Conditional Random Fields

LING 572
Advanced Statistical Methods in NLP
February 11, 2020

Announcements

- HW4 grades out: 93.1 mean
- HW6 posted later today
  - Implement beam search
  - Note: pay attention to data format + feature vectors (in test time situation)
- Reading #2 posted!
  - Due Feb 18 **at 11AM**

Highlights

- CRF is a form of undirected graphical model
- Proposed by Lafferty, McCallum and Pereira in 2001
- Used in many NLP tasks: e.g., Named-entity detection
  - Often conjoined with neural models, e.g. LSTM + CRF
- Types:
  - Linear-chain CRF
  - Skip-chain CRF
  - General CRF

Outline

- Graphical models
- Linear-chain CRF
- Skip-chain CRF

Graphical models

Graphical model

- A graphical model is a probabilistic model for which a graph denotes the conditional independence structure between random variables:
  - Nodes: random variables
  - Edges: dependency relation between random variables

- Types of graphical models:
  - Bayesian network: directed acyclic graph (DAG)
  - Markov random fields: undirected graph

Bayesian network

Bayesian network

- Graph: directed acyclic graph (DAG)
  - Nodes: random variables
  - Edges: conditional dependencies
  - Each node X is associated with a probability function $P(X \mid \text{parents}(X))$

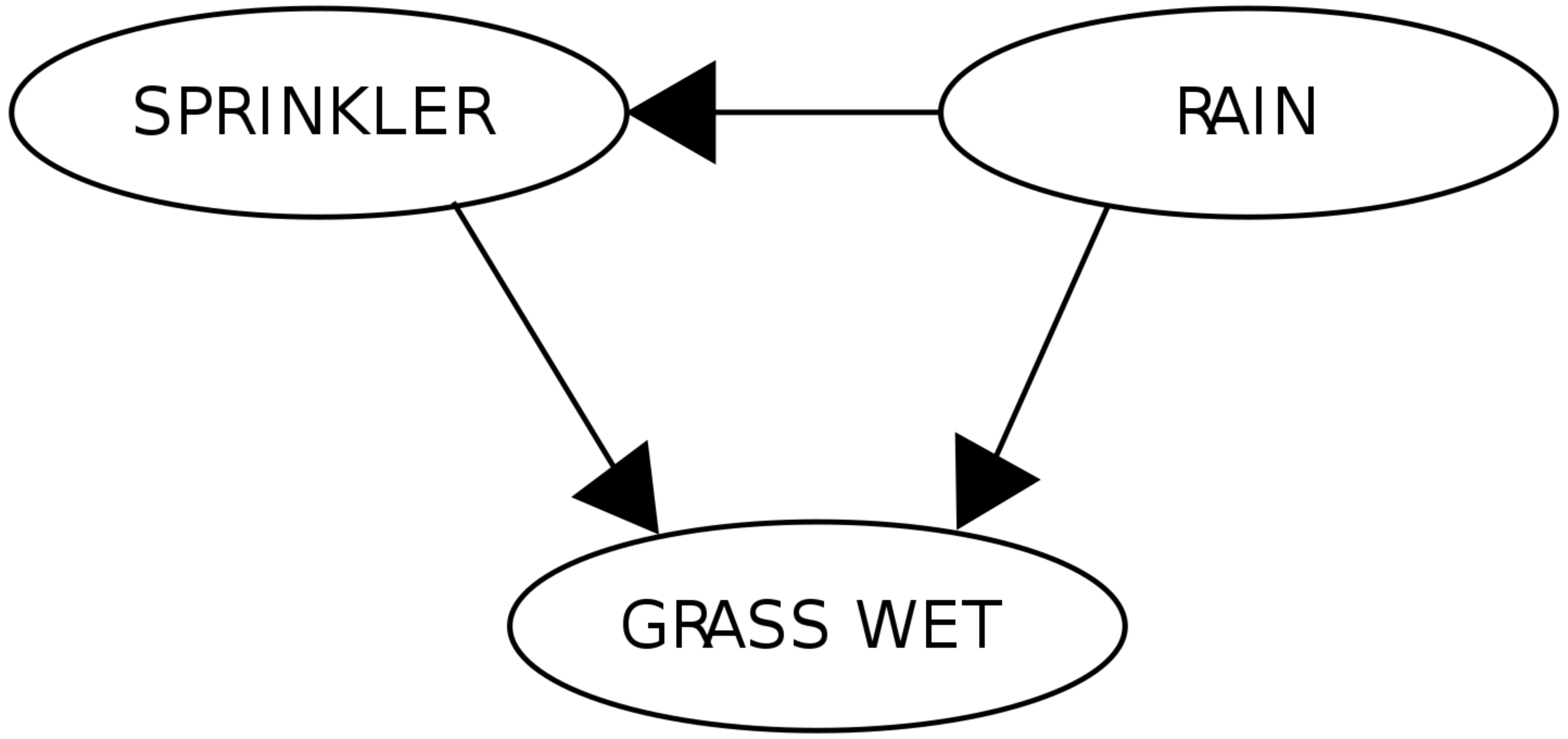
- Learning and inference: efficient algorithms exist.

