Advanced Deep Learning

Structured Probabilistic Models for Deep Learning

U Kang
Seoul National University

In This Lecture

- Challenge of Unstructured Modeling
- Using Graphs to Describe Model Structure
- Sampling from Graphical Models
- Advantages of Structured Modeling
- Learning about Dependencies
- Inference and Approximate Inference
- Deep Learning Approach to Structured PM

Outline

- ☐ **Challenge of Unstructured Modeling**
- ☐ Using Graphs to Describe Model Structure
- ☐ Sampling from Graphical Models
- ☐ Advantages of Structured Modeling
- ☐ Learning about Dependencies
- ☐ Inference and Approximate Inference
- ☐ Deep Learning Approach to Structured PM

Probabilistic Model

- Goal of deep learning: scale machine learning to the kinds of challenges needed to solve artificial intelligence
- Classification
  - Discards most of the information in the input and produces a single output (or a probability distribution over values of that single output)
- It is possible to ask probabilistic models to do many other tasks, which are often more expensive than classification
  - Producing multiple output values
  - Complete understanding of the entire structure of the input

Probabilistic Model

- Examples of expensive tasks
  - Density estimation
    - Given an input x, estimate the true density p(x) under the data generating distribution
    - Requires a complete understanding of the entire input
  - Denoising
    - Given a damaged x', return an estimate of the correct x
    - Requires multiple outputs and an understanding of the entire input

Probabilistic Model

- Examples of expensive tasks
  - Missing value imputation
    - Given the observations of some elements of x, return estimates or a probability distribution over some or all of the unobserved elements of x
    - Requires multiple outputs and a complete understanding of the entire input
  - Sampling
    - Generate samples from distribution p(x). E.g., speech synthesis
    - Requires multiple output values and a good model of the entire input

Challenges

- Modeling a distribution over a random vector x containing n discrete variables capable of taking on k values each
  - Naïve approach for representing P(x) requires a lookup table with k^n entries
- Problems of lookup table based approach
  - Memory
  - Statistical efficiency: requires exponential number of training data points to fit
  - Runtime for inference: e.g., computing marginal distribution $P(x_1)$ requires summing up large entries
  - Runtime for sampling: e.g., sampling some value u~ U(0,1) and iterating through the table until the cumulative probability is u is inefficient

Challenges

- Problem with the table-based approach
  - Explicitly model every possible kind of interactions
- Structured probabilistic models provide a formal framework for modeling only limited interactions between random variables
  - This allows the models to have significantly fewer parameters and therefore be reliably estimated from less data
  - These smaller models also have dramatically reduced computational cost in terms of storing the model, performing inference in the model, and drawing samples from the model

Outline

- ☑ Challenge of Unstructured Modeling
- ☐ **Using Graphs to Describe Model Structure**
- ☐ Sampling from Graphical Models
- ☐ Advantages of Structured Modeling
- ☐ Learning about Dependencies
- ☐ Inference and Approximate Inference
- ☐ Deep Learning Approach to Structured PM

Probabilistic Graphical Model

- Use graphs to represent interaction between random variables
- Nodes - random variables
- Edges - statistical dependencies between these variables
  - Correlations relationships
  - Causality relationships

- 2 categories of graphical models
  - Directed graphical model (=Belief network, Bayesian network)
  - Undirected graphical model (=Markov network, Markov random field)

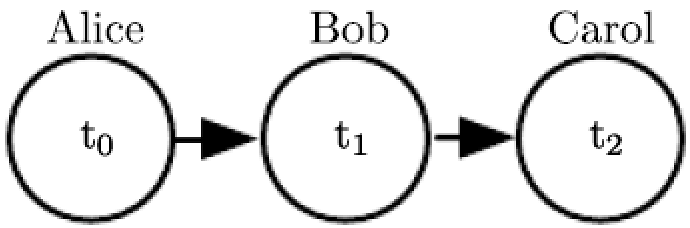
Directed Graphical Model

- Edges are directed
- Each direction indicates which variable's probability distribution is defined in terms of the others
  - E.g., an arrow from a to b means that the distribution of b depends on a
- A directed graphical model defined on variables x is defined by a directed acyclic graph G whose vertices are the random variables in the model, and a set of local conditional probability distributions $p(x_i|Pa_G(x_i))$ where $Pa_G(x_i)$ gives the parents of x_i in G
- The probability distribution over x is given by

