Chapter 1
Introduction to Sequence Analysis for Human Behavior Understanding

Hugues Salamin and Alessandro Vinciarelli

1.1 Introduction

Human sciences recognize sequence analysis as a key aspect of any serious attempt of understanding human behavior [1]. While recognizing that nonsequential analysis can provide important insights, the literature still observes that taking into account sequential aspects "*provide[s] an additional level of information about whatever behavior we are observing, a level that is not accessible to nonsequential analyses.*" [2]. The emphasis on sequential aspects is even higher when it comes to domains related to social interactions like, e.g., Conversation Analysis: "[...] *it is through the knowledge of the place of an action in a sequence that one reaches an understanding of what the action was (or turned out to be).*" [4]. Furthermore, social interactions are typically defined as "*sequences of social actions*" in the cognitive psychology literature [21].

In parallel, and independently of human sciences, sequence analysis is an important topic in machine learning and pattern recognition [5, 7]. Probabilistic sequential models, i.e. probability distributions defined over sequences of discrete or continuous stochastic variables, have been shown to be effective in a wide range of problems involving sequential information like, e.g., speech and handwriting recognition [6], bioinformatics [3] and, more recently, Social Signal Processing and social behavior understanding [28].

Given a sequence $\mathbf{X} = (x_1, \ldots, x_N)$, where x_t is generally a D-dimensional vector with continuous components, the sequence analysis problem (in machine learning) takes typically two forms: The first is called *classification* and it consists in assigning $\mathbf{X}$ a class c belonging to a predefined set $C = \{c_1, \ldots, c_K\}$. The second is called *la-*

Hugues Salamin
School of Computing Science, University of Glasgow, e-mail: hsalamin@dcs.gla.ac.uk

Alessandro Vinciarelli
School of Computing Science, University of Glasgow and Idiap Research Institute, e-mail: vincia@dcs.gla.ac.uk

beling and it corresponds to mapping $\mathbf{X}$ into a sequence $\mathbf{Z} = (z_1, \ldots, z_N)$ of the same length as $\mathbf{X}$, where each z_t belongs to a discrete set $S = \{s_1, \ldots, s_T\}$. An example of classification in Human Behavior Understanding is the recognition of gestures, where a sequence of hand positions is mapped into a particular gesture (e.g., hand waving) [30]. An example of labeling is role recognition in conversations, where a sequence of turns is mapped into a sequence of roles assigned to the speaker of each turn [24].

In both cases, the problem can be thought of as finding the value $\mathbf{Y}^*$ satisfying the following equation:

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} \mathrm{P}(\mathbf{X}, \mathbf{Y}) \tag{1.1}$$

where $\mathbf{Y}^*$ can be one of the classes belonging to C, or a sequence $\mathbf{Z}$ of the same length as $\mathbf{X}$. In this respect, the main problem is to find a model $\mathrm{P}(\mathbf{X}, \mathbf{Y})$ suitable for the problem at hand, i.e. an actual expression of the probability to be used in the equation above. This chapter adopts the unifying framework of *graphical models* [14] to introduce two of the most common probabilistic sequential models used to estimate $\mathrm{P}(\mathbf{X}, \mathbf{Y})$, namely Bayesian Networks (in particular Markov Models and Hidden Markov Models [10, 23]) and Conditional Random Fields [15, 27].

The chapter focuses in particular on two major aspects of the sequence analysis problem: On one hand, the role that conditional independence assumptions have in making the problem tractable and, on the other hand, the relationship between independence assumptions and the particular factorization that the models mentioned above show. The text provides some details about inference and training as well, including pointers to the relevant literature.

The rest of the chapter is organized as follows: Section 1.2 describes the graphical models framework, Section 1.3 and 1.4 introduce Bayesian Networks and Conditional Random Fields respectively, Section 1.5 propose training and inference methods and Section 1.6 draws some conclusions.

1.2 Graphical Models

The main problem in estimating $\mathrm{P}(\mathbf{X}, \mathbf{Y})$ is that the state spaces of the random variables $\mathbf{X}$ and $\mathbf{Y}$ increase exponentially with the length of $\mathbf{X}$. The resulting challenge is to find a suitable trade-off between two conflicting needs: to use a compact and tractable representation of $\mathrm{P}(\mathbf{X}, \mathbf{Y})$ on one side and to take into account (possibly long-term) time dependencies on the other side. Probability theory offers two main means to tackle the above, the first is to *factorize* the probability distribution, i.e. to express it as a product of factors that involve only part of the random variables in $\mathbf{X}$ and $\mathbf{Y}$ (e.g., only a subsequence of $\mathbf{X}$). In this way, the global problem is broken into small, possibly simpler, problems. The second is to make *independence assumptions* about the random variables, i.e. to make hypotheses about what are the variables that actually influence one another in the problem.

As an example of how factorization and independence assumptions can be effective, consider the simple case where $\mathbf{Y}$ is a sequence of binary variables. By applying the chain rule, it is possible to write the following:

$$\mathrm{P}(Y_1,\ldots,Y_N) = \mathrm{P}(Y_1)\prod_{i=2}^{N}\mathrm{P}(Y_i\,|\,Y_1,\ldots,Y_{i-1}). \tag{1.2}$$

As the number of possible sequences is 2^N, a probability distribution expressed as a table of experimental frequencies (the percentage of times each sequence is observed) requires 2^N-1 parameters.

In this respect, the factorization helps to concentrate on a subset of the variables at a time and maybe to better understand the problem (if there is a good way of selecting the order of the variables), but still it does not help in making the representation more compact, the number of the parameters is the same as before the factorization. In order to decrease the number of parameters, it is necessary to make independence assumptions like, e.g., the following (known as *Markov property*):

$$\mathrm{P}(Y_i\,|\,Y_1,\ldots,Y_{i-1}) = \mathrm{P}(Y_i\,|\,Y_{i-1}). \tag{1.3}$$

The above transforms Equation (1.2) as follows:

$$\mathrm{P}(Y_1,\ldots,Y_N) = \mathrm{P}(Y_1)\prod_{i=2}^{N}\mathrm{P}(Y_i\,|\,Y_{i-1}), \tag{1.4}$$

where the number of parameters is only $2(N-1)+1$, way much less than the original 2^N-1. The number of parameters can be reduced to just 3 if we consider that $\mathrm{P}(Y_i\,|\,Y_{i-1})$ is independent of i, thus it does not change depending on the particular point of the sequence. The combination of factorization and independence assumptions has thus made it possible to reduce the number of parameters and model long sequences with a compact and tractable representation.

Probabilistic graphical models offer a theoretic framework where factorization and independence assumptions are equivalent. Distributions $\mathrm{P}(\mathbf{X},\mathbf{Y})$ are represented with graphs where the nodes correspond to the random variables and the missing edges account for the independence assumptions. More in particular, the graph acts as a filter that, out of all possible $\mathrm{P}(\mathbf{X},\mathbf{Y})$, selects only the set DF of those that *factorize over the graph* (see below what this means depending on the type of graph). In parallel the graph acts as a filter that selects the set DI of those distributions $\mathrm{P}(\mathbf{X},\mathbf{Y})$ that respect the independence assumptions encoded by the graph (see below how to identify such independence assumptions). The main advantage of graphical models is that $DF = DI$, i.e. factorization and independence assumptions are equivalent (see [5] for an extensive description of this point). Furthermore, inference and training techniques developed for a certain type of graph can be extended to all of the distributions encompassed by the same type of graph (see [13] for an extensive account of training techniques in graphical models).