An example

(from http://en.wikipedia.org/wiki/Bayesian_network)

	SPRINKLER	
RAIN	T	F
F	0.4	0.6
T	0.01	0.99

P(sprinkler | rain)



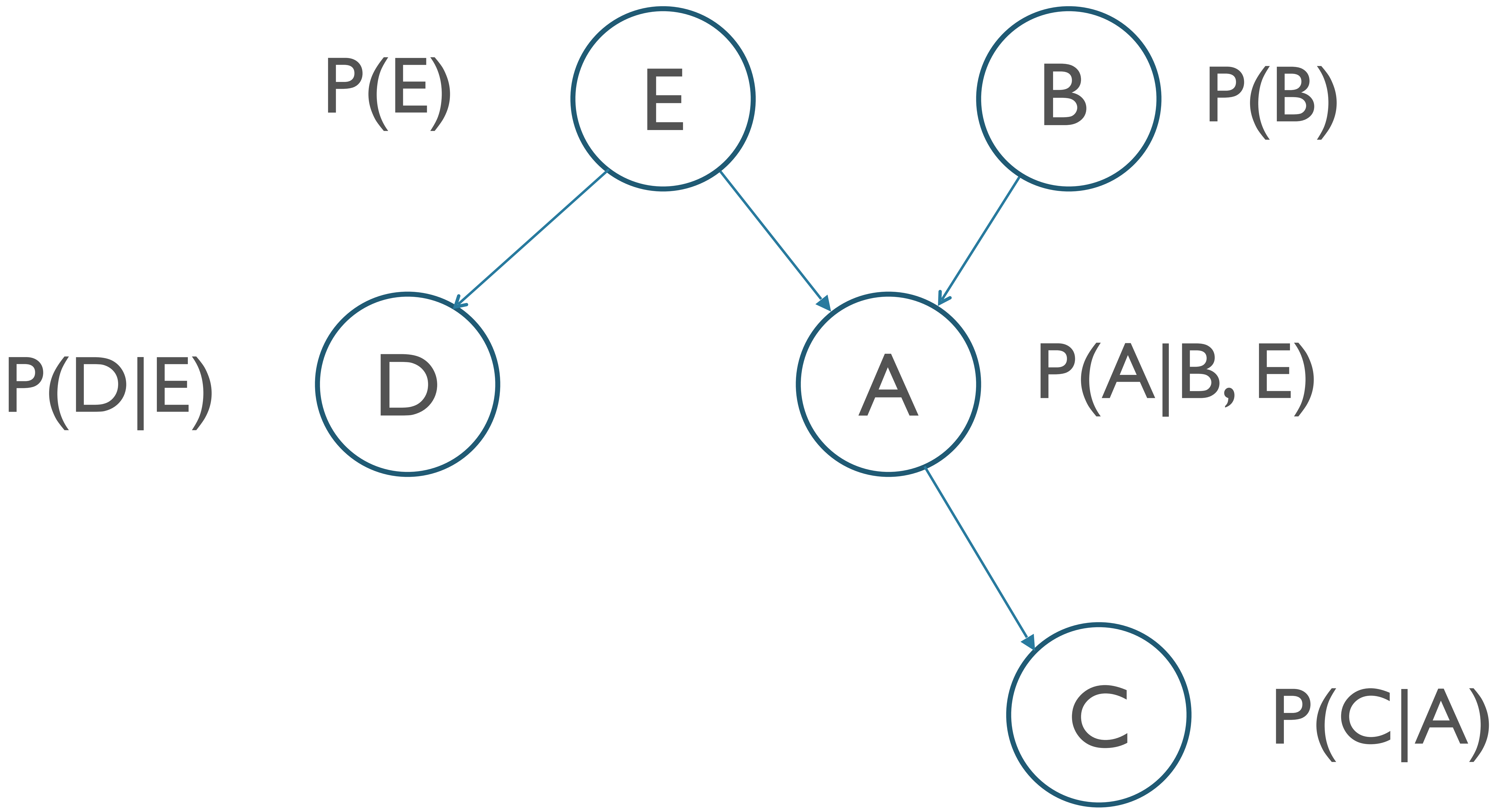
RAIN	
T	F
0.2	0.8

P(rain)

		GRASS WET	
SPRINKLER	RAIN	T	F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

P(grassWet | sprinkler, rain)