$$p(x) = \prod_i \ p(x_i|Pa_G(x_i))$$

Directed Graphical Model

- Example: relay race

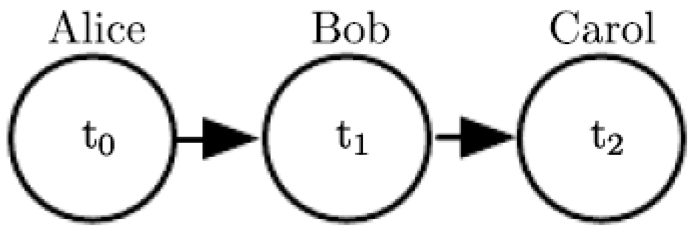


  - Alice's finishing time t_0 influences Bob's finishing time t_1, because Bob does not get to start running until Alice finishes
  - Likewise, Bob's finishing time t_1 influences Carol's finishing time t_2

  - $p(t_0, t_1, t_2) = p(t_0)p(t_1|t_0)p(t_2|t_1)$

Directed Graphical Model

- Example: relay race



  - $p(t_0, t_1, t_2) = p(t_0)p(t_1|t_0)p(t_2|t_1)$

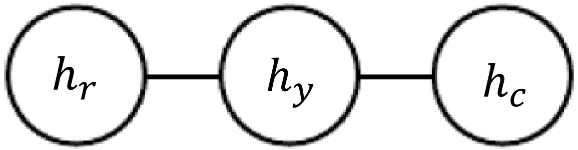
  - Assume we discretize time into 100 possible values
  - Table based approach: requires 999,999 values
  - Graphical model: 99 + 99*100 + 99*100 = 19,899 values
  - Using graphical model reduced the number of parameters more than 50 times!

Directed Graphical Model

- Table based approach vs graphical model
  - To model n discrete variables each having k values, the cost of the single table approach scales like O(k^n)
  - A directed graphical model where m is the maximum number of variables appearing in a single conditional probability distribution, then the cost of the graphical model scales like O(k^m)
  - As long as we can design a model such that m << n, we get dramatic savings

Undirected Graphical Model

- Directed models are most naturally applicable to situations where there is a clear causality
- Undirected models are appropriate where we might not clearly define the causation
  - Example: modeling health condition



  - h_r: your roommate's health
  - h_y: your health
  - h_c: your work colleague's health

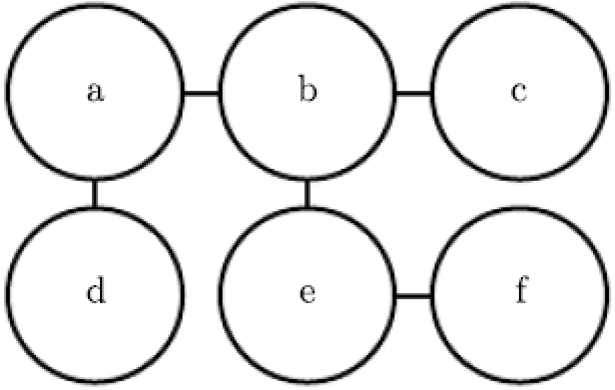
Undirected Graphical Model

- An undirected graphical model is defined on an undirected graph G
- For each clique C in the graph, a non-negative factor (or clique potential) $\phi(C)$ measures the affinity of the variables in that clique
- An **unnormalized probability distribution** is given by

$$\tilde{p}(\mathbf{x}) = \prod_{C \in G} \phi(C)$$

Undirected Graphical Model

- Example



- This graph implies that p(a,b,c,d,e,f) can be written as $\frac{1}{Z}\phi_{a,b}(a,b)\phi_{b,c}(b,c)\phi_{a,d}(a,d)\phi_{b,e}(b,e)\phi_{e,f}(e,f)$ for an appropriate choice of the ϕ function; Z is called the partition function

