Decoding in Latent Conditional Models:
A Practically Fast Solution for an NP-hard Problem

Xu Sun (孫 栩)
University of Tokyo

2010.06.16

Outline

- Introduction
- Related Work & Motivations
- Our proposals
- Experiments
- Conclusions

Latent dynamics

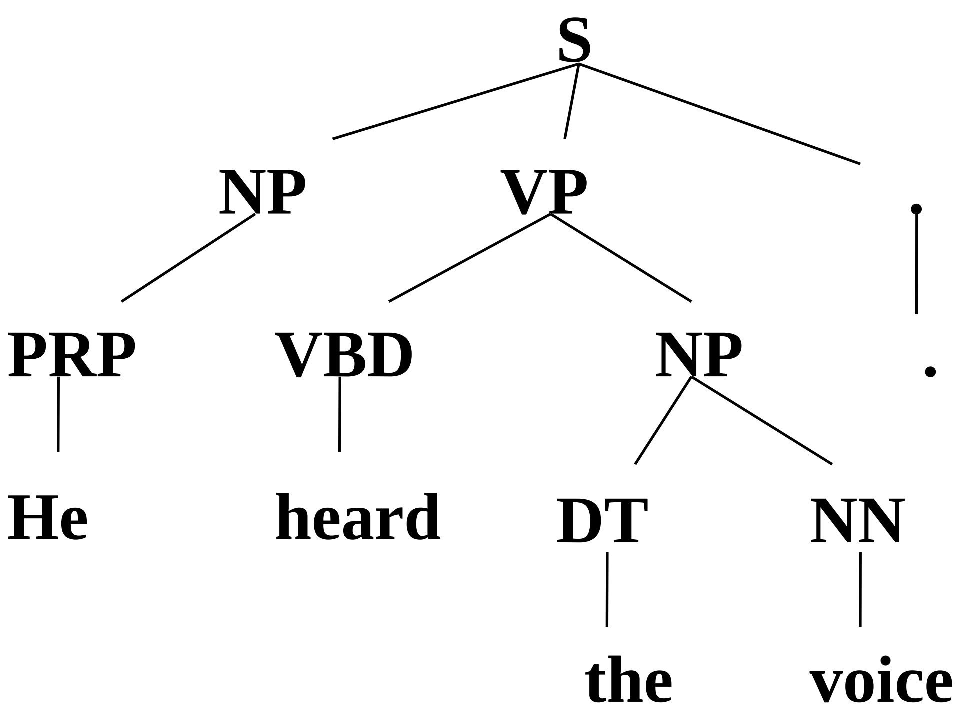
- Latent-structures (latent dynamics here) are important in information processing
  - Natural language processing
  - Data mining
  - Vision recognition

- Modeling latent dynamics: Latent-dynamic conditional random fields (LDCRF)

Latent dynamics

- Latent-structures (latent dynamics here) are important in information processing

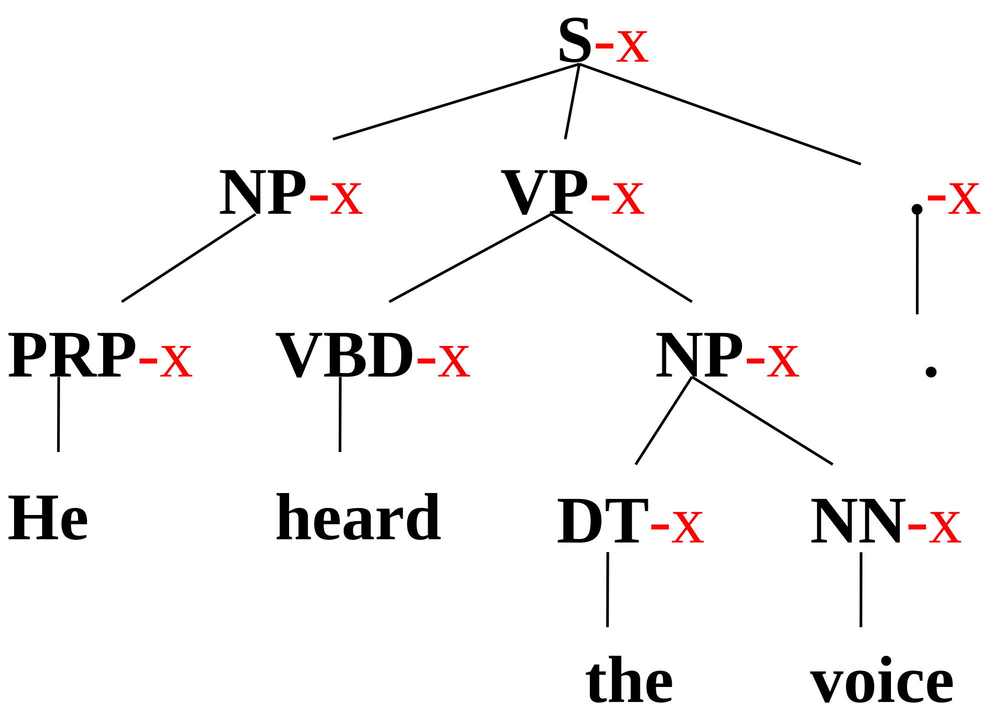
Parsing: Learn refined grammars with latent info



Latent dynamics

- Latent-structures (latent dynamics here) are important in information processing

Parsing: Learn refined grammars with latent info



More common cases: linear-chain latent dynamics

- The previous example is a tree-structure
- More common cases could be linear-chain latent dynamics
  - Named entity recognition
  - Phrase segmentation
  - Word segmentation

seg	seg	seg	noSeg
These	are	her	flowers.

Phrase segmentation [Sun+ COLING 08]

A solution without latent annotation: Latent-dynamic CRFs

A solution: Latent-dynamic conditional random fields (LDCRFs)

[Morency+ CVPR 07]

* No need to annotate latent info

seg	seg	seg	noSeg
These	are	her	flowers.

Phrase segmentation [Sun+ COLING 08]

Current problem & our target

A solution: Latent-dynamic conditional random fields (LDCRFs)

[Morency+ CVPR 07]

* No need to annotate latent info

Current problem:

Inference (decoding) is an NP-hard problem.