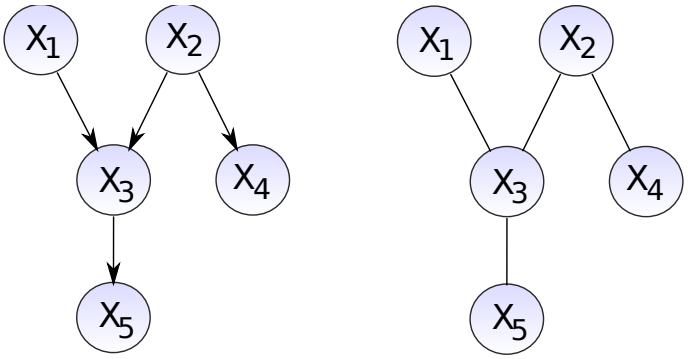

Fig. 1.1 Probabilistic graphical models: each node corresponds to a random variable and the graph represents the joint probability distribution over all of the variables. The edges can be directed (left graph) or undirected (right graph).

The rest of this section introduces notions and terminology that will be used throughout the rest of this chapter.

1.2.1 Graph Theory

The basic data-structure used in the chapter is the graph.

Definition 1. A *graph* is a data structure composed of a set of nodes and a set of edges. Two nodes can be connected by a directed or undirected edge.

We will denote by $G = (\mathbf{N}, \mathbf{E})$ a graph, where $\mathbf{N}$ is the set of nodes and $\mathbf{E}$ is the set of the edges. We write $n_i \rightarrow n_j$ when two nodes are connected by a directed edge and n_i—n_j when they are connected by an undirected one. If there is an edge between n_i and n_j, we say that these are *connected* and we write that $n_i \rightleftharpoons n_j$. An element of $\mathbf{E}$ is denoted with (i, j) meaning that nodes n_i and n_j are connected.

Definition 2. If $n \rightleftharpoons m$, then m is said to be a *neighbour* of n (and vice-versa). The set of all neighbours of n is called the *neighbourhood* and it is denoted by $\mathrm{Nb}(n)$. The set of the *parents* of a node n contains all nodes m such that $m \rightarrow n$. This set is denoted by $\mathrm{Pa}(n)$. Similarly, the set of the *children* of a node n contains all nodes m such that $n \rightarrow m$. This is set is denoted by $\mathrm{Ch}(n)$.

Definition 3. A *path* is a list of nodes $(p_1, \ldots, p_n)$ such that $p_i \rightarrow p_{i+1}$ or p_i—p_{i+1} holds for all i. A *trail* is a list of nodes $(p_1, \ldots, p_n)$ such that $p_i \rightleftharpoons p_{i+1}$ holds for all i.

The difference between a trail and a path is that a trail can contain $p_i \leftarrow p_{i+1}$ edges. In other words, in a trail it is possible to follow a directed edge in the wrong direction. In undirected graphs, there is no difference between paths and trails

Definition 4. A *cycle* is a path $(p_1, \ldots, p_n)$ such that $p_1 = p_n$. A graph is *acyclic* if there are no cycles in it.

1.2.2 Conditional Independence

Consider two random variables X and Y that can take values in $Val(X)$ and $Val(Y)$, respectively.

Definition 5. Two random variables X and Y are *independent*, if and only if $\mathrm{P}(Y|X) = \mathrm{P}(X)\ \forall x \in \mathrm{Val}(X), \forall y \in \mathrm{Val}(Y)$. When X and Y are independent, we write that $\mathrm{P} \models (X \perp Y)$.

The definition can be easily extended to sets of variables $\mathbf{X}$ and $\mathbf{Y}$:

Definition 6. Two sets of random variables $\mathbf{X}$ and $\mathbf{Y}$ are *independent*, if and only if $\mathrm{P}(Y|X) = \mathrm{P}(X)\ \forall X \in \mathrm{Val}(\mathbf{X}), \forall Y \in \mathrm{Val}(\mathbf{Y})$. When $\mathbf{X}$ and $\mathbf{Y}$ are independent, we write that $\mathrm{P} \models (\mathbf{X} \perp \mathbf{Y})$.

Definition 7. Let $\mathbf{X}, \mathbf{Y}$, and $\mathbf{Z}$ be sets of random variables. We say that $\mathbf{X}$ is *conditionally independent* of $\mathbf{Y}$ given $\mathbf{Z}$ if and only if:

$$\mathrm{P}(\mathbf{X}, \mathbf{Y} \,|\, \mathbf{Z}) = \mathrm{P}(\mathbf{X} \,|\, \mathbf{Z})\, \mathrm{P}(\mathbf{Y} \,|\, \mathbf{Z})$$

We write that $\mathrm{P} \models (\mathbf{X} \perp \mathbf{Y} \,|\, \mathbf{Z})$.

The rest of the chapter shows how the notion of conditional independence is more useful, in practice, than the simple independence. Fore example, the Markov property (see above) can be seen as a conditional independence assumption where the future X_{t+1} is conditionally independent of the past $(X_1, \ldots, X_{t-1})$ given the present X_t. Such an assumption might not be true in reality (X_t is likely to be dependent on $X_1, \ldots, X_{t-1}$), but it introduces a simplification that makes the simple model of Equation (1.4) tractable.

1.3 Bayesian Networks

Bayesian Networks [11, 12, 20] are probabilistic graphical models encompassed by Directed Acyclic Graphs (DAGs), i.e. those graphs where the edges are directed and no cycles are allowed. The rest of the section shows how a probability distribution factorizes over a DAG and how the structure of the edges encodes conditional independence assumptions. As factorization and independence assumptions are equivalent for graphical models, it is possible to say that all of the distributions that factorize over a DAG respect the conditional independence assumptions that the DAG encodes. Inference and training approaches will not be presented for directed models because each directed graph can be transformed into an equivalent

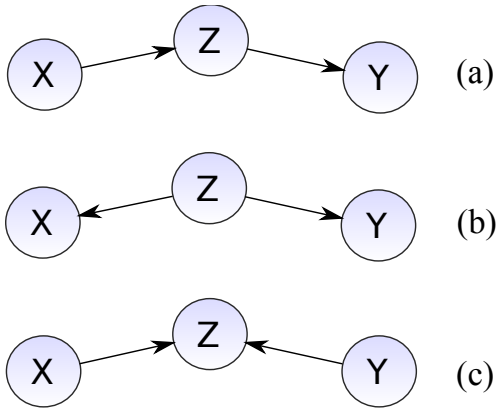

Fig. 1.2 The picture shows the three ways it is possible to pass through a node (Z in this case) along a trail going from X to Y: head-to-tail, tail-to-tail and head-to-head.

undirected one and related inference and training approaches can be applied. The interested reader can refer to [9, 13] for extensive surveys of these aspects.