Another example

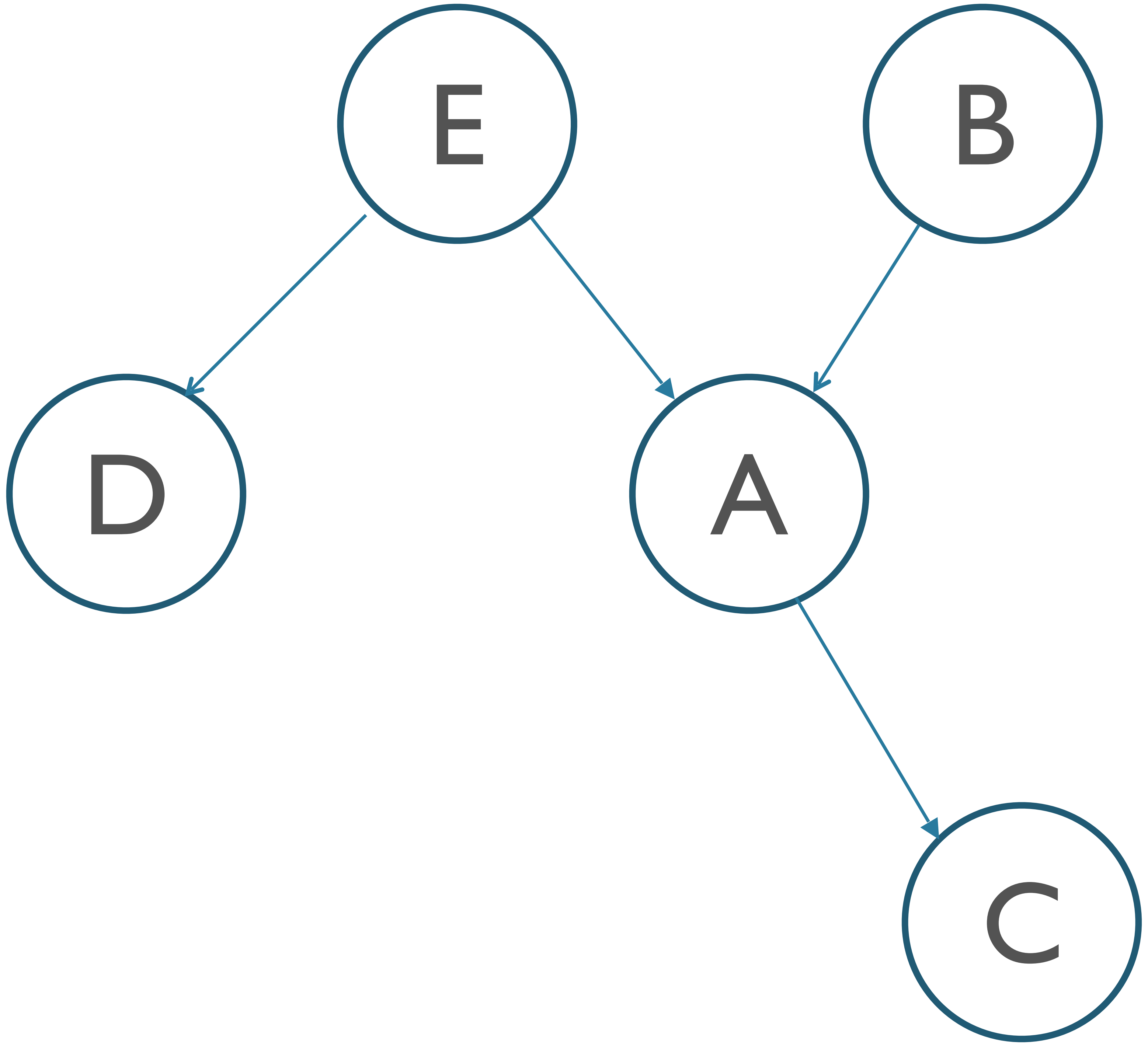


Bayesian network: properties

Local Markov property: each variable X_i is conditionally independent of its nondecendants given its parents variables.

$$P(X_1, ..., X_n)$$
$$= \prod_{i=1}^{n} P(X_i | X_1, ..., X_{i-1}))$$

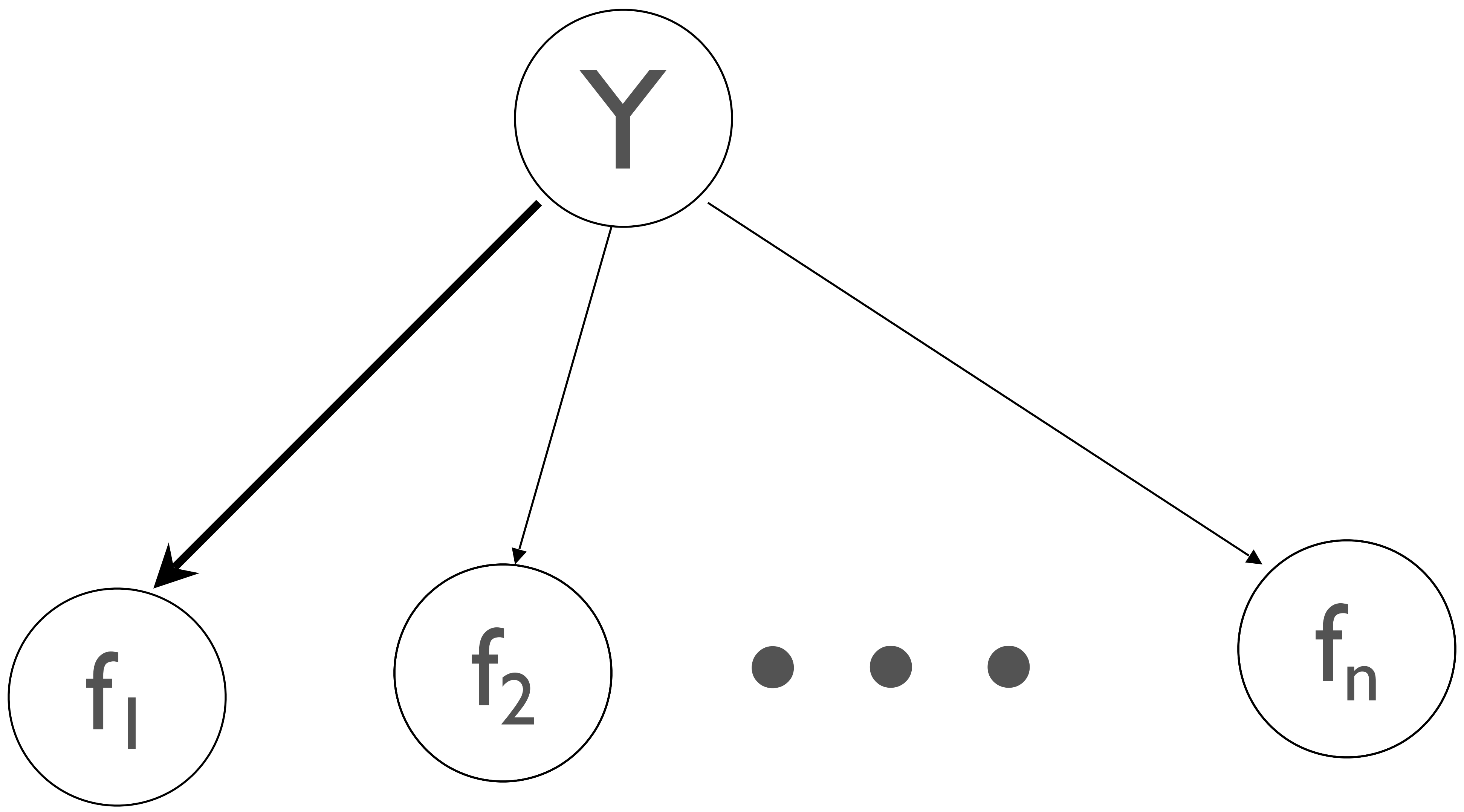
$$= \prod_{i=1}^{n} P(X_i | parents(X_i))$$



$$P(A,B,C,D,E) = P(B)P(E)P(A|B,E)P(C|A)P(D|E)$$

$$P(B,E|C,D) = \frac{P(B,E,C,D)}{P(C,D)} = \frac{\sum_A P(A,B,C,D,E)}{\sum_A \sum_B \sum_E P(A,B,C,D,E)}$$