Our target:

An ***almost exact*** inference method with fast speed.

Outline

- Introduction
- <u>Related Work & Motivations</u>
- Our proposals
- Experiments
- Conclusions

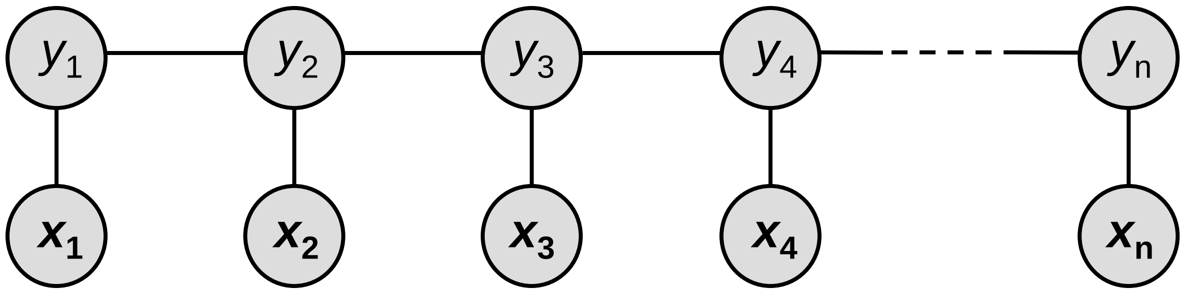
Traditional methods

- Traditional sequential labeling models
  - Hidden Markov Model (HMM)
    [Rabiner IEEE 89]
  - Maximum Entropy Model (MEM)
    [Ratnaparkhi EMNLP 96]
  - Conditional Random Fields (CRF)
    [Lafferty+ ICML 01]
  - Collins Perceptr
    [Collins

Arguably the most accurate one.
We will use it as one of the baseline.

Conditional random field (CRF)

[Lafferty+ ICML 01]

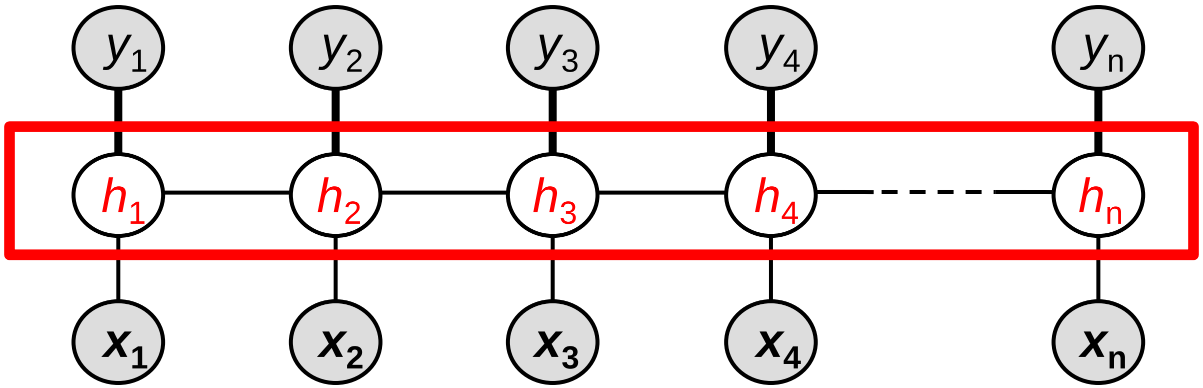


$$P(\boldsymbol{y} \mid \boldsymbol{x}, \theta) = \frac{1}{\mathcal{Z}(x, \theta)} \exp\left(\sum_k \theta_k \mathbf{F}_k (\boldsymbol{y}, \boldsymbol{x}) \right)$$

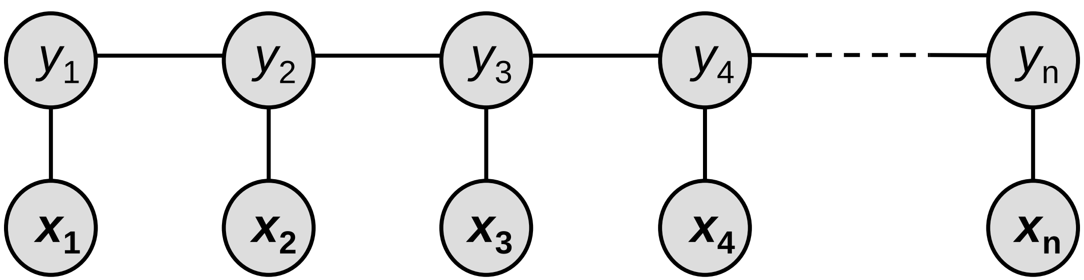
Problem: CRF does not model latent info

Latent-Dynamic CRFs

[Morency+ CVPR 07]