1.3.1 Factorization

Definition 8. Let $\mathbf{X} = (X_i, \ldots, X_N)$ be a set of random variables and G be a DAG whose node set is $\mathbf{X}$. The probability distribution P over $\mathbf{X}$ is said to *factorize* over G if

$$P(\mathbf{X}) = \prod_{i=1}^{n} \mathrm{P}(X_i \,|\, \mathrm{Pa}(X_i)). \tag{1.5}$$

A pair (G, P) where P factorizes over G is called *Bayesian Network*.

1.3.2 The d-Separation Criterion

A DAG allows one to read conditional independence assumptions through the concept of *d-separation* for directed graphs.

Definition 9. Let (G, P) be a Bayesian Network and $X_1 \rightleftharpoons \ldots \rightleftharpoons X_N$ a path in G. Let $\mathbf{Z}$ be a subset of variables. The path is blocked by $\mathbf{Z}$ if there is a node W such that either:

- W has converging arrows along the path ($\rightarrow W \leftarrow$) and neither W nor its descendants are in $\mathbf{Z}$

- W does not have converging arrows ($\rightarrow W \rightarrow$ or $\leftarrow W \rightarrow$), and $W \in \mathbf{Z}$

Definition 10. The set $\mathbf{Z}$ d-separates $\mathbf{X}$ and $\mathbf{Y}$ if every undirected path between any $X \in \mathbf{X}$ and any $Y \in \mathbf{Y}$ is blocked by $\mathbf{Z}$

The definition is more clear if we consider the three structures depicted in Figure 1.2. In the case of Figure 1.2 (a), $\mathbf{Z}$, d-separates X and Y and we can write the following:

$$\mathrm{P}(X,Y,Z) = \mathrm{P}(X)\,\mathrm{P}(Z\,|\,X)\,\mathrm{P}(Y\,|\,Z) = \mathrm{P}(Z)\,\mathrm{P}(X\,|\,Z)\,\mathrm{P}(Y\,|\,Z). \tag{1.6}$$

As $\mathrm{P}(X,Y,Z) = \mathrm{P}(X,Y\,|\,Z)\,\mathrm{P}(Z)$, the above means that $\mathrm{P} \models (X \perp Y\,|\,Z)$. The case of Figure 1.2 (b) leads to the same result (the demonstration is left to the reader), while the structure of Figure 1.2 (c) has a different outcome:

$$\mathrm{P}(X,Y\,|\,Z) = \mathrm{P}(X\,|\,Z)\,\mathrm{P}(Y\,|\,Z)\,\mathrm{P}(Z). \tag{1.7}$$

In this case, Z does not d-separate X and Y and it is not true that $\mathrm{P} \models (X \perp Y\,|\,Z)$, even if $\mathrm{P} \models (X \perp Y)$. This phenomenon is called *explaining away* and it is the reason of the condition about the nodes with converging arrows in the definition of d-separation. In more general terms, the equivalence between d-separation and conditional independence is stated as follows:

Theorem 1. *Let* (G,P) *be a Bayesian Network. Then if* $\mathbf{Z}$ *d-separates* $\mathbf{X}$ *and* $\mathbf{Y}$*,* $\mathrm{P} \models (\mathbf{X} \perp \mathbf{Y}\,|\,\mathbf{Z})$ *holds.*

Thus, the conditional independence assumptions underlying a Bayesian Network can be obtained by simply applying the d-separation criterion to the corresponding directed graph.

1.3.3 Hidden Markov Models

The example presented in Section 1.2, known as *Markov Model*, can be thought of as a Bayesian Network where $Pa(Y_t) = \{Y_{t-1}\}$:

$$\mathrm{P}(Y_1,\ldots,Y_N) = \mathrm{P}(Y_1)\prod_{i=2}^{N}\mathrm{P}(Y_i\,|\,Y_{i-1}) = \prod_{i=1}^{N}\mathrm{P}(Y_i\,|\,\mathrm{Pa}(Y_i)), \tag{1.8}$$

The DAG corresponding to this distribution is a linear chain of random variables.

An important related model is the Hidden Markov Model (HMM) [10, 23], where the variables can be split into two sets, the states $\mathbf{Y}$ and the observations $\mathbf{X}$:

$$\mathrm{P}(\mathbf{X},\mathbf{Y}) = \mathrm{P}(Y_1)\,\mathrm{P}(X_1\,|\,Y_1)\prod_{t=2}^{N}\mathrm{P}(Y_t\,|\,Y_{t-1})\,\mathrm{P}(X_t\,|\,Y_t) \tag{1.9}$$

where the terms $\mathrm{P}(Y_t\,|\,Y_{t-1})$ are called *transition probabilities*, the terms $\mathrm{P}(X_t\,|\,Y_t)$ are called *emission probability functions*, and the term $\mathrm{P}(Y_1)$ is called *initial state*

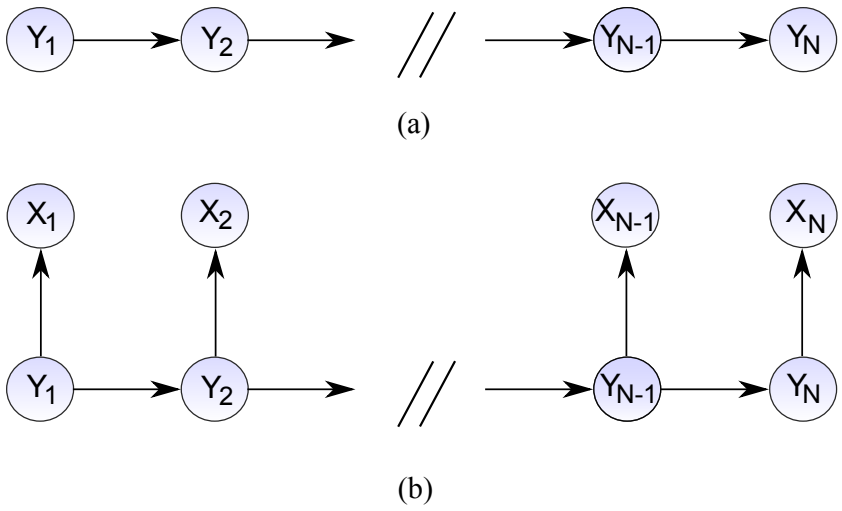

Fig. 1.3 The figure depicts the Bayesian Networks representing a Markov Model (a) and a Hidden Markov Model (b).

probability. The underlying assumptions are the Markov Property for the states and, for what concerns the observations, the conditional independence of one observation with respect to all of the others given the state at the same time.

HMMs have been used extensively for both classification and labeling problems. In the first case, one class is assigned to the whole sequence $\mathbf{X}$. For C classes, different sequences of states $\mathbf{Y^i}$ are used to estimate the probability $\mathrm{P}(\mathbf{X}, \mathbf{Y^i})$ and the one leading to the highest value is retained as the winning one:

$$k = \arg\max_{i \in [1,C]} = \mathrm{P}(\mathbf{X}, \mathbf{Y^i}), \tag{1.10}$$

where k is assigned to $\mathbf{X}$ as class. In the labeling case, the sequence of states $\hat{\mathbf{Y}}$ that satisfies the following equation:

$$\hat{\mathbf{Y}} = \arg\max_{\mathbf{Y} \in \mathscr{Y}} = \mathrm{P}(\mathbf{X}, \mathbf{Y}), \tag{1.11}$$

is used to label the observations of $\mathbf{X}$ ($\mathscr{Y}$ is the set of the state sequences of the same length as X). Each element X_t is labeled with the value y_t of variable $\hat{Y}_t$ in $\hat{\mathbf{Y}}$.

