# Text Processing on the Web

## **Week 12**
## Sequence Labeling

Edited from source slides from Chua Tat-Seng's lecture notes with other contributions from Inna Weiner, Rongkun Shen and Jie Tang

# Recap

Clustering as unsupervised pattern learning

- Partitional methods
  - K means and variants
  - Speedups and approximations
  - Version of EM
- Hierarchical methods
  - HAC (bottom up) and HDC (top down)

- Selecting Labels for clusters
- Cluster Evaluation

# Text Analysis Example

Photo credit: markehr


## Singapore Flyer

**Singapore Flyer Pte Ltd** 30 Raffles Avenue, #01-07 Singapore 039803
Telephone: (65) 6854 5200 Fax: (65) 6339 9167

Singapore Flyer is the world's largest observation wheel. Standing at a stunning 165m from the ground, the Flyer offers you breathtaking, panoramic views of the Marina Bay, our island city and beyond. There's also a wide range of shops, restaurants, activities and facilities. READ MORE >>

- Information Units
  - IR: terms: raffles x 1; Singapore x 3; pte x 1 …
  - IE: info units: Singapore Flyer, Raffles Avenue, Marina Bay, (65) 6854-5200 … and their relations
  - QA: Which is the nearest MRT to Singapore Flyer?
    Answer: City Hall MRT
  - NLP: *understanding the contents*

# GATE Example NE tagging

GATE — General Architecture for Text Engineering

ANNIE Output for http://www.cnn.com/2007/WEATHER/03/29/spring.storm.ap/index.html

Annotation Key:
Person Location Organization Date Address Money Percent

## Twisters sweep Plains, kill 3

**Story Highlights**

? NEW: Tornado kills 2 in Oklahoma Panhandle; 1 in Colorado
? Eleven tornadoes reported in western Nebraska
? At least three injured in Texas Panhandle
? Watches, warnings stretch from South Dakota to Texas
Adjust font size:

OKLAHOMA CITY, Oklahoma (AP) -- An early spring storm swept across the Plains early Thursday, spinning off to Oklahoma couple in a home that was blown to pieces and a Colorado woman whose small town was nearly destroyed.

A tornado as wide as two football fields carved a destructive path through Holly, Colorado, late Wednesday, destroying dozens more and littering the streets with broken power lines, tree limbs and debris.

"Homes were there and now they're gone," county administrator Linda Fairbairn said. "Many, if not all, the structures in tc of damage."

A 28-year-old woman who suffered massive injuries during the twister died after she was airlifted to a hospital in Colorado County Coroner Joe Giadone said Thursday.

The line of storms stretched nearly the length of the nation, from South Dakota to Texas. As it headed east on Thursday

# Defining NER

- Identifying salient named entities in free text
  - PERSON – PLACE – NAMED OBJECTS
  - ORGANIZATION – QUANTITY – ADDRESSES
  - DATE/TIME – EVENT – UNITS OF MEASURE
  - And other **Domain-Specific** fields of interest (email address)

- Applications

  Moving from **doc** to **information** retrieval
  - QA answer typing: "What date was the last election on?": Election, GRCs, early April
  - As fundamental basis for IE: candidate text for output to templates, case frames, database records.
  - As a basis for summarization: focus-based summarization

# Characterizing NER

Can we cast NER in terms of another text processing task?

- It's a dictionary lookup problem – is it?
- It's a classification problem
- It's a pattern recognition problem
- It's unsupervised clustering

What's the difference?

- One critical difference: Chunks!

# Sequence Labeling

- Sequences of words as part of same entity
- Certainly not limited to NER

If viewing as TC, how would you classify them?

- One way: post-process to enforce constraints
- Maybe better: Chunk tagging
  - IOB scheme: Inside Outside Begin

# Example chunks

He/PRP saw/VBD the/DT big/JJ dog/NN

Begin Outside Begin Inside Inside

Pop quiz:

- What are the above tags trying to capture?
- What about End or Island tags? Do we even need a Begin tag?