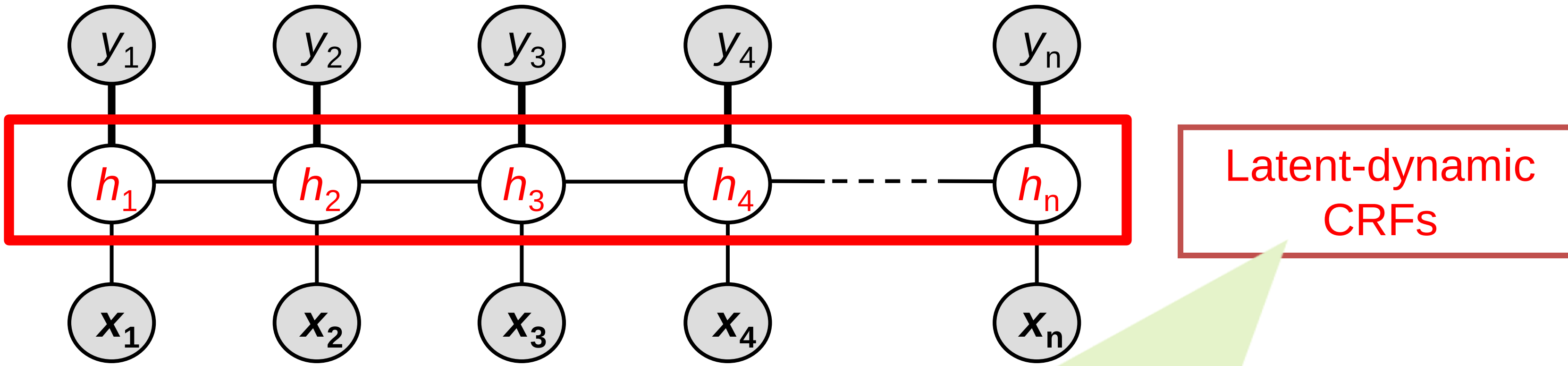


Latent-dynamic CRFs

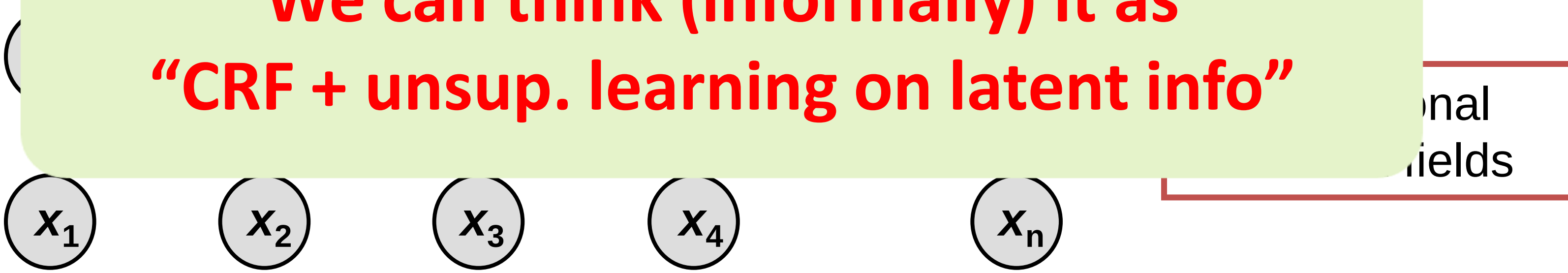


Conditional random fields

Latent-Dynamic CRFs
[Morency+ CVPR 07]



We can think (informally) it as “CRF + unsup. learning on latent info”



Latent-Dynamic CRFs
[Morency+ CVPR 07]

$$P(\boldsymbol{y} \mid \boldsymbol{x}, \theta) = \sum_{\boldsymbol{h}: \forall h_j \in \mathcal{H}_{y_j}} P(\boldsymbol{h} \mid \boldsymbol{x}, \theta) \quad = \sum_{\boldsymbol{h}: \forall h_j \in \mathcal{H}_{y_j}} \frac{1}{\mathcal{Z}(x, \theta)} \exp\left(\sum_k \theta_k \mathbf{F}_k(\boldsymbol{h}, \boldsymbol{x}) \right)$$

Good performance reports

* Outperforming HMM, MEMM, SVM, CRF, etc.
* Syntactic parsing [Petrov+ NIPS 08]
* Syntactic chunking [Sun+ COLING 08]
* Vision object recognition [Morency+ CVPR 07; Quattoni+ PAMI 08]

Outline

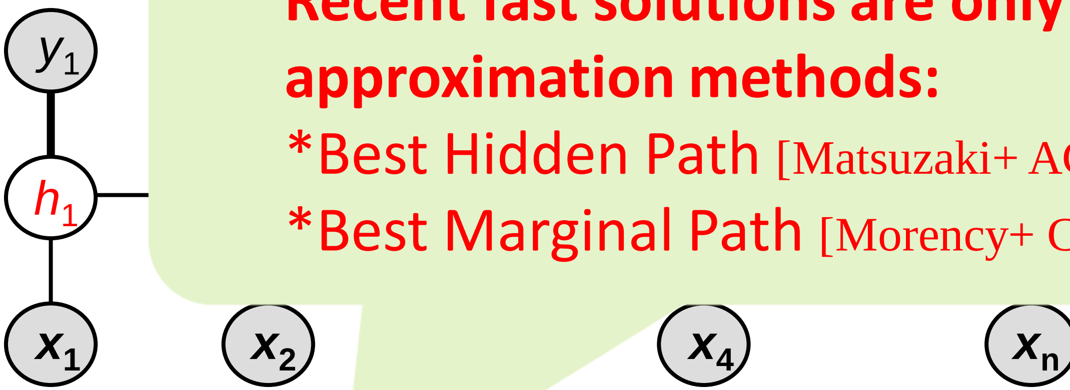
- Introduction
- Related Work & <u>Motivations</u>
- Our proposals
- Experiments
- Conclusions

Inference problem

Recent fast solutions are only approximation methods:

*Best Hidden Path [Matsuzaki+ ACL 05]