HMM have been widely used for speaker diarization (i.e. the task of segmenting an audio recording in speaker turn). In this scenario, the HMM is used as an unsupervised clustering algorithm. The hidden states $\mathbf{Y}$ of the model correspond to the speakers and the observations are features extracted from the audio spectrum (usually Mel-frequency cepstral coefficients [17]). For a description of a state of the art system using this approach see [8].

HMM suffers from two main limitations. The first is that the observations are assumed to be independent given the states. In the case of human behavior analysis, this assumption does not generally hold. The model presented in the next section, the Conditional Random Field, can address this problem.

The second limitation is that the Markov property makes it difficult to model the duration of the hidden states, i.e. the number of consecutive observations labeled with the same state. The reason is that the probability of transition to a state y_t depends only on y_{t-1}. The Hidden Semi-Markov Model [26] was developed to address this limitation. A complete description of this model is beyond the scope of this chapter, but the key idea is to have the transition probabilities to y_t that depend not only on y_{t-1}, but also on the number of consecutive observations that have been labeled with y_{t-1}.

1.4 Conditional Random Fields

Conditional Random Fields [14, 15, 27] differ from Bayesian Networks under two main respects: The first is that they are encompassed by undirected graphical models, the second is that they are *discriminative*, i.e. they model $\mathrm{P}(\mathbf{Y}\,|\,\mathbf{X})$ and not $\mathrm{P}(\mathbf{X},\mathbf{Y})$. The former aspect influences the factorization as well as the way the graph encodes conditional independence assumptions. The latter aspect brings the important advantage that no assumptions about $\mathbf{X}$ need to be made (see below for more detail).

1.4.1 Factorization and Conditional Independence

Definition 11. Let $G=(\mathbf{N},\mathbf{E})$ be a graph such that the random variables in $\mathbf{Y}$ correspond to the nodes of G and let P be a joint probability distribution defined over $\mathbf{Y}$. A pair (G,P) is a Markov Random Field if:

$$\mathrm{P}(Y\,|\,\mathbf{Y}\setminus\{Y\})=\mathrm{P}(Y\,|\,\mathrm{Nb}(Y))\forall Y\in\mathbf{Y}. \tag{1.12}$$

The factorization of P is given by the following theorem:

Theorem 2. *Let (G,P) be a Markov Random Field, then there exists a set of functions $\{\varphi_c\,|\,c$ is a clique of $G\}$ such that*

$$\mathrm{P}(\mathbf{Y})=\frac{1}{Z}\prod_c \varphi_c(\mathbf{Y}|_c), \tag{1.13}$$

where $\mathbf{Y}|_c$ is the subset of $\mathbf{Y}$ that includes only variables associated to the nodes in c, and Z is a normalization constant:

$$Z = \sum_{\mathbf{y}} \prod_{c} \varphi_c(\mathbf{y}|_c), \tag{1.14}$$

where $\mathbf{y}$ *iterates over all possible assignments on* $\mathbf{Y}$.

The functions φ_c are often called potentials. They need to be positive functions but they do not necessarily need to be probabilities, i.e. they are not bound to range between 0 and 1. The conditional independence assumptions underlying the factorization above can be inferred by considering the definition of the Markov Network. Each variable is conditionally independent of all of the others given those who correspond to the nodes in its neighborhood: $\mathrm{P} \models (Y \perp \mathbf{Y} \setminus \{Y, Nb(Y)\} \,|\, Nb(Y))$.

Conditional Random Fields are based on Markov Networks and are defined as follows:

Definition 12. Let $G = (\mathbf{N}, \mathbf{E})$ be a graph such that the random variables in $\mathbf{Y}$ correspond to the nodes of G. The pair $(\mathbf{X}, \mathbf{Y})$ is a *Conditional Random Field* (CRF) if the random variables in $\mathbf{Y}$ obey the Markov property with respect to the graph G when conditioned on X:

$$\mathrm{P}(Y \,|\, \mathbf{X}, \mathbf{Y} \setminus Y) = \mathrm{P}(Y \,|\, \mathbf{X}, Nb(Y)). \tag{1.15}$$

the variables in $\mathbf{X}$ are called *observations* and those in $\mathbf{Y}$ *labels*.

The definition above does not require any assumption about $\mathbf{X}$ and this is an important advantage. In both labeling and classification problems, $\mathbf{X}$ is a constant and the value of $\mathrm{P}(\mathbf{X}, \mathbf{Y})$ must be maximized with respect to $\mathbf{Y}$:

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} \mathrm{P}(\mathbf{Y} \,|\, \mathbf{X}) \mathrm{P}(\mathbf{X}) = \arg\max_{\mathbf{Y}} \mathrm{P}(\mathbf{Y} \,|\, \mathbf{X}) \tag{1.16}$$

Thus, modeling explicitly $\mathbf{X}$ (like it happens, e.g., in Hidden Markov Models) is not really necessary. The model does not require conditional independence assumptions for the observations that might make the models too restrictive for the data and affect negatively the performance. In this respect, modeling $\mathrm{P}(\mathbf{Y} \,|\, \mathbf{X})$ makes the model more fit to the actual needs of labeling and classification (see equation above) and limits the need of conditional independence assumptions to the only $\mathbf{Y}$.

The factorization of Conditional Random Fields is as follows:

Theorem 3. *Le* (G, P) *be a Markov Network, then there exists a set of functions* $\{\varphi_c \,|\, c \text{ is a clique of } G\}$ *such that*

$$\mathrm{P}(\mathbf{y} \,|\, \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{c} \varphi_c(\mathbf{y}|_c, \mathbf{x}). \tag{1.17}$$

Z is is a normalization constant called the partition function*:*

$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \prod_{c} \varphi_c(\mathbf{y}|_c, \mathbf{x}), \tag{1.18}$$

where $\mathbf{y}$ *iterates over all possible assignments on* $\mathbf{Y}$.

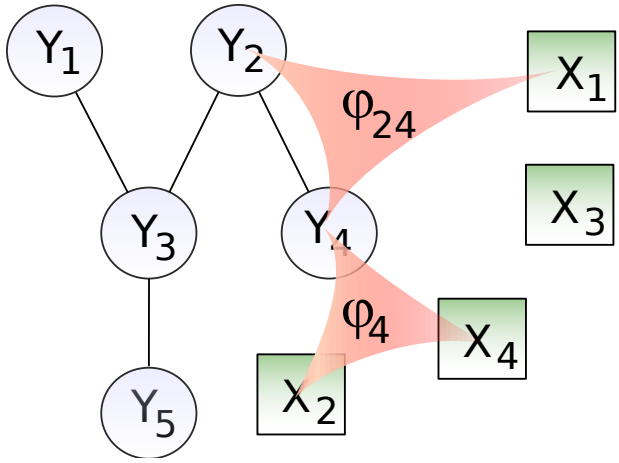

Fig. 1.4 Conditional Random Fields. The potentials are defined over cliques and have as argument the variables corresponding to the nodes of the clique and an arbitrary subset of the observation sequence X.