Naïve Bayes Model

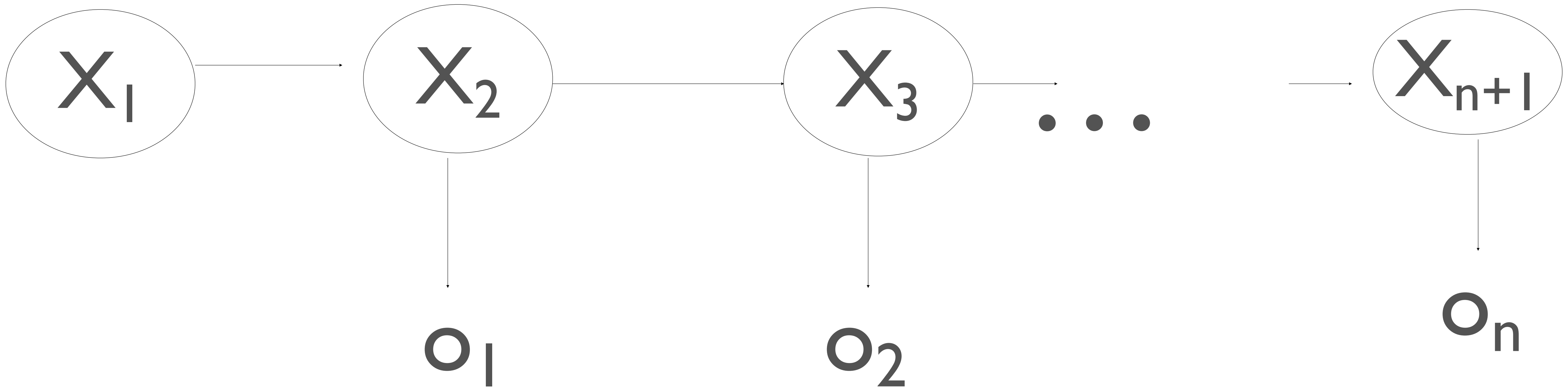


$$P(X,Y) = P(f1, f2, ..., f_n, Y)$$

$$= P(Y)P(f_1|Y)...P(f_n|Y)$$

$$= P(Y)\prod_{k=1}^{n} P(f_k|Y)$$

HMM



- State sequence: $X_{1,n+1}$
- Output sequence: $O_{1,n}$

$$P(O_{1:n}, X_{1:n+1}) = \pi(X_1) \prod_{i=1}^{n} P(X_{i+1} \mid X_i) P(O_i \mid X_{i+1})$$

Generative model

- A directed graphical model in which the output (i.e., what to predict) topologically precedes the input (i.e., what is given as observation).

- Naïve Bayes and HMM are generative models.

Markov Random Field

Markov random field

- Also called “Markov network”

- A graphical model in which a set of random variables have a Markov property:
  - Local Markov property: A variable is conditionally independent of all other variables given its neighbors.

$$P(X_i|X_j, ne(X_i)) = P(X_i|ne(X_i))$$

Cliques

- A **clique** in an undirected graph is a subset of its vertices such that every two vertices in the subset are connected by an edge.

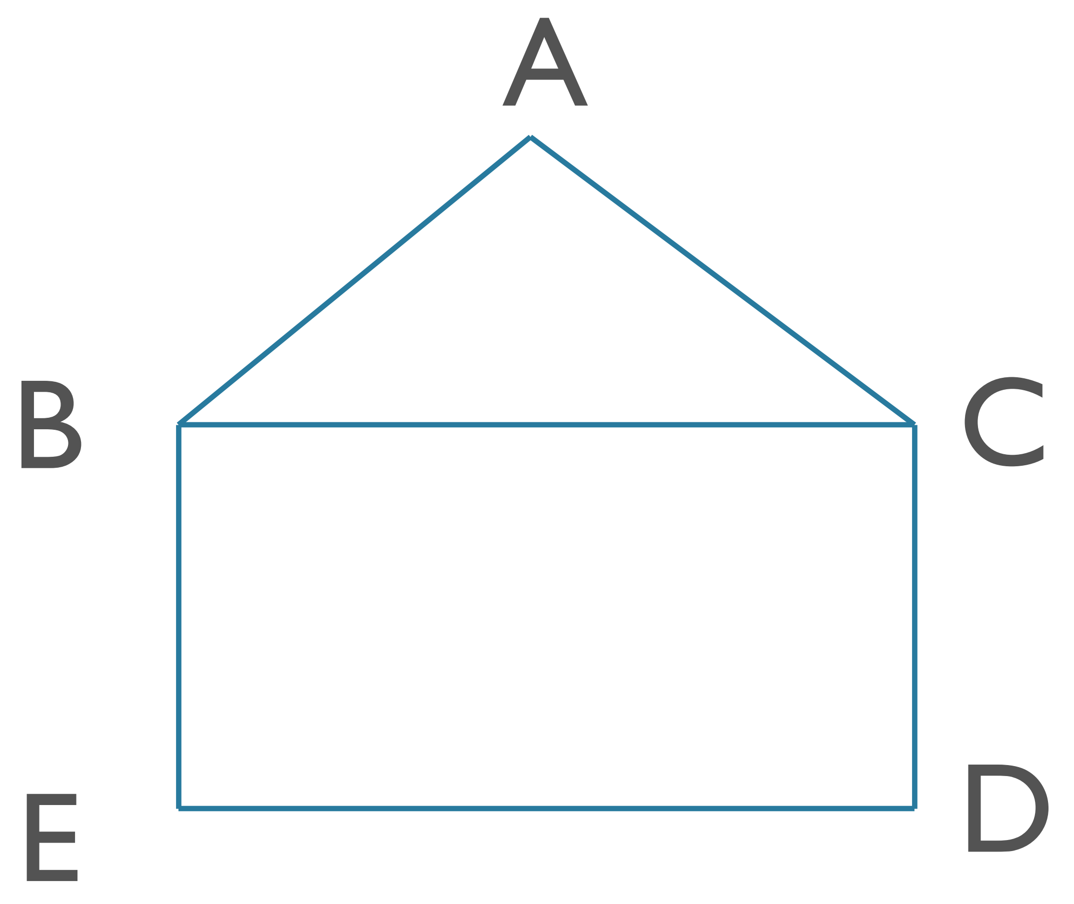
- A **maximal clique** is a clique that cannot be extended by adding one more vertex.