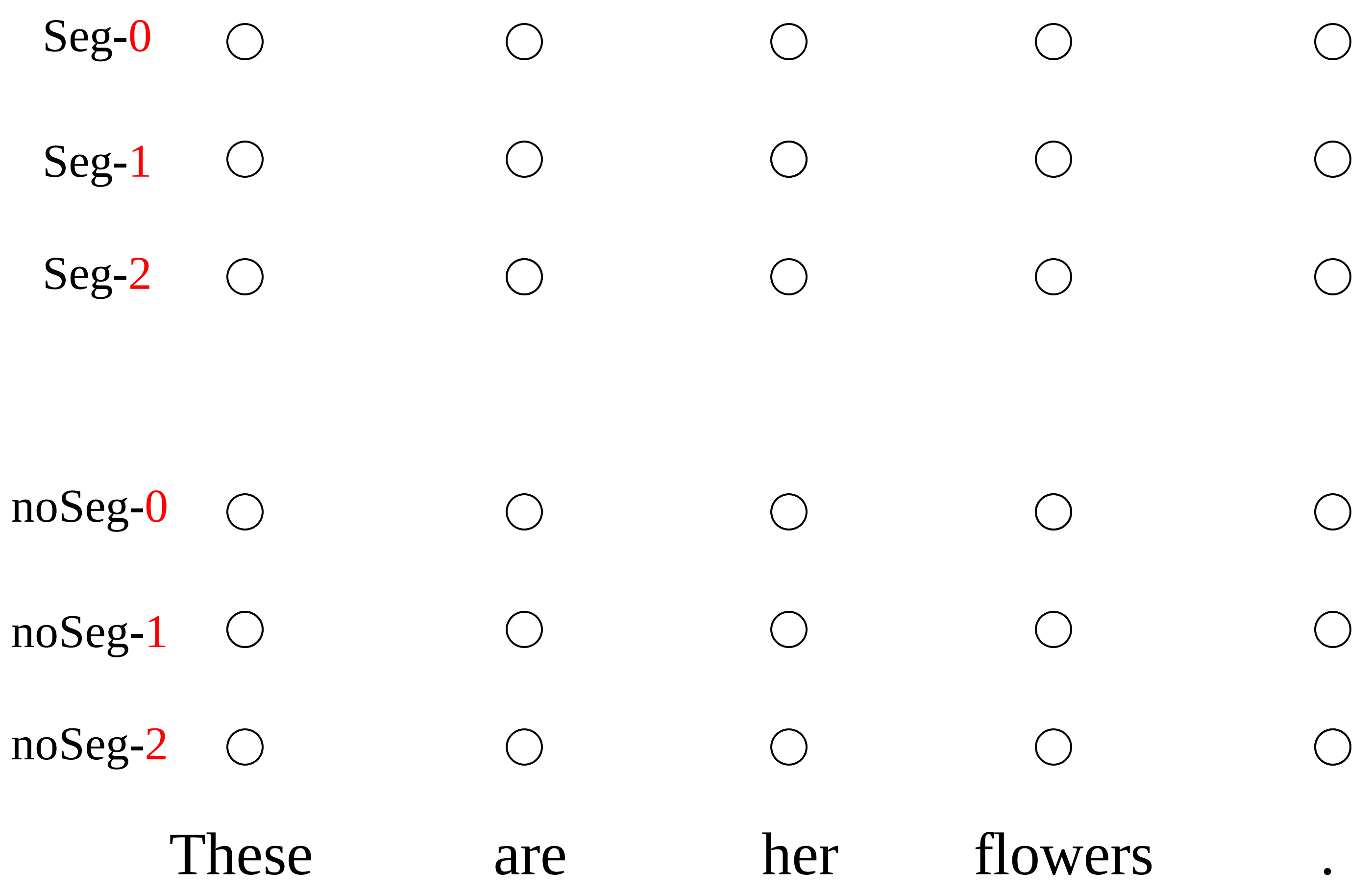
*Best Marginal Path [Morency+ CVPR 07]



- Prob: Exact inference (find the sequence with max probability) is NP-hard!
  - no fast solution existing

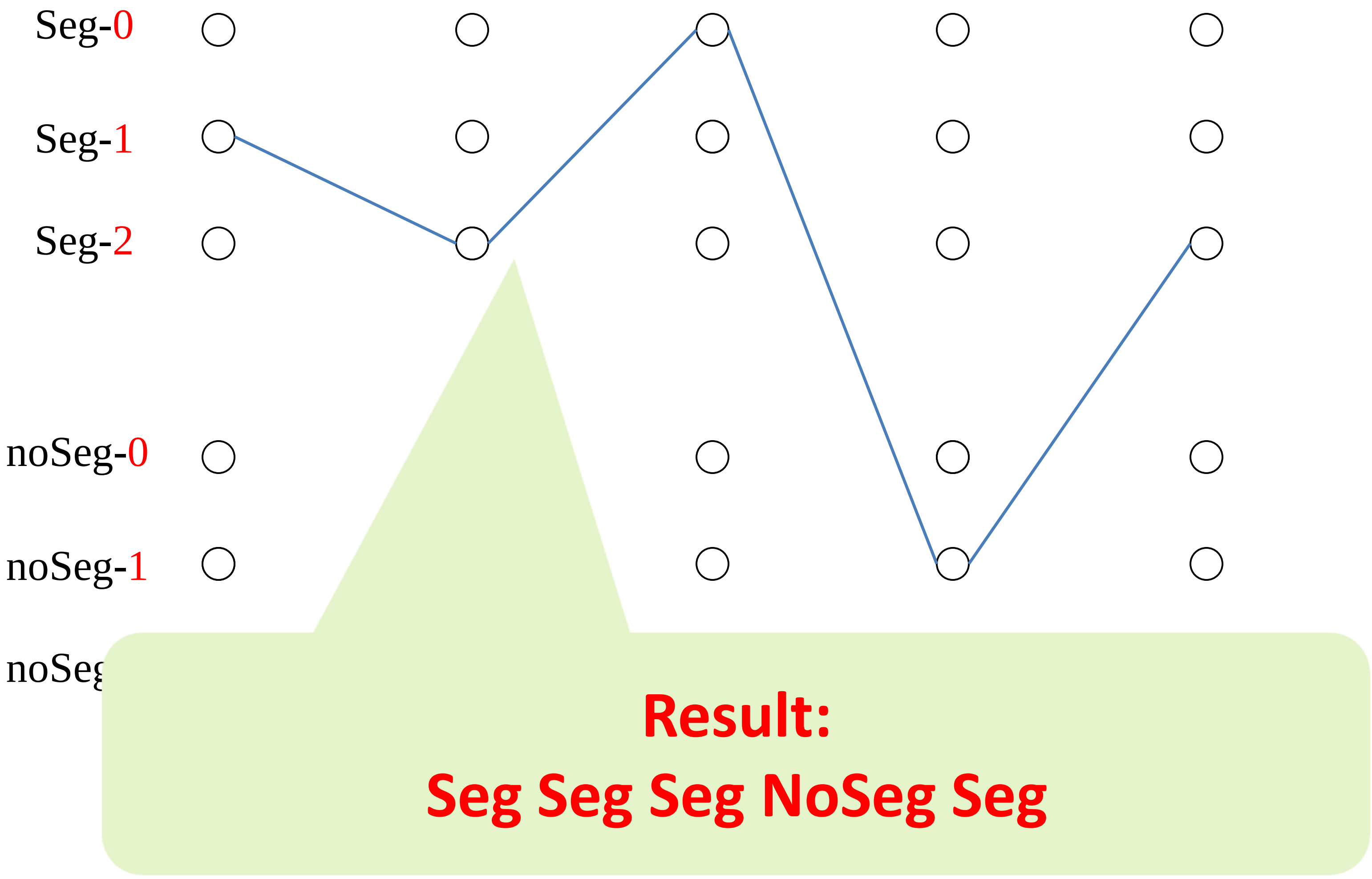
Related work 1: Best hidden path (BHP)