The problem left open so far is the definition of the potentials. As this chapter focuses on sequence analysis, the rest of this section will consider the particular case of *Linear Chain Conditional Random Fields*, one of the models most commonly applied for the sequence labeling problem.

1.4.2 Linear Chain Conditional Random Fields

In linear chain CRFs, the cliques are pairs of nodes corresponding to adjacent elements in the sequence of the labels or individual nodes (see Figure 1.5):

Definition 13. A graph is a *chain* if and only if $E = \{(y_i, y_{i+1}), 1 \leq i < |Y|\}$.

where E is the set of the edges and (y_i, y_{i+1}) represents the edge between the nodes corresponding to elements Y_i and Y_{i+1} in $\mathbf{Y}$.

The following assumptions must be made about the potentials to make the model tractable:

1. The potential over $\{y_t, y_{t+1}\}$ depends only on y_t and y_{t+1}.
2. The potential over $\{y_t\}$ depends only on y_t and x_t.
3. The potentials are the same for all t.
4. The potentials are never zero.

These first three assumptions mean that the marginal distribution for y_t is fully determined by y_{t-1}, y_{t+1} and x_t. The fourth assumption means that every sequence

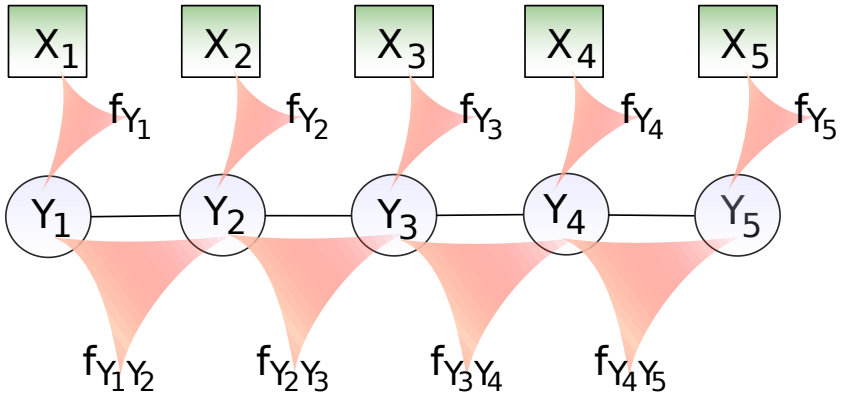

Fig. 1.5 Linear Chain Conditional Random Fields. The cliques in a chain are pair of adjacent labels or individual labels. The potentials are function of adjacent nodes or of a node and the corresponding observation.

of labels $\mathbf{Y}$ has a probability strictly greater than zero. This last assumption is important in practice because it allows the product of potentials to be replaced by the exponential of a sum as follows [14] :

$$\mathrm{P}(Y|X) = \frac{\exp\left(\sum_{t=1}^{N} f_1(y_t, \mathbf{x}_t) + \sum_{t=1}^{N-1} f_2(y_t, y_{t+1})\right)}{Z(X)}$$

$$Z(X) = \sum_{Y \in \mathscr{Y}^N} \exp\left(\sum_{t=1}^{N} f_1(y_t, \mathbf{x}_t) + \sum_{t=1}^{N-1} f_2(y_t, y_{t+1})\right)$$

where f_1 and f_2 represent potentials having as argument only one label y_t or a pair of adjacent labels $\{y_t, y_{t+1}\}$. Thus, the potentials have been represented as a linear combination of simpler terms called *feature functions*.

In general, the feature functions used for f_1 are as follows:

$$f_{y,t}(y_t, \mathbf{x}) = \begin{cases} x_t & \text{if } y_t = y \\ 0 & \text{otherwise} \end{cases} \tag{1.19}$$

where x_t is the observation at time t. This family of feature functions can capture linear relations between a label and an observation x_t. For f_2, the feature functions are typically as follows:

$$f_{y,y'}(y_t, y_{t+1}) = \begin{cases} 1 & \text{if } y_t = y \text{ and } y_{t+1} = y' \\ 0 & \text{otherwise} \end{cases} \tag{1.20}$$

In summary, Linear Chain CRFs estimate $p(\mathbf{Y}|\mathbf{X})$ as follows:

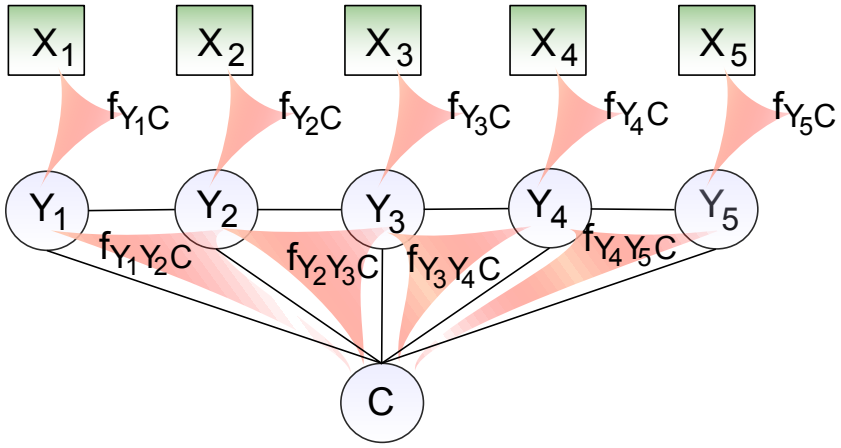

Fig. 1.6 hidden Conditional Random Fields. The class is represented by the C. The variables Y_i are not observed. The potentials are function of adjacent nodes and the class ($f_{Y_iY_{i+1}C}$) or of a node the corresponding observation and the class (f_{Y_iC}). The potentials f_{Y_iC} are not drawn connected to C to keep the figure readable.

$$p(\mathbf{Y}|\mathbf{X},\alpha) = \frac{1}{Z(\mathbf{X})} \exp\left(\begin{array}{l} \sum_{t=1}^{N} \sum_{y\in\mathscr{Y}} \alpha_y f_{y,t}(y_t, x_t) + \\ \sum_{t=1}^{N-1} \sum_{(y,y')\in\mathscr{Y}^2} \alpha_{y,y'} f_{y,y'}(y_t, y_{t+1}) \end{array} \right) \tag{1.21}$$

The weights α_y of the feature functions of form $f_{y,t}(\mathbf{X},\mathbf{Y})$ account for how much the value of a given observation is related to a particular label. The weights of the feature functions of form $f_{y,y'}(\mathbf{X},\mathbf{Y})$ account for how frequent it is to find label y followed by label y'.

Linear Chain CRF have been used with success in role recognition [25], where the goal is to map each turn into a role. In this case, the labels correspond to a sequence of roles. The observations are feature vectors accounting for prosody and turn taking patterns associated to each turn.