- A **maximum clique** is a clique of the largest possible size in a given graph.

clique:

maximum clique:

maximal clique:

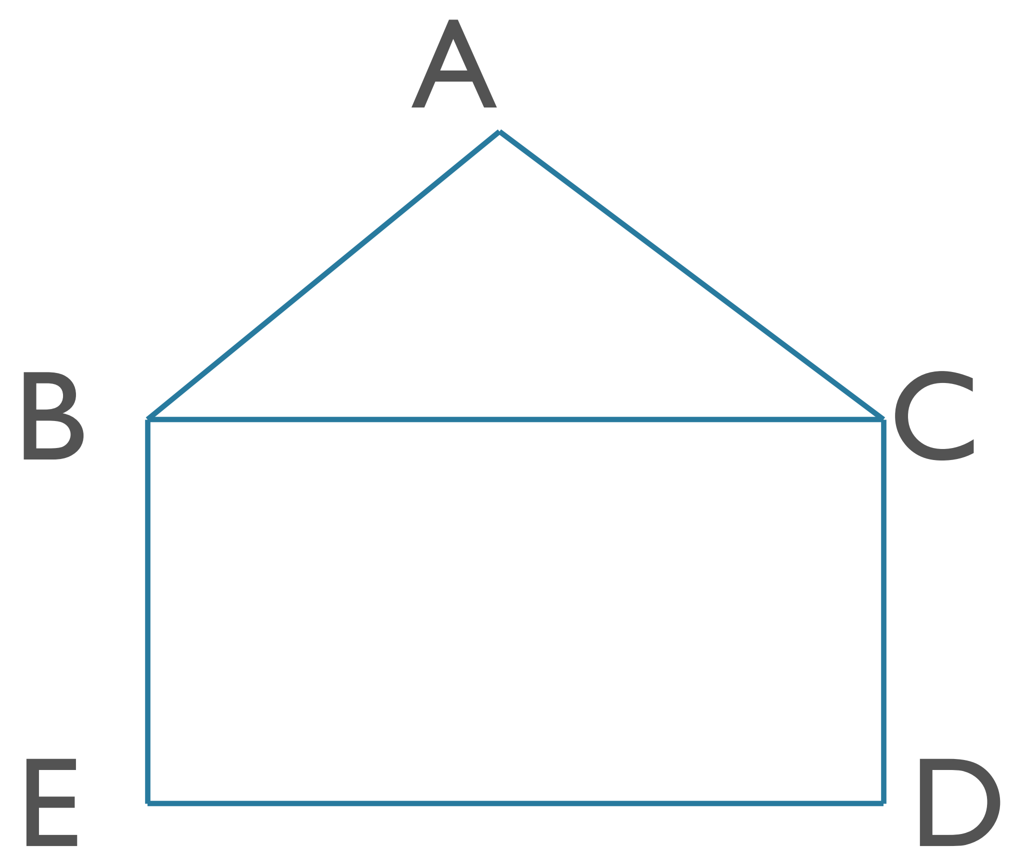


Clique factorization

$G = (V, E)$ be an undirected graph.

$cl(G)$ be the set of cliques of G.

$$P(X) = \frac{1}{Z} \prod_{C \in cl(G)} \phi_C(X_C)$$



$$P(A, B, C, D, E)$$

$$= \frac{1}{Z} \phi_{ABC}(A, B, C) \phi_{BE}(B, E) \phi_{DE}(D, E) \phi_{C,D}(C, D)$$

Conditional Random Field

Definition. *Let $G = (V, E)$ be a graph such that $\mathbf{Y} = (\mathbf{Y}_v)_{v \in V}$, so that $\mathbf{Y}$ is indexed by the vertices of G. Then $(\mathbf{X}, \mathbf{Y})$ is a conditional random field in case, when conditioned on $\mathbf{X}$, the random variables $\mathbf{Y}_v$ obey the Markov property with respect to the graph: $p(\mathbf{Y}_v \mid \boxed{\mathbf{X}}, \mathbf{Y}_w, w \neq v) = p(\mathbf{Y}_v \mid \boxed{\mathbf{X}}, \mathbf{Y}_w, w \sim v)$, where $w \sim v$ means that w and v are neighbors in G.*

A CRF is a random field globally conditioned on the observation X.