[Matsuzaki+ ACL 05]



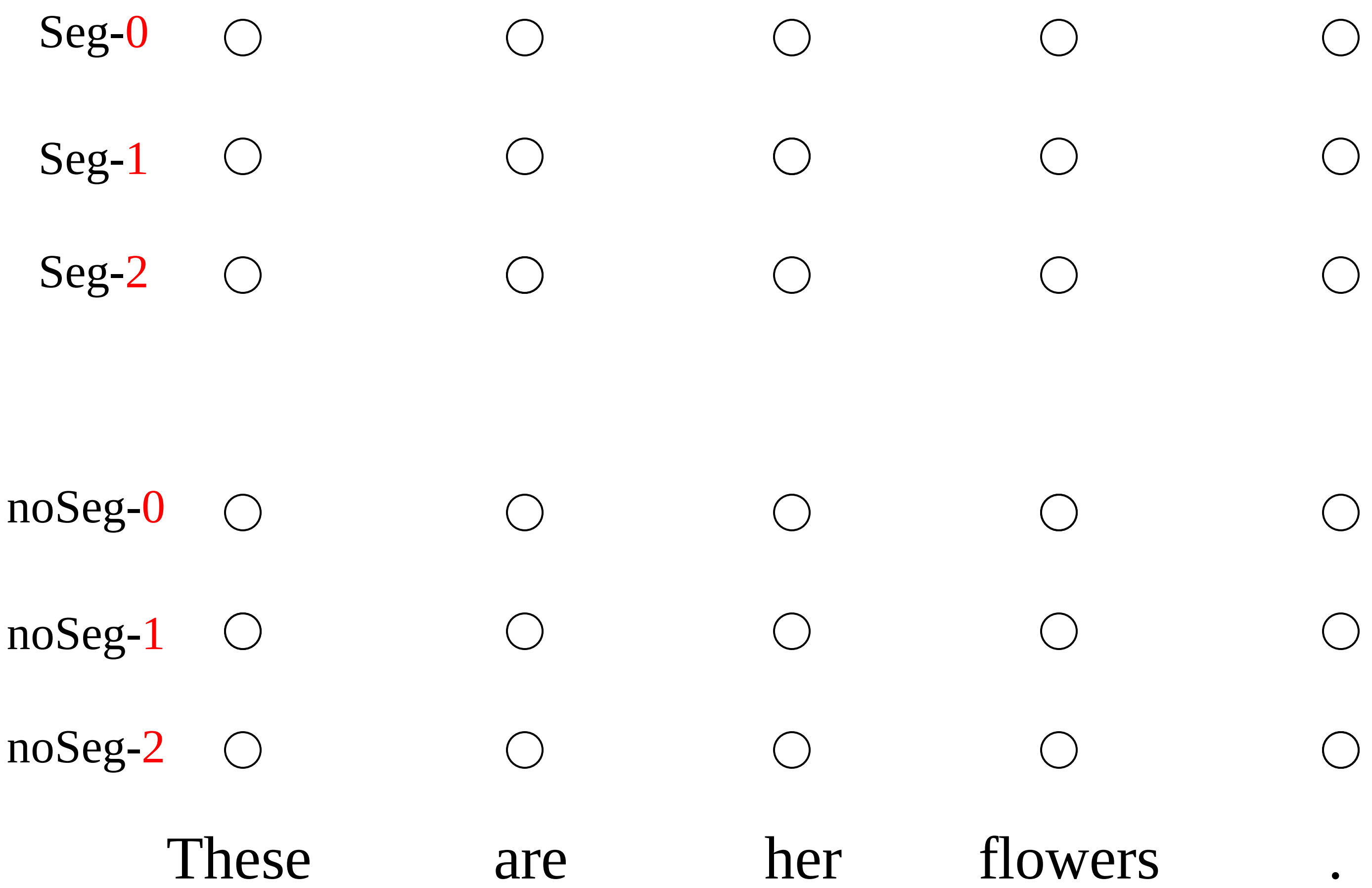
Related work 1: Best hidden path (BHP)

[Matsuzaki+ ACL 05]



Related work 2: Best marginal path (BMP)

[Morency+ CVPR 07]



Related work 2: Best marginal path (BMP)

[Morency+ CVPR 07]

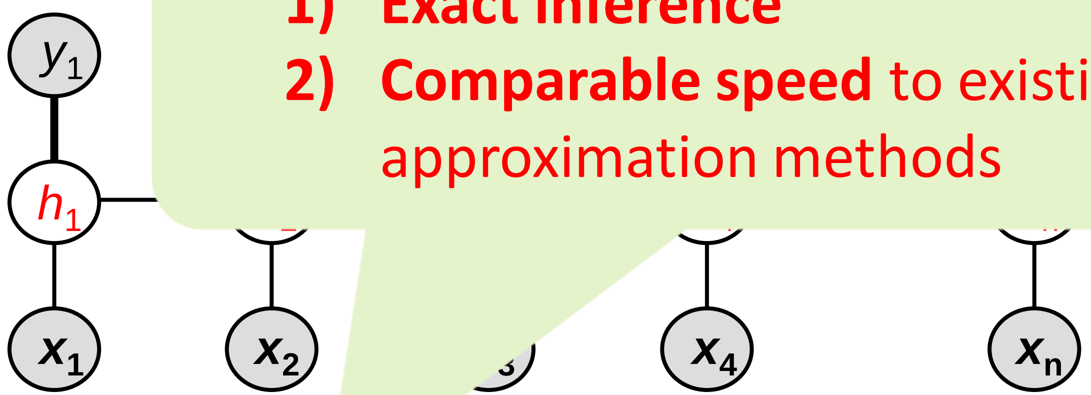
Seg-0	0.1	0.1	0.4	0.0	0.1
Seg-1	0.6	0.1	0.3	0.1	0.1
Seg-2	0.2	0.5	0.0	0.1	0.5
noSeg-0	0.1		0.2	0.1	0.2
noSeg-1	0.0		0.0	0.7	0.0
noSeg					