CRFs have several extensions aimed at addressing the weaknesses of the basic model, in particular the impossibility of labeling sequences as a whole and of modeling latent dynamics. Two effective extensions are obtained by introducing latent variables in the model. The first of these extensions is the hidden Conditional Random Field (hCRF) [22] and it aims at labeling a sequence as a whole. The hCRFs are based on linear chain CRFs, where the chain of labels $\mathbf{Y}$ is latent and a new variable C is added (see Figure 1.6). The new variable C represents the class of the observations and is connected connected to every label. All of the potentials are modified to depend on the class C (see Figure 1.6).

The second extension aims at modeling latent dynamics like, for example, a single gesture (e.g., hand waving) that can have several states (hand moving left and

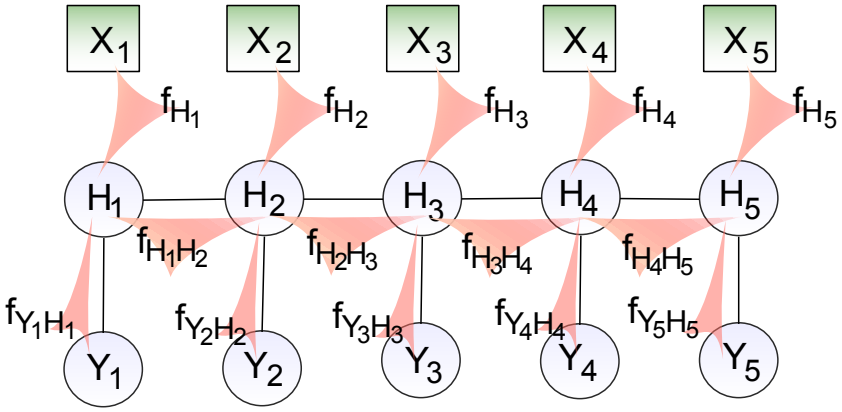

Fig. 1.7 Latent Dynamic Conditional Random Fields. The variables H_i are not observed and capture the latent dynamic. The potentials are function of adjacent hidden states, a hidden state and the corresponding label, or an hidden state and the corresponding observation.

hand moving right) associated with a single label. CRFs cannot model these states and the dynamics associated with them. The Latent Discriminative Conditional Random Fields (LDCRF) [18] were introduced to overcome this drawback. LDCRF introduce a linear chain of latent variables between the observations and the labels (see Figure 1.7). The labels are disconnected and thus assumed to be conditionally independent given the hidden states. Also, the labels are not directly connected to the observations.

1.5 Training and Inference

The models presented so far cannot be used without appropriate *training* and *inference* techniques. The training consists in finding the parameters of a model (e.g., the transition probabilities in a Hidden Markov Model or the α coefficients in a Conditional Random Field) that *better fit* the data of a training set, i.e. a collection of pairs $\mathscr{T} = \{(\mathbf{X^i}, \mathbf{Y^i})\}$ $(i = 1, \ldots, |\mathscr{T}|)$ where each observation is accompanied by a label supposed to be true. By "better fit" it is meant the optimization of some criterion like, e.g., the maximization of the likelihood or the maximization of the entropy (see below for more details).

The inference consists in finding the value of $\mathbf{Y}$ that better fits an observation sequence $\mathbf{X}$, whether this means to find the individual value of each Y_j that better matches each $\mathbf{X}$:

$$\mathrm{P}(Y_j = y \,|\, \mathbf{X}) = \sum_{\mathbf{Y} \in \{\mathbf{Y}, Y_j = y\}} \mathrm{P}(\mathbf{Y} \,|\, \mathbf{X}) \tag{1.22}$$

or finding the sequence $\mathbf{Y}^*$ that globally better matches $\mathbf{X}$:

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} \mathrm{P}(\mathbf{Y}\,|\,\mathbf{X}). \tag{1.23}$$

The number of possible sequences increases exponentially with $\mathbf{Y}$, thus training and inference cannot be performed by simply estimating $\mathrm{P}(\mathbf{Y}\,|\,\mathbf{X})$ for every possible $\mathbf{Y}$. The next two sections introduce some of the key techniques necessary to address both tasks with a reasonable computational load.

1.5.1 Message Passing

One of the main issues in both training and inference is to estimate the probability $\mathrm{P}(Y_j = y)$ that a given label Y_j takes the value y. The *Message Passing* algorithm allows one to perform such a task in an efficient way by exploiting the local structure of the graph around the node corresponding to Y_j (see [31] for an extensive survey of the subject). In particular, the key idea is that the marginal distribution of a node Y_j can be determined if the value of the variables corresponding to its neighboring nodes are known. In practice, those values are unknown, but it is possible to estimate the *belief* that measures the relative probability of the different values. For this reason, the message passing algorithm is sometimes referred to as *belief propagation*.

This section will focus in particular on the message passing algorithm for Pairwise Markov Networks, namely Markov Networks where the cliques include no more than two nodes. While being an important constraint, still it includes cases of major practical importance such as chains, trees and grids (the Linear Chain Conditional Random Fields fall in this class).

The beliefs are defined as follows:

$$b_j(y_j) = \phi_j(y_j) \prod_{k \in Nb(Y_j)} m_{kj}(y_j) \tag{1.24}$$

where $\varphi_j(y_j)$ is the potential for node Y_j, m_{kj} is the message from node Y_k to node Y_j (see below for the definition of the messages). Formally, a belief is a function that maps each possible value of Y_j into a real number.

A message is another function that maps the value of one node into a real number and it represents the influence that the sending node has on the receiving one:

$$m_{kj}(y_j) = \sum_{y_k} \left(\varphi_k(y_k)\varphi_{jk}(y_j, y_k) \prod_{n \in Nb(Y_k)\setminus\{Y_j\}} m_{nk}(y_k) \right) \tag{1.25}$$

where φ_{jk} is the potential of the clique including Y_j and Y_k (this equation motivates the name *sum-product* algorithm that it is used sometimes for this algorithm).

The belief propagation requires the variables to be ordered and this might create problems when a graph contain cycles. When cycles are absent (which is the case

for the models considered in this chapter), the following procedures allow one to find a suitable ordering:

1. Choose a root node
2. Compute messages starting at the leaf moving to the root
3. Compute messages starting at the root, going to the leafs

It is important to note that the value of the message is independent of the order in which the messages are passed.

At the end of the procedure, each node is associated with a belief that can be used to compute the marginal probabilities as shown by the following:

Theorem 4. *Let G be a pairwise random field on $\mathbf{Y}$ and b_j the beliefs computed using the message passing algorithm, then the following holds:*

$$\mathrm{P}(Y_j = y_j) = \frac{b_j(y_j)}{\sum_{y_i} b_j(y_i)}. \tag{1.26}$$

In the case of Conditional Random Fields, the observations in $\mathbf{X}$ have to be taken into account. The message and the beliefs are now dependent on $\mathbf{X}$:

$$b_j(y_j, \mathbf{X}) = \varphi_j(y_j, \mathbf{X}) \prod_{Y_k \in Nb(Y_j)} m_{kj}(y_j, \mathbf{X}) \tag{1.27}$$

$$m_{kj}(y_j, \mathbf{X}) = \sum_{y_k, \mathbf{X}} \left(\varphi_k(y_k, \mathbf{X}) \varphi_{jk}(y_j, y_k, \mathbf{X}) \prod_{Y_n \in Nb(Y_k) \setminus \{Y_j\}} m_{nk}(y_k, \mathbf{X}) \right) \tag{1.28}$$

$$\tag{1.29}$$

As $\mathbf{X}$ is a constant and it is known a-priori, it is possible to apply exactly the same equations as those used for the Markov Networks.