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{\Psi_A \in G} \exp\left\{ \sum_{k=1}^{K(A)} \lambda_{Ak} f_{Ak}(\mathbf{y}_A, \mathbf{x}_A) \right\}$$

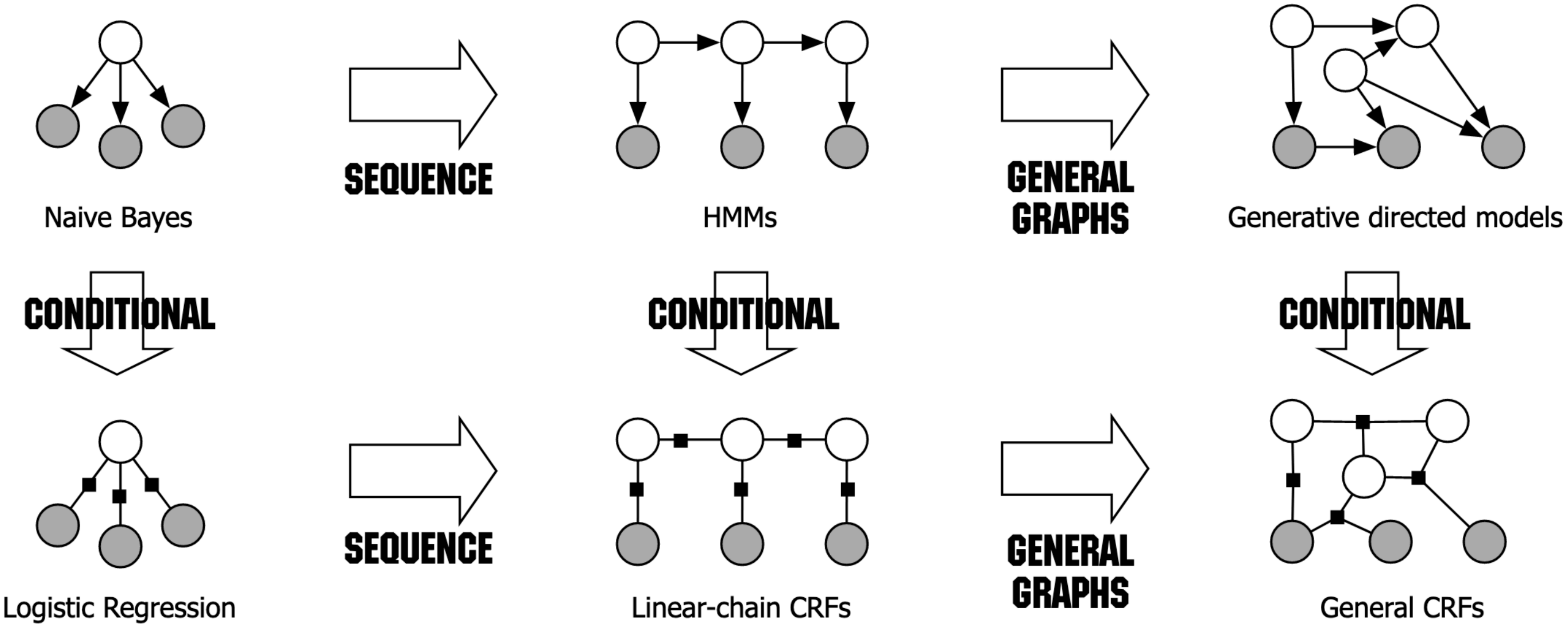
Linear-chain CRF

Motivation

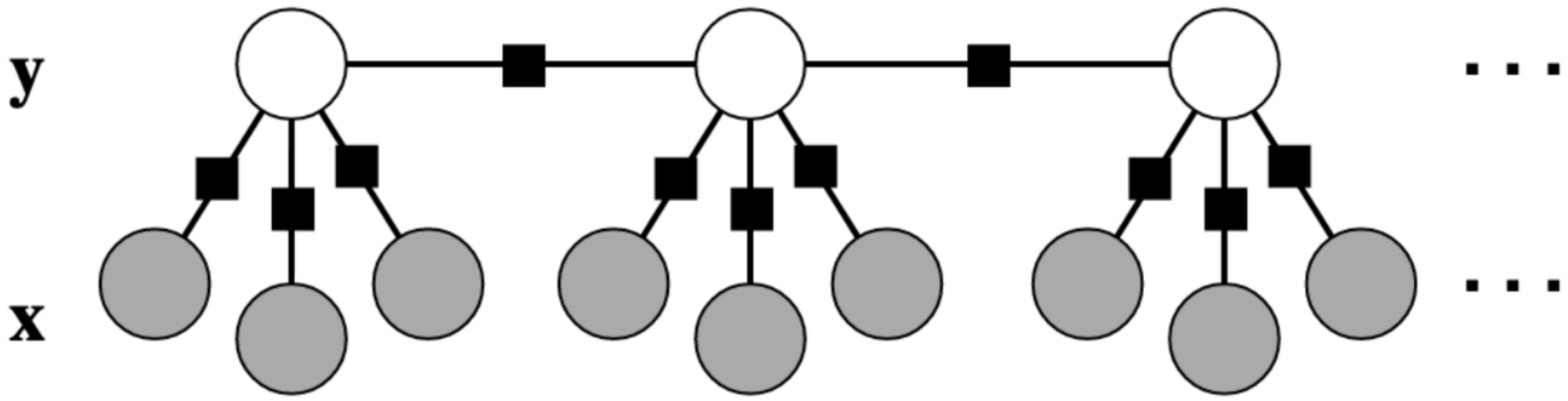
- Sequence labeling problem: e.g., POS tagging
  - HMM: Find best sequence, but cannot use rich features
  - MaxEnt: Use rich features, but may not find the best sequence