# NER Approaches

- Use a machine learner, give lots of good features
- What are some good features? Let's go back to our example. Come on…

1. Internal
   1. Orthography
   2. Punctuation
   3. Gazetteer
   4. POS, Parsing information
2. External (Context)
   1. Nearby words

Not always applicable. Why?

Quite sensitive. Why?

# Internal Features

**Orthography**

- Capitalization
- Numerals
- Accents
- Formatting

Surprisingly good with End of Sentence detection.

**Punctuation**

- Clause markers: , . ! ; :
- Entity markers: $ % * ’s
- Often needs to be paired with context – why?

**Gazetteer**

Is sensitive to vocabulary, domain sensitivity

- Get features from a list
- Construct/infer list from a dictionary
- Often hierarchical: PLACE/RIVER
- Many names are also common words: Baker, Stone

**Grammatical / Syntactic Info**

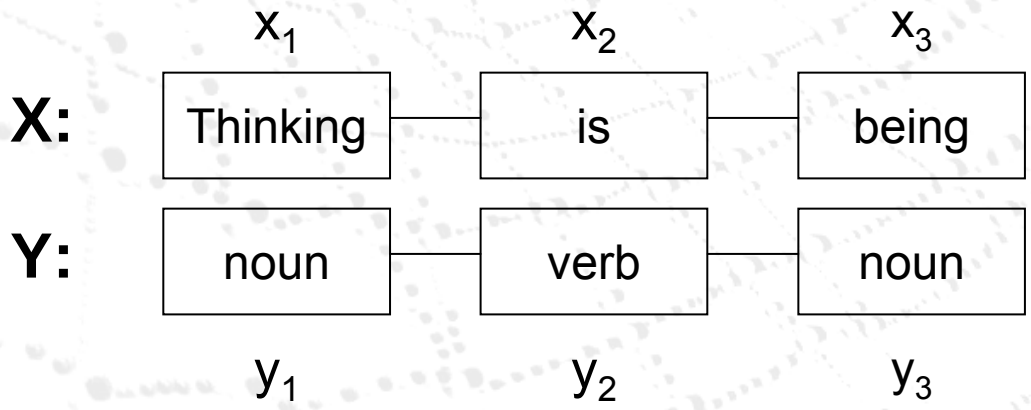
- Part of speech
- Phrase chunks (itself a chunk problem; base NPs)
- Parsing
- Layout
- Error prone (noisy) source

# Context

- Often, local decision dependent on decision in the near past
  - Try HMM (first order)
- But also the conditioned on future
  - E.g.,William B. Gates Foundation
  - Features need info on other decisions, nonlocal
- Context features often needed in combination
  - E.g., Previous detection of NE as some type X in document
  - E.g., $w_{-2}w_{-1}$ or $w_{-1}w_{+1}$
  - End up with many features, curse of dimensionality again
  - Back-off modeling

# Labeling Sequence Data

- X is a random variable over data sequences
- Y is a random variable over label sequences
- $Y_i$ is assumed to range over a finite label alphabet A
- The problem:
  - Learn how to give labels from a closed set Y to a data sequence **X**



# Generative Probabilistic Models

- Learning problem:

  Choose Θ to maximize *joint likelihood*:

$$L(\Theta)= \Sigma \log p_{\Theta}(y_i,x_i)$$

- The goal: maximization of the joint likelihood of training examples

  $y = \text{argmax } p^*(y|x) = \text{argmax } p^*(y,x)/p(x)$

  Joint likelihood

- Needs to enumerate all possible observation sequences

# Markov Model

A Markov process or model assumes that we can predict the future based just on the present (or on a limited horizon into the past):

Let $\{X_1,\ldots,X_T\}$ be a sequence of random variables taking values $\{1,\ldots,N\}$ then the Markov properties are:

1. Limited Horizon:

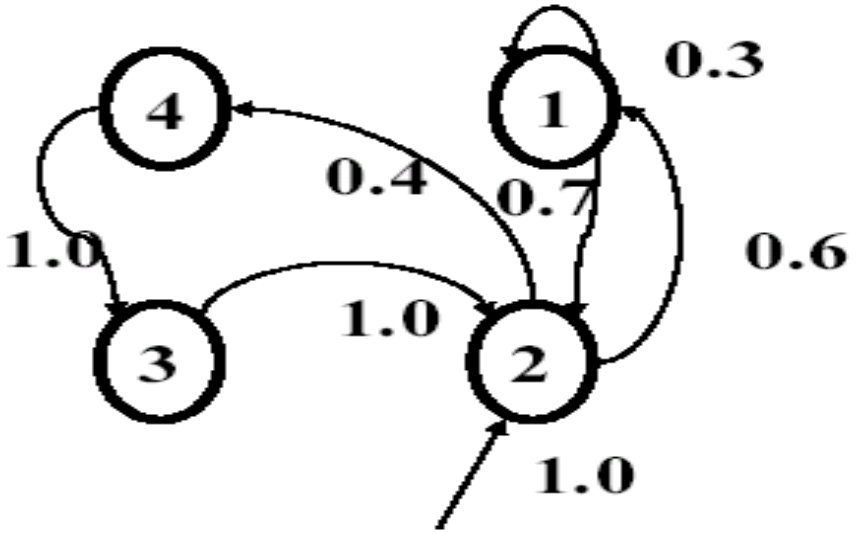
$$P(X_{t+1}|X_1,\ldots,X_t) = P(X_{t+1}|X_t) =$$

2. Time invariant (stationary):

$$= P(X_2|X_1)$$

# Describing a Markov Chain

Markov Chains can be described by the transition matrix A and the initial (start) probabilities Q:



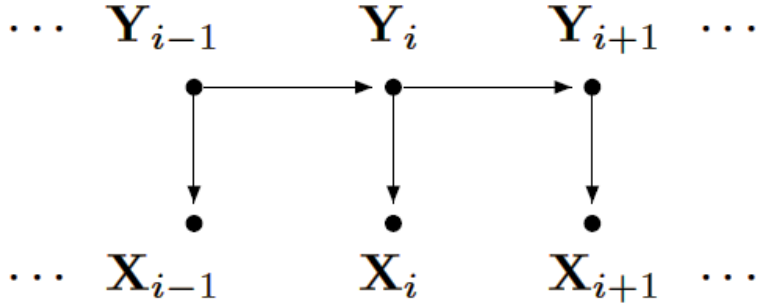

$A_{ij} = P(X_{t+1}=j|X_t=i)$

$q_i = P(X_1=i)$

# Hidden Markov Model

Do not observe the sequence that the model passes through (X) but only some probabilistic function of it (Y). Thus, it is a Markov model with the addition of *emission probabilities*:

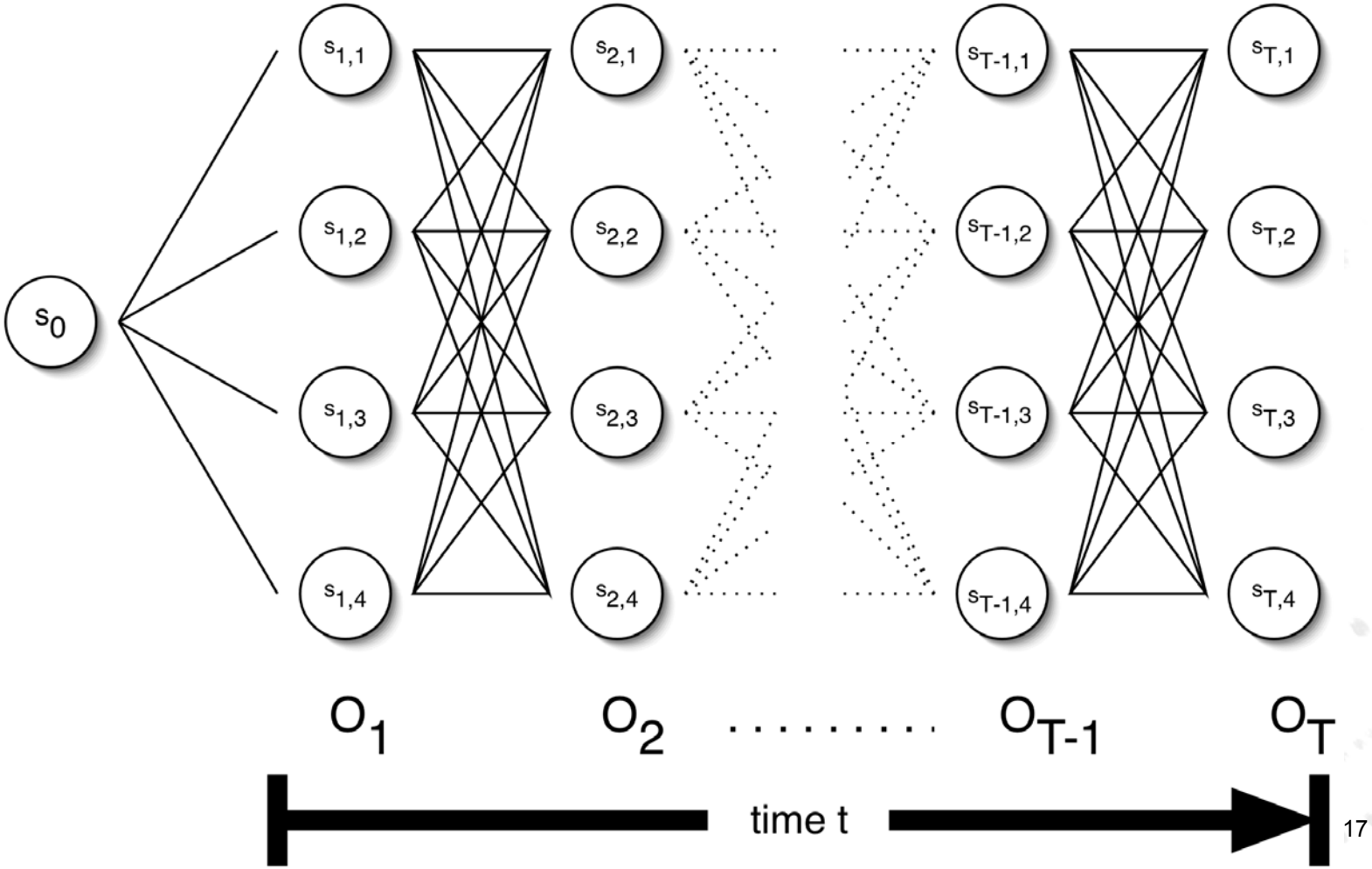
Standard tool is the hidden Markov Model (HMM).




**Hidden = Latent**

$$P(\mathbf{X}, \mathbf{Y}) = \prod_i P(\mathbf{X}_i \,|\, \mathbf{Y}_i)\, P(\mathbf{Y}_i \,|\, \mathbf{Y}_{i-1})$$

# The Trellis


$s_0$
$s_{1,1}$
$s_{1,2}$
$s_{1,3}$
$s_{1,4}$
$s_{2,1}$
$s_{2,2}$
$s_{2,3}$
$s_{2,4}$
$s_{T-1,1}$
$s_{T-1,2}$
$s_{T-1,3}$
$s_{T-1,4}$
$s_{T,1}$
$s_{T,2}$
$s_{T,3}$
$s_{T,4}$
$O_1$
$O_2$
$O_{T-1}$
$O_T$
time t

# The Three Problems in HMMs

- **Likelihood/Evaluation**: Given a series of observations y and a model λ = {A,B,q}, compute the likelihood p(y| λ)

  **>> Forward Algorithm**

- **Inference/Decoding**: Given a series of observations y and a model λ = {A,B,q}, compute the most likely sequence of hidden states x