Result:

Seg Seg Seg NoSeg Seg

Our target

1) **Exact inference**
2) **Comparable speed** to existing approximation methods



Challenge/Difficulty:

Exact & practically-fast solution on an NP-hard problem

- Prob: E … ce with max pr…
  - no fas…

Outline

- Introduction
- Related Work & Motivations
- <u>Our proposals</u>
- Experiments
- Conclusions

Essential ideas

[Sun+ EACL 09]

- Fast & exact inference from a key observation
  - A key observation on prob. Distribution
  - Dynamic top-n search
  - Fast decision on optimal result from top-n candidates

Key observation

- Natural problems (e.g., NLP problems) are not completely ambiguous
- Normally, Only a few result candidate are highly probable
- Therefore, probability distribution on latent models could be sharp

Key observation

- Probability distribution on latent models is sharp

These	are	her	flowers	.	
seg	noSeg	seg	seg	seg	P = 0.2
seg	seg	seg	noSeg	seg	P = 0.3
seg	seg	seg	seg	seg	P = 0.2
seg	seg	noSeg	noSeg	seg	P = 0.1
seg	noSeg	seg	noSeg	seg	P = …
…	…	…	…	…	P = …

0.8 prob

Key observation

- Pr
  sh

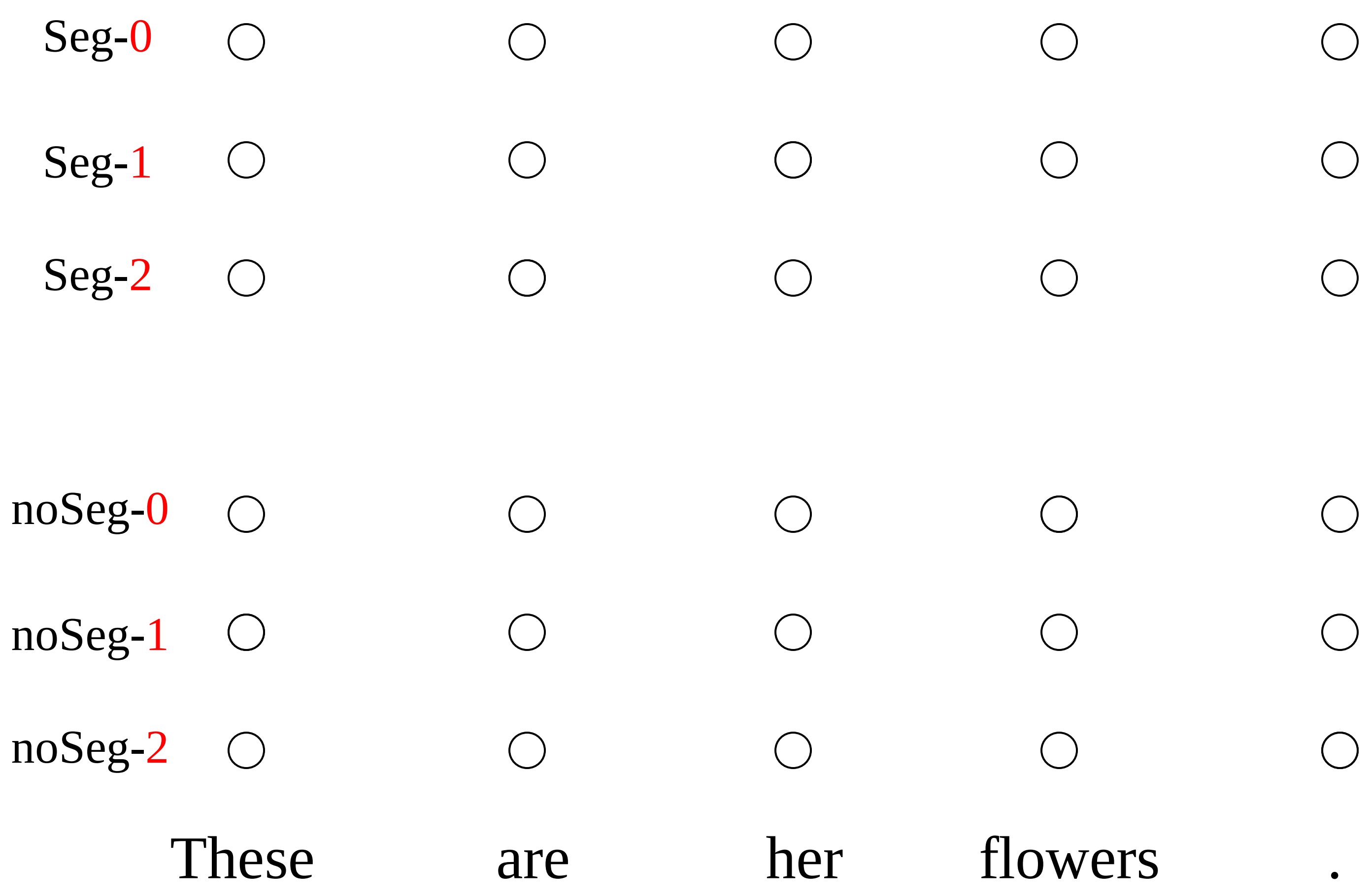
- Challenge: the number of ***probable*** candidates are **unknown & changing**
- Need a method which can **automatically adapt** itself on different cases