- Linear-chain CRF: HMM + MaxEnt

Relations between NB, MaxEnt, HMM, and CRF



Most Basic Linear-chain CRF



Linear-chain CRF (**)

$$f_j(y_{t-1}, y_t, x, t) = \begin{cases} 1 & (y_{t-1} = IN) \wedge (y_t = NNP) \wedge (x_t = Sept) \\ 0 & otherwise \end{cases}$$

$$F_j(y, x) = \sum_{t=1}^{T} f_j(y_{t-1}, y_t, x, t)$$

$$P(y|x) = \frac{1}{Z(x)} exp(\sum_j \lambda_j F_j(y, x))$$

$$= \frac{1}{Z(x)} exp(\sum_j (\lambda_j \sum_{t=1}^{T} f_j(y_t, y_{t-1}, x, t)))$$

$$= \frac{1}{Z(x)} exp(\sum_j \sum_{t=1}^{T} (\lambda_j f_j(y_t, y_{t-1}, x, t)))$$

$$= \frac{1}{Z(x)} exp(\sum_{t=1}^{T} \sum_j (\lambda_j f_j(y_t, y_{t-1}, x, t)))$$

$$= \frac{1}{Z(x)} \prod_{t=1}^{T} \boxed{exp(\sum_j (\lambda_j f_j(y_t, y_{t-1}, x, t)))}$$

$$= \frac{1}{Z(x)} \prod_{t=1}^{T} \phi_t(y_t, y_{t-1}, x)$$

Training and decoding

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} \phi_t(y_t, y_{t-1}, x)$$

$$\phi_t(y_t, y_{t-1}, x) = exp(\sum_j (\lambda_j f_j(y_t, y_{t-1}, x, t)))$$

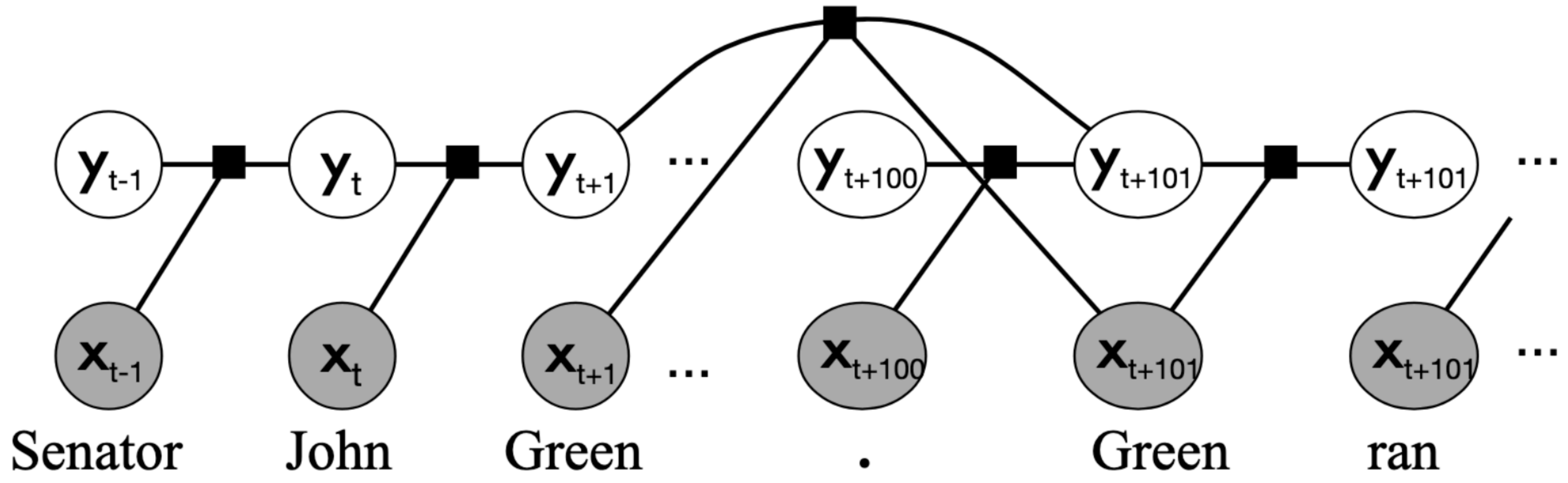
- Training: estimate λ_j
  - similar to the one used for MaxEnt
  - Ex: L-BFGS

- Decoding: find the best sequence y
  - similar to the one used for HMM
  - Viterbi algorithm

Skip-chain CRF

Motivation

- Sometimes, we need to handle long-distance dependency, which is not allowed by linear-chain CRF
- An example: NE detection
  - “Senator John Green …   Green ran …”



Linear-chain CRF:

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} \phi_t(y_t, y_{t-1}, x)$$

$$\phi_t(y_t, y_{t-1}, x) = exp(\sum_k (\lambda_k f_k(y_t, y_{t-1}, x, t)))$$

Skip-chain CRF:

$$P(y|x) = \frac{1}{Z(x)} \boxed{\prod_{t=1}^{T} \phi_t(y_t, y_{t-1}, x)} \boxed{\prod_{(u,v)\in D} \phi_{uv}(y_u, y_v, x)}$$

$$\phi_t(y_t, y_{t-1}, x) = exp(\sum_k (\lambda_k f_k(y_t, y_{t-1}, x, t)))$$

$$\phi_{uv}(y_u, y_v, x) = exp(\sum_k (\lambda_{2k} f_{2k}(y_u, y_v, x, u, v)))$$

CRFs in Larger Models

Semi-supervised sequence tagging with bidirectional language models

Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, Russell Power
Allen Institute for Artificial Intelligence
{matthewp,waleeda,chandrab,russellp}@allenai.org