1.5.1.1 Inference

There are two possible inference scenarios (see beginning of this section): The first consists in finding, for each label, the assignment that maximizes the marginal probability. The second consists in finding the assignment that maximizes the joint probability distribution over the entire labels sequence $\mathbf{Y}$.

The first case is a straightforward application of the message passing algorithm. For a given label Y_j, it is sufficient to use the beliefs to find the particular value y^* that maximizes the following probability:

$$y^* = \arg\max_y \mathrm{P}(Y_j = y) = \arg\max_y b_j(y). \tag{1.30}$$

It can be demonstrated that this particular way of assigning the values to the labels minimizes the misclassification rate.

In the second case, the expression of the messages in Equation (1.25) must be modified as follows:

$$m_{kj}(y_j) = \max_{y_k} \left(\varphi_k(y_k)\varphi_{jk}(y_j, y_k) \prod_{n \in Nb(Y_k)\setminus\{Y_j\}} m_{nk}(y_k) \right) \tag{1.31}$$

where the initial sum has been changed to a maximization. This ensures that the message received by the node corresponding to label Y_j brings information about the sequence $(Y_1, \ldots, Y_{j-1})$ with the highest possible probability rather than about the sum of the probabilities over all possible sequences.

It is again possible to assign to each Y_j, the value y_j^* that maximize the beliefs obtained using the modified messages:

$$y_j^* = \arg\max_y b_j(y). \tag{1.32}$$

It can be shown that the resulting assignment $Y^* = \{y_1^*, \ldots, y_n^*\}$ is the sequence with the maximum probability:

$$Y^* = \arg\max_Y \mathrm{P}(Y) \tag{1.33}$$

1.5.1.2 Training

The last important aspect of probabilistic sequential models is the training. The topic is way too extensive to be covered in detail and the section will focus in particular on Markov Networks as this can be a good starting point towards training Conditional Random Fields. If the assumption is made that the potentials are strictly greater than zero, then Markov Networks can be factorized as follows:

$$\mathrm{P}(\mathbf{Y} \mid \alpha) = \frac{1}{Z} \exp\left(\sum_c \sum_{i=1}^{n_c} \alpha_c^i f_c^i(\mathbf{Y}|_c) \right) \tag{1.34}$$

$$Z = \sum_{\mathbf{Y}} \exp\left(\sum_c \sum_{i=1}^{n_c} \alpha_c^i f_c^i(\mathbf{Y}|_c) \right) \tag{1.35}$$

where the $f_c^i(Y|_c)$ are feature functions defined over a clique c. The same expression as the same as the one used for Conditional Random Fields, but without the observations $\mathbf{X}$.

Training such a model it means to find the values of the coefficients α that optimize some criteria over a training set. This section considers in particular the maximization of the likelihood:

$$\alpha^* = \arg\max_\alpha \sum_j \log \mathrm{P}(\mathbf{Y^j} \mid \alpha) \tag{1.36}$$

where the $\mathbf{Y^j}$ are the sequences of the training set.

The main problem is that solving the above equation leads to an expression for the α coefficients which is not in closed form, thus it is necessary to apply gradient ascent techniques. On the other hand, these are effective because of the following:

Theorem 5. *The log-likelihood function is concave with respect to the weights.*

In practice, the LBFGS algorithm [16] works well and this has two main motivations: The first is that the algorithm approximates the second derivative and thus converges faster, the second is that has a low memory usage and work well on large scale problems. One of the main steps of the LFBGS is the estimation of the derivative of the loglikelihood with respect to α.

$$\frac{\partial}{\partial \alpha_i^c} \sum_j \log \mathrm{P}(\mathbf{Y}^\mathbf{j}) = \frac{\partial}{\partial \alpha_i^c} \sum_j \log \left(\frac{1}{Z} \exp \left(\sum_c \sum_{i=1}^{n_c} \alpha_c^i f_c^i(\mathbf{Y}^\mathbf{j}|_c) \right) \right) \tag{1.37}$$

$$= \frac{\partial}{\partial \alpha_i^c} \sum_j \left(\sum_c \sum_{i=1}^{n_c} \alpha_c^i f_c^i(\mathbf{Y}^\mathbf{j}|_c) \right) - \frac{\partial}{\partial \alpha_i^c} \sum_j \log Z \tag{1.38}$$

$$= \sum_j \left(f_c^i(\mathbf{Y}^\mathbf{j}|_c) - E[f_c^i] \right) \tag{1.39}$$

The equation above shows that the optimal solution is the one where the theoretical expected value of the feature functions is equal to their empirical expected value. This corresponds to the application of the Maximum Entropy Principle and it further explains the close relationship between Conditional Random Fields and Maximum Entropy Principle introduced in this section.

1.6 Conclusions

This chapter has introduced the problem of sequence analysis in machine learning. The problem has been formulated in terms of two major issues, namely classification (assigning a label to an entire sequence of observations) and labeling (assigning a label to each observation in a sequence). The chapter has introduced some of the most important statistical models for sequence analysis, Hidden Markov Models and Conditional Random Fields. The unifying framework of Probabilistic Graphical Models has been used in both cases and the accent has been put on factorization and conditional independence assumptions. Some details about training and inference issues have been provided for Conditional Random Fields and, more generally, for undirected graphical models.

The models introduced in this chapter are not aimed in particular at human behavior understanding, but they have been used successfully in the domain (see [28] for an extensive survey of the domain). Sequences arise naturally in many behavior analysis problems, especially in the case of social interactions where two or more individuals react to one another and produce sequences of social actions [21].

While trying to provide an extensive description of the sequence analysis problem in machine learning, this chapter cannot be considered exhaustive. However, the chapter, and the references therein, can be considered a good starting point towards a deeper understanding of the problem. In particular, graphical models have been the subject of both tutorials (see, e.g., [19] and Chapter 8 of [5]) and dedicated monographies [14], the same applies to Hidden Markov Models (see, e.g., [23] for a tutorial and [10] for a monography) and Conditional Random Fields (see, e.g., [29] for a tutorial and [14] for a monography).

Last, but not least, so far Human Sciences and Computing Science (in particular machine learning) have looked at the sequence analysis problem in an independent way. As the cross-pollination between the two domains improves, it is likely to expect models more explicitly aimed at the human behavior understanding problem.

Questions

Question 1. What is the *rationale* behind Equation (1.1)?