Thes					
seg	noSeg	seg	seg	seg	$P = 0.2$
seg	seg	seg	noSeg	seg	$P = 0.3$
seg	seg	seg	seg	seg	$P = 0.2$
seg	seg	noSeg	noSeg	seg	$P = 0.1$
seg	noSeg	seg	noSeg	seg	$P = \ldots$
…	…	…	…	…	$P = \ldots$

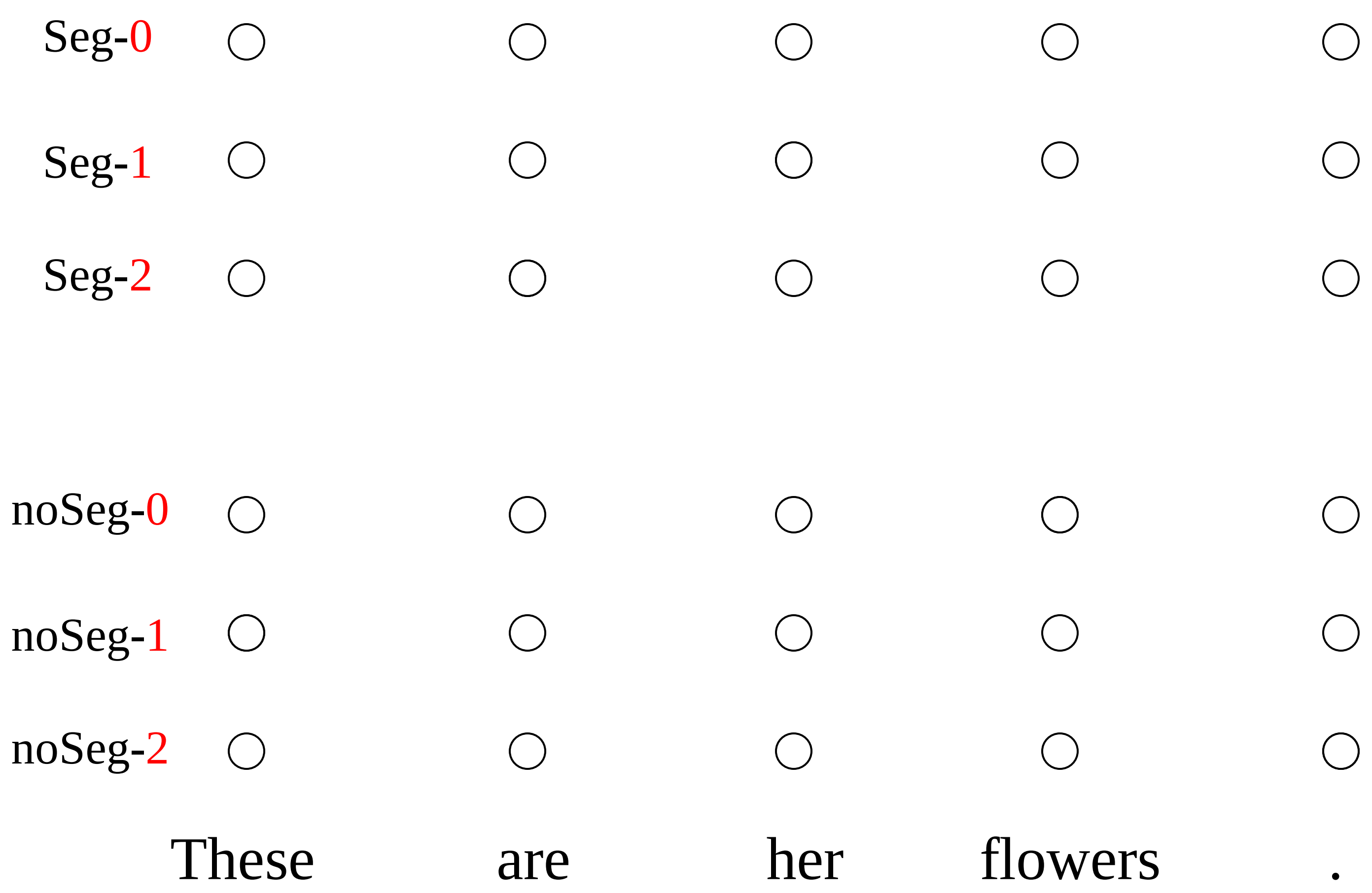