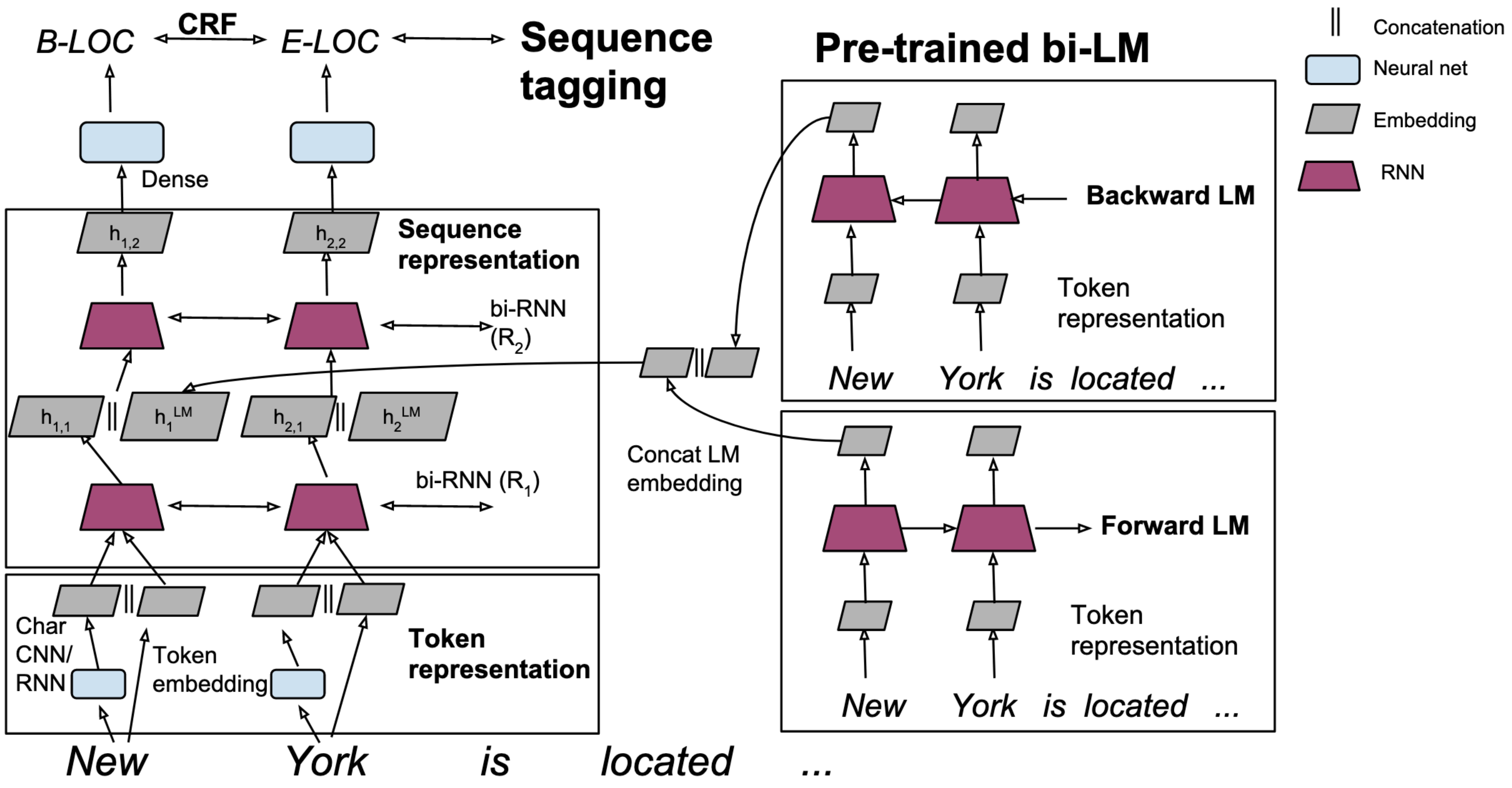
Abstract

Pre-trained word embeddings learned from unlabeled text have become a standard component of neural network architectures for NLP tasks. However, in most cases, the recurrent network that oper-

current neural network (RNN) that encodes token sequences into a context sensitive representation before making token specific predictions (Yang et al., 2017; Ma and Hovy, 2016; Lample et al., 2016; Hashimoto et al., 2016).

Although the token representation is initialized with pre-trained embeddings, the parameters of

CRFs in Larger Models



CoNLL 2003 (English)

The CoNLL 2003 NER task consists of newswire text from the Reuters RCV1 corpus tagged with four different entity types (PER, LOC, ORG, MISC). Models are evaluated based on span-based F1 on the test set. ♦ used both the train and development splits for training.

Model	F1	Paper / Source	Code
CNN Large + fine-tune (Baevski et al., 2019)	93.5	Cloze-driven Pretraining of Self-attention Networks	
RNN-CRF+Flair	93.47	Improved Differentiable Architecture Search for Language Modeling and Named Entity Recognition	
LSTM-CRF+ELMo+BERT+Flair	93.38	Neural Architectures for Nested NER through Linearization	Official
Flair embeddings (Akbik et al., 2018)♦	93.09	Contextual String Embeddings for Sequence Labeling	Flair framework
BERT Large (Devlin et al., 2018)	92.8	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	
CVT + Multi-Task (Clark et al., 2018)	92.61	Semi-Supervised Sequence Modeling with Cross-View Training	Official
BERT Base (Devlin et al., 2018)	92.4	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	
BiLSTM-CRF+ELMo (Peters et al., 2018)	92.22	Deep contextualized word representations	AllenNLP Project AllenNLP GitHub
Peters et al. (2017) ♦	91.93	Semi-supervised sequence tagging with bidirectional language models	

Source: NLP Progress

Summary

- Graphical models:
  - Bayesian network (BN)
  - Markov random field (MRF)

- CRF is a variant of MRF:
  - Linear-chain CRF: HMM + MaxEnt
  - Skip-chain CRF: can handle long-distance dependency
  - General CRF

- Pros and cons of CRF:
  - Pros: higher accuracy than HMM and MaxEnt
  - Cons: training and inference can be very slow