Undirected Graphical Model

- Unlike in a Bayesian network, there is little structure to the definition of the cliques, so there is nothing to guarantee that multiplying them together will yield a valid probability distribution

The Partition Function (1)

- A probability distribution must sum to 1
- The unnormalized probability distributions can be normalized as follows:

$$p(\mathbf{x}) = \frac{1}{Z}\tilde{p}(\mathbf{x})$$

- We call a constant Z the **partition function**:

$$Z = \int \tilde{p}(\mathbf{x})d\mathbf{x}$$

The Partition Function (2)

- Z is an integral or sum over all possible joint assignments of the state $\mathbf{x}$
- Thus it is usually *intractable* to compute
- Therefore, we resort to approximations (details in chapter 18)

The Partition Function (3)

- It is possible that Z does not exist
- For example, suppose a single variable $x \in \mathbb{R}$
- The clique potential is given as $\phi(x) = x^2$
- Then, the potential function Z is defined as

$$Z = \int x^2 dx$$

- Since it diverges, it is not a probability distribution

Energy-Based Models (1)

- Many undirected models assume that
$$\forall x, \tilde{p}(\mathbf{x}) > 0$$
- A convenient way to enforce this assumption is to use an **energy-based model** (EBM)
$$\tilde{p}(\mathbf{x}) = \exp\big(-E(\mathbf{x})\big),$$
where $E(x)$ is known as the **energy function**
- Note that $\exp(z)$ is positive for all z; by learning the energy function, we can use *unconstrained optimization* since we do not need to impose non-negative probability for any setting

Energy-Based Models (2)

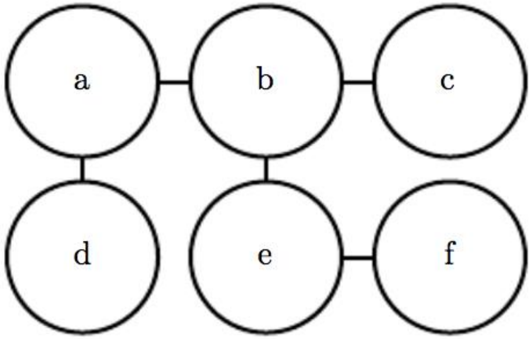
- **energy-based model** (EBM)

$$\tilde{p}(\mathbf{x}) = \exp\big(-E(\mathbf{x})\big)$$

- Any distribution given by the above equation is an example of Boltzman distribution; for this reason, many energy-based models are called Boltzman machines

Energy-Based Models (3)

- The following shows an example:



- It implies that $E(\mathrm{a,b,c,d,e,f})$ can be

$$E_{\mathrm{a,b}}(\mathrm{a,b}) + \cdots + E_{\mathrm{e,f}}(\mathrm{e,f})$$

- We can obtain the ϕ functions as

$$\phi_{\mathrm{a,b}}(\mathrm{a,b}) = \exp\big(-E(\mathrm{a,b})\big)$$

Separation (1)

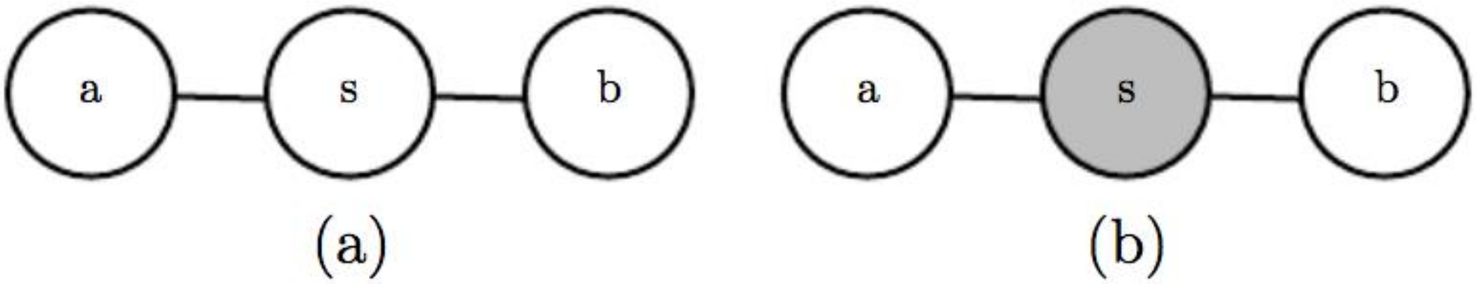
- The edges in a graphical model tell us *direct* interactions
- We often need to know *indirect* interactions
- More formally, we want to know *conditional independences* between the variables
- Conditional independence implied by undirected graph is called **separation**
- A set of variables A is separated by a set of variables B given a third set of variables S if the graph structure implies that A is independent from B given S