  **>> Viterbi Algorithm (like forward algorithm but just do max instead of sum)**

- **Learning**: Given a series of observations, learn the best model λ

  **>> Forward-Backward Algorithm (Baum Welch)**

  **(Iterative algorithm to re-estimate parameters, like EM)**

# Likelihood in HMMs

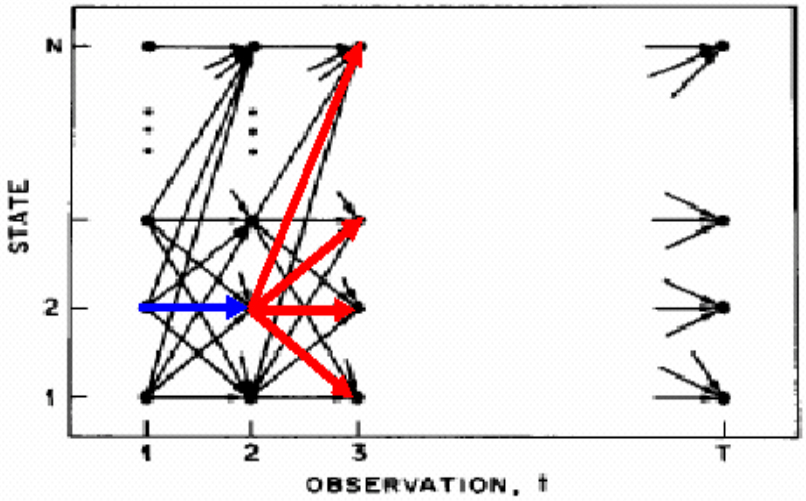
- **Given a model $\lambda = \{A,B,q\}$, we can compute the likelihood by**

$$
\begin{aligned}
P(y) &= p(y|\lambda) \\
&= \Sigma\, p(x)p(y|x) \\
&= q(x_1)\, \Pi A(x_{t+1}|x_t)\, \Pi B(y_t|x_t)
\end{aligned}
$$

- **But … this computation complexity is $O(N^T)$, when $|x_i| = N \rightarrow$ impossible in practice**

# Forward-Backward algorithm

- To compute likelihood:
  - –Need to enumerate over all paths in the lattice (all possible instantiations of $X_1 \ldots X_T$). But … some starting subpath (blue) is common to many continuing paths (blue+red)

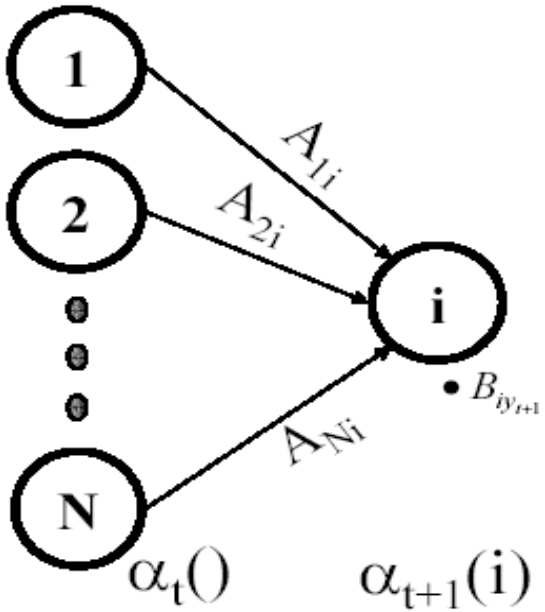



The idea:

Use **dynamic programming**, calculate a path in terms of shorter sub-paths

# Forward-Backward Algorithm (cont'd)

- We build a matrix of the probability of being at time t at state i: $\alpha_t(i) = P(x_t=i, y_1 y_2 \ldots y_t)$. This is a function of the previous column (forward procedure):




$$\alpha_1(i) = q_i B_{iy_1}$$

$$\alpha_{t+1}(i) = B_{iy_{t+1}} \sum_{j=1}^{N} \alpha_t(j) A_{ji}$$

$$P(Y) = \sum_{i=1}^{N} \alpha_T(i)$$

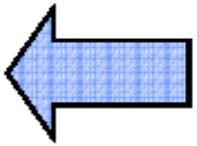
# Forward-Backward algorithm (cont'd)