Question 2. Consider the graph represented in Figure 1.2 (c). Let X, Y and Z be binary random variables. Let the probability of the Bayesian Network be defined by the following conditional probabilities:

X	$\mathrm{P}(X)$
0	0.6
1	0.4

Y	$\mathrm{P}(Y)$
0	0.5
1	0.5

X	Y	$\mathrm{P}(Z=0\,\vert\,X,Y)$	$\mathrm{P}(Z=1\,\vert\,X,Y)$
0	0	0.8	0.2
0	1	0.6	0.4
1	0	0.5	0.5
1	1	0.6	0.4

Without using Theorem 1, prove the following:

1. $\mathrm{P} \models (X \perp Y)$
2. $\mathrm{P} \not\models (X \perp Y \,|\, Z)$

Question 3. Consider the Markov Model (MM) and the Hidden Makov Model (HMM) presented in Figure 1.3. Find a smallest possible set that:

1. d-separates Y_1 from Y_N in the case of MM.
2. d-separates Y_1 from Y_N in the case of HMM.

Prove that there is no subset of the observations $\mathbf{X}$ that d-separates Y_1 from Y_N in the case of HMMs.

Question 4. What is the conditional independence assumption made by the Linear Chain Conditional Random Fields?

Question 5. Let (G,P) be a Markov Random Field, where G is the undirected graph in Figure 1.1. By applying Equations (1.25) and (1.24) give the expressions for:

1. $m_{45}(y_5)$.
2. $b_5(y_5)$.
3. $\sum_{y_5} b_5(y_5)$.

Remark that the product in the third case can be rearranged to yield Z as this is a special case of Theorem 4.

Question 6. Prove Theorem 5: The log-likelihood function is concave with respect to the weights. This proof requires some background in analysis and use materials not presented in this chapter. A proof is given in [14], Chapter 20.3.

Glossary

Probabilistic Sequential Model Probability distribution defined over sequences of continuous or discrete random variables.

Sequence Ordered set of continuous or discrete random variables (typically corresponding to measurements collected at regular steps in time or space).

Probabilistic Graphical Model Joint probability distribution defined over a set of random variables corresponding to the nodes of a (directed or undirected) graph.

Graph Data structure composed of a set of nodes and a set of edges, two nodes can be connected by a directed or undirected edge.

Conditional Independence Let $\mathbf{X}, \mathbf{Y}$, and $\mathbf{Z}$ be sets of random variables. We say that $\mathbf{X}$ is *conditionally independent* of $\mathbf{Y}$ given $\mathbf{Z}$ if and only if:

$$\mathrm{P}(\mathbf{X}, \mathbf{Y} \,|\, \mathbf{Z}) = \mathrm{P}(\mathbf{X} \,|\, \mathbf{Z})\, \mathrm{P}(\mathbf{Y} \,|\, \mathbf{Z})$$

References

1. A. Abbott. Sequence analysis: New methods for old ideas. *Annual Review of Sociology*, 21:93–113, 1995.
2. R. Bakeman and J.M. Gottman. *Observing interaction: An introduction to sequential analysis*. Cambridge University Press, 1986.
3. P. Baldi and S. Brunak. *Bioinformatics: the machine learning approach*. The MIT Press, 2001.
4. J. Bilmes. The concept of preference in conversation analysis. *Language in Society*, 17(2):161–181, 1988.
5. C.M. Bishop. *Pattern recognition and machine learning*. Springer New York, 2006.
6. F. Camastra and A. Vinciarelli. *Machine Learning for Audio, Image and Video Analysis: Theory and Applications*. Springer Verlag, 2008.
7. T. Dietterich. Machine learning for sequential data: A review. In T. Caelli, A. Amin, R. Duin, D. de Ridder, and M. Kamel, editors, *Structural, Syntactic, and Statistical Pattern Recognition*, volume 2396 of *Lecture Notes in Computer Science*, pages 227–246. Springer Berlin / Heidelberg, 2002.
8. G. Friedland, O. Vinyals, Y. Huang, and C. Muller. Prosodic and other long-term features for speaker diarization. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(5):985–993, 2009.

9. D. Heckerman. A tutorial on learning with bayesian networks. In D. Holmes and L. Jain, editors, *Innovations in Bayesian Networks*, pages 33–82. Springer Berlin / Heidelberg, 2008.
10. F. Jelinek. *Statistical methods for speech recognition*. the MIT Press, 1997.
11. F.V. Jensen. *An introduction to Bayesian Networks*. UCL press London, 1996.
12. F.V. Jensen and T.D. Nielsen. *Bayesian Networks and decision graphs*. Springer Verlag, 2007.
13. M.I. Jordan. *Learning in graphical models*. Kluwer Academic Publishers, 1998.
14. D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, 2009.
15. J. Lafferty, A. McCallum, and F. Pereira. Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning*, pages 282–289, 2001.
16. D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.
17. P. Mermelstein. Distance measures for speech recognition, psychological and instrumental. *Pattern Recognition and Artificial Intelligence*, 116, 1976.
18. L.P. Morency, A. Quattoni, and T. Darrell. Latent-dynamic discriminative models for continuous gesture recognition. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.
19. K. Murphy. An introduction to graphical models. Technical report, University of British Columbia, 2001.
20. J. Pearl. *Bayesian networks: A model of self-activated memory for evidential reasoning*. Computer Science Department, University of California, 1985.
21. I. Poggi and F. D'Errico. Cognitive modelling of human social signals. In *Proceedings of the 2nd International Workshop on Social Signal Processing*, pages 21–26, 2010.
22. A. Quattoni, S. Wang, L.P. Morency, M. Collins, and T. Darrell. Hidden conditional random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1848–1852, 2007.
23. L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
24. H. Salamin, A. Vinciarelli, K. Truong, and G. Mohammadi. Automatic role recognition based on conversational and prosodic behaviour. In *Proceedings of the ACM International Conference on Multimedia*, pages 847–850, 2010.
25. H. Salamin, A. Vinciarelli, K. Truong, and G. Mohammadi. Automatic role recognition based on conversational and prosodic behaviour. In *Proceedings of the international conference on Multimedia*, pages 847–850. ACM, 2010.
26. J. Sansom and P. Thomson. Fitting hidden semi-Markov models to breakpoint rainfall data. *Journal of Applied Probability*, 38:142–157, 2001.
27. C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to statistical relational learning*. The MIT Press, 2007.
28. A. Vinciarelli, M. Pantic, and H. Bourlard. Social Signal Processing: Survey of an emerging domain. *Image and Vision Computing*, 27(12):1743–1759, 2009.
29. H.M. Wallach. Conditional Random Fields: An introduction. Technical Report MS-CIS-04-21, Department of Computer and Information Science, University of Pennsylvania, 2004.
30. Y. Wu and T. Huang. Vision-based gesture recognition: A review. In A. Braffort, R. Gherbi, S. Gibet, D. Teil, and J. Richardson, editors, *Gesture-Based Communication in Human-Computer Interaction*, volume 1739 of *Lecture Notes in Computer Science*, pages 103–115. Springer Berlin / Heidelberg, 1999.
31. J.S. Yedidia, W.T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. In G. Lakemeyer and B. Nebel, editors, *Exploring artificial intelligence in the new millennium*, pages 239–270. Morgan Kaufman, 2003.