Separation (2)

- Two variables a and b are not separated when they are connected by a path involving only unobserved variables
- Two variables a and b are separated when
  - No path exists between them
  - All paths contain an observed variable
- Paths of only *unobserved* var. are "**active**"
- Paths including an *observed* var. are "**inactive**"

Separation (3)

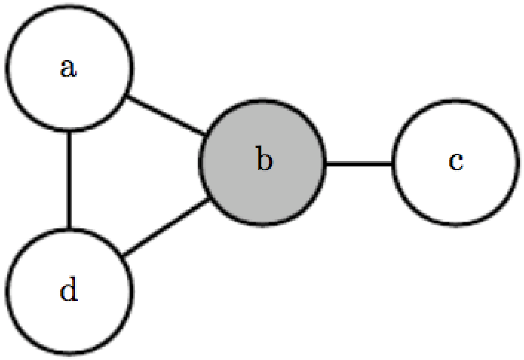
- The following shows an example:



- (a) a and b are not separated
  - The path is active because s is not observed
- (b) a and b are separated given s
  - The path is inactive because s is observed

Separation (4)

- The following shows another example:



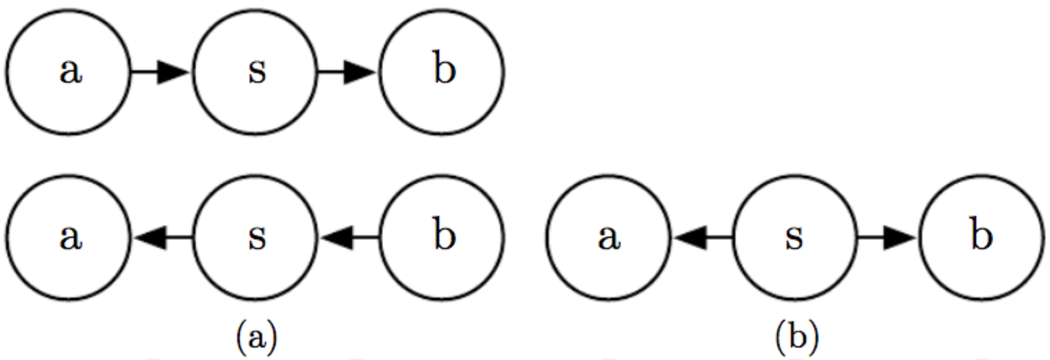
- Here b is shaded to indicate that it is observed
- a and c are separated from each other given b
- But, a and d are not separated given b
  - Since there is a second, active path between them

D-Separation (1)

- Similar concepts of separation apply to directed models
- These concepts are referred to as **d-separation**
- The “d” stands for “dependence”
- A set of variables A is d-separated by a set of variables B given a third set of variables S if the graph structure implies that A is independent from B given S

D-Separation (2)