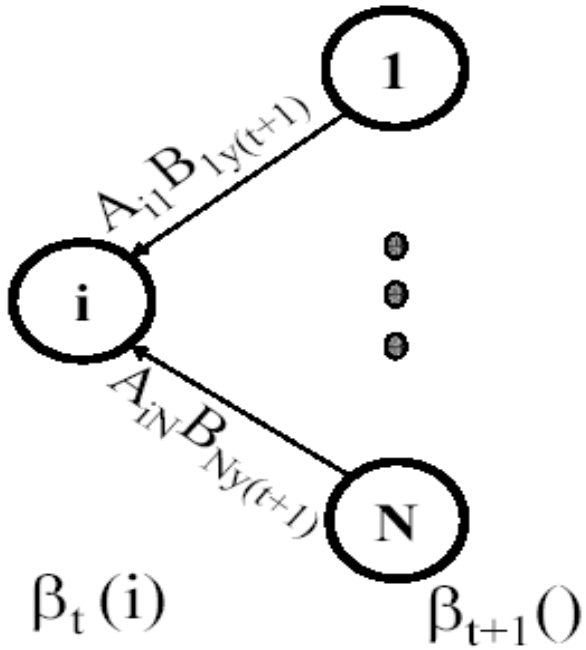
We can similarly define a backwards procedure for filling the matrix $\beta_t(i) = P(y_{t+1} \ldots y_T | x_t = i)$

$$\beta_T(i) = 1$$

$$\beta_t(i) = \sum_{j=1}^{n} A_{ij} B_{jy_{t+1}} \beta_{t+1}(j)$$

$$P(Y) = \sum_{i=1}^{N} q_i B_{iy_1} \beta_1(i)$$





# Combine both …

- **Combine both processes to arrive at likelihood:**

$$P(y,x_t=i) = P(x_t=i,y_1y_2\ldots y_t)^* P(y_{t+1}\ldots y_T|x_t=i)$$
$$= \alpha_t(i)\,\beta_t(i)$$

- **And then we get:**

$$P(y) = \Sigma\, P(y,x_t=i) = \Sigma\, \alpha_t(i)\,\beta_t(i)$$

# HMM Summary

- Advantages:
  - Estimation very easy
  - Closed form solution
  - The parameters can be estimated with relatively high confidence from small samples

- But:
  - The model represents all possible (x,y) sequences and defines **joint probability** over all possible observation and label sequences
  - Need to enumerate all possible observation sequences
  - Impossible to represent multiple interacting features
  - Difficult to model long-range dependencies of the observations
  - Very strict independence assumptions on the observations

# References

*Discussion of some of these in more detail next week*


- Baluja, Mittal and Sukthankar (99) *Applying Machine Learning for high performance named-entity extraction*
  - Good first read for ML approach to NE
- Zhou and Su (02) *Named Entity Recognition using an HMM-based Chunk Tagger*
  - More comprehensive set of analyses with more features. Explains chunk tags
- Maynard, Tablan and Cunningham (03) *NE Recognition Without Training Data on a Language You Don't Speak*
  - Case study of building NER in quick time without gazetteer
- Marnard, Tablan, Ursu, Cunningham and Wilks (01) *Named Entity Recognition from Diverse Text Types*
  - Sensitivity and portability of system over
- Pevzner and Hearst (02) *A Critique and Improvement of an Evaluation Metric for Text Segmentation*
  - Evaluation metrics for longer chunks
- CoNLL conferences - http://www.cnts.ua.ac.be/conll2003/
  - Includes scripts and software for evaluating chunk tagging