1993). New domain motif: the structure of pectate lyase c, a secreted plant virulence factor. *Science*, *260*, 1503–7.

Yu, K., Tresp, V., & Schwaighofer, A. (2005). Learning gaussian processes from multiple tasks. *ICML '05: Proceedings of the 22nd international conference on Machine learning* (pp. 1012–1019). New York, NY, USA: ACM Press.

Zemla, A., Venclovas, C., K, K. F., & Rost, B. (1999). A modified definition of sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins.*, *34*, 220–223.

Zhang, S., Pan, Q., Zhang, H., Zhang, Y., & Wang, H. (2003). Classification of protein quaternary structure with support vector machine. *Bioinformatics.*, *19*, 2390–6.

Zhu, J., & Hastie, T. (2001). Kernel logistic regression and the import vector machine. *NIPS* (pp. 1081–1088).

Zhu, J., Nie, Z., Wen, J.-R., Zhang, B., & Ma, W.-Y. (2005). 2d conditional random fields for web information extraction. *ICML '05: Proceedings of the 22nd international conference on Machine learning* (pp. 1044–1051).