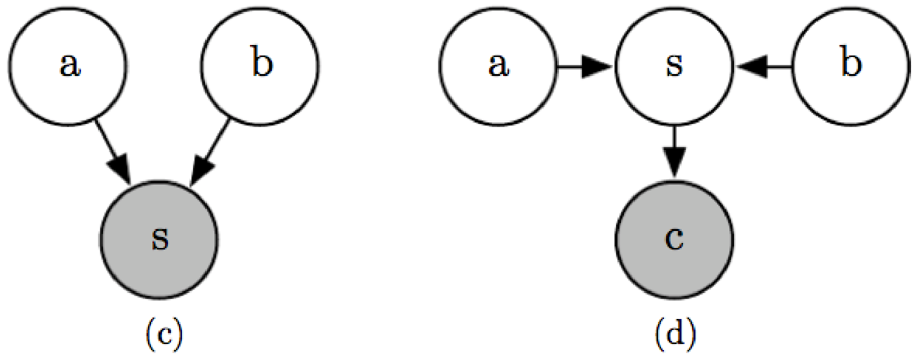
- The following shows an example:



- (a) This kind of path is blocked if s is observed
- (b) a and b are connected by a *common cause* s
  - This kind of path is also blocked if s is observed
- If s is not observed, than a and b are dependent

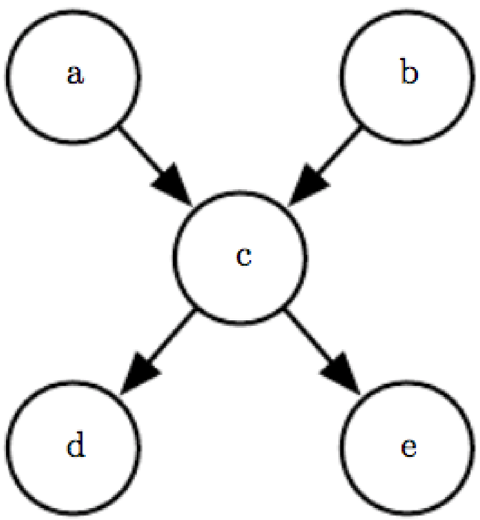
D-Separation (3)

- The following shows another example:



- (c) Variables a and b are both parents of s
  - This is called a **V-structure**, or **the collider case**
  - The path actually *active* when s is *observed*
- (d) That is the same in this case (descendant)

D-Separation (4)



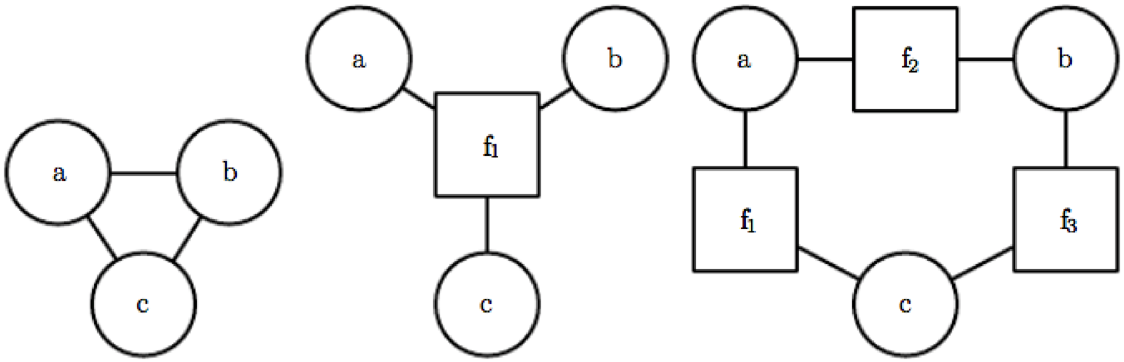
- Given the empty set:
  - a and b are d-separated
- Given c:
  - a and b are not d-sep.
  - a and e are d-separated
  - d and e are d-separated
- Given d:
  - a and b are not d-sep.

Factor Graphs (1)

- **Factor graphs** are another graphical models
- A factor graph is a bipartite undirected graph
- Some of the nodes are drawn as circles
  - These correspond to random variables
- The rest of the nodes are drawn as squares
  - These correspond to factors ϕ

Factor Graphs (2)

- The following show examples:



- *(Left)* An undirected network of a, b, and c
- *(Center)* A factor graph having one factor
- *(Right)* Another factor graph having three factors

Questions?