compare

$P(\text{unknown}) \leq 0.2$

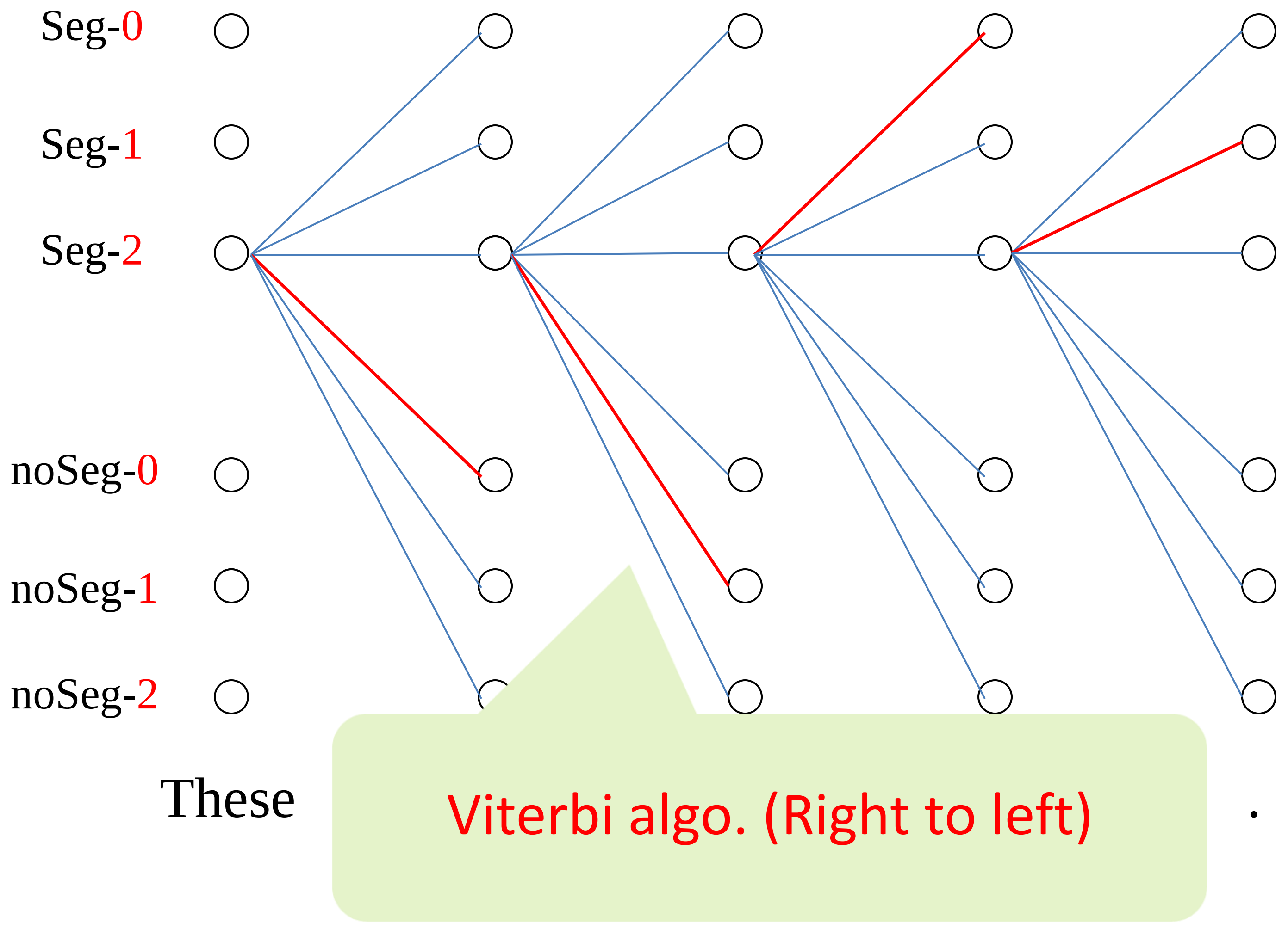
A demo on lattice



(1) Admissible heuristics for A* search



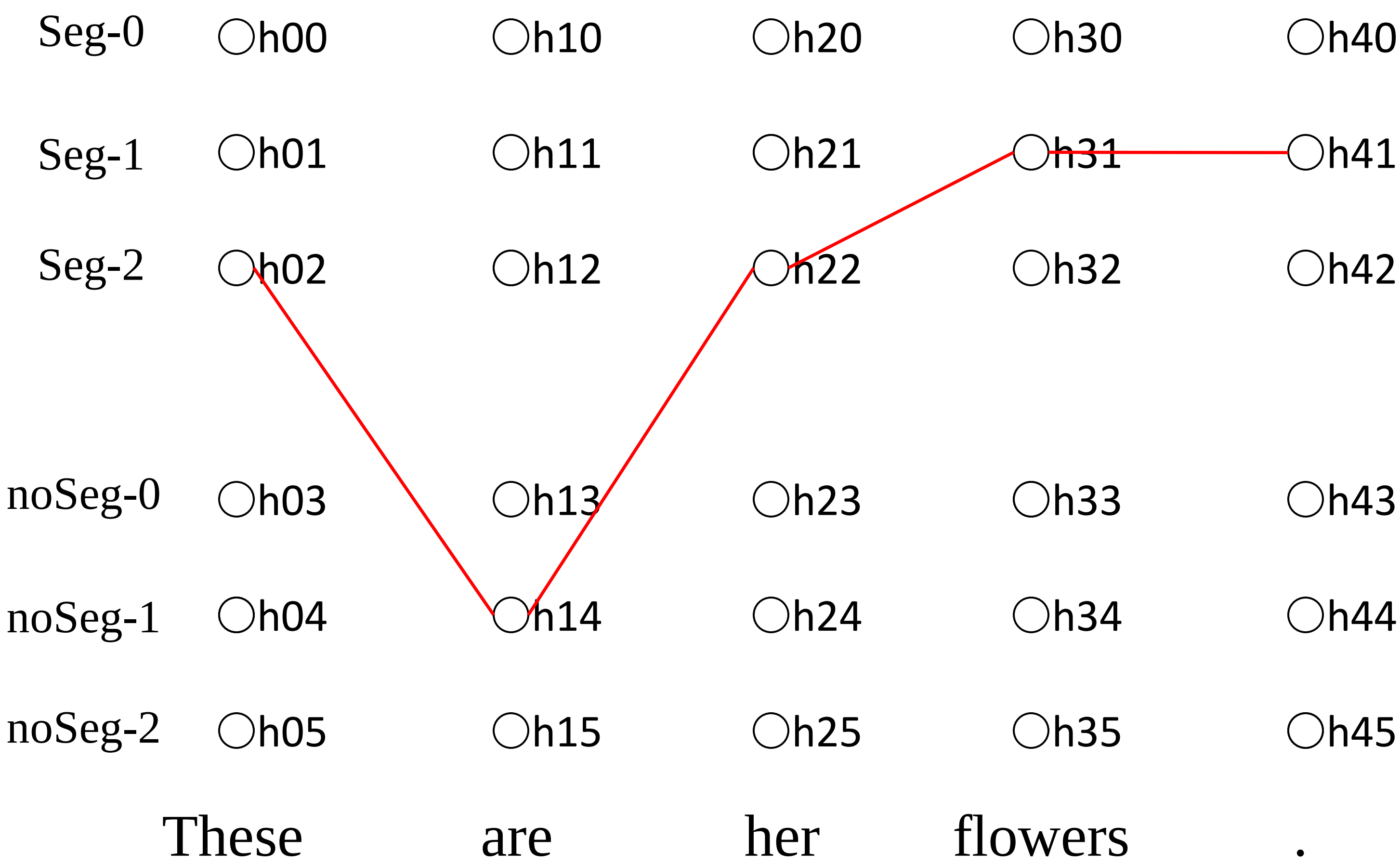
(1) Admissible heuristics for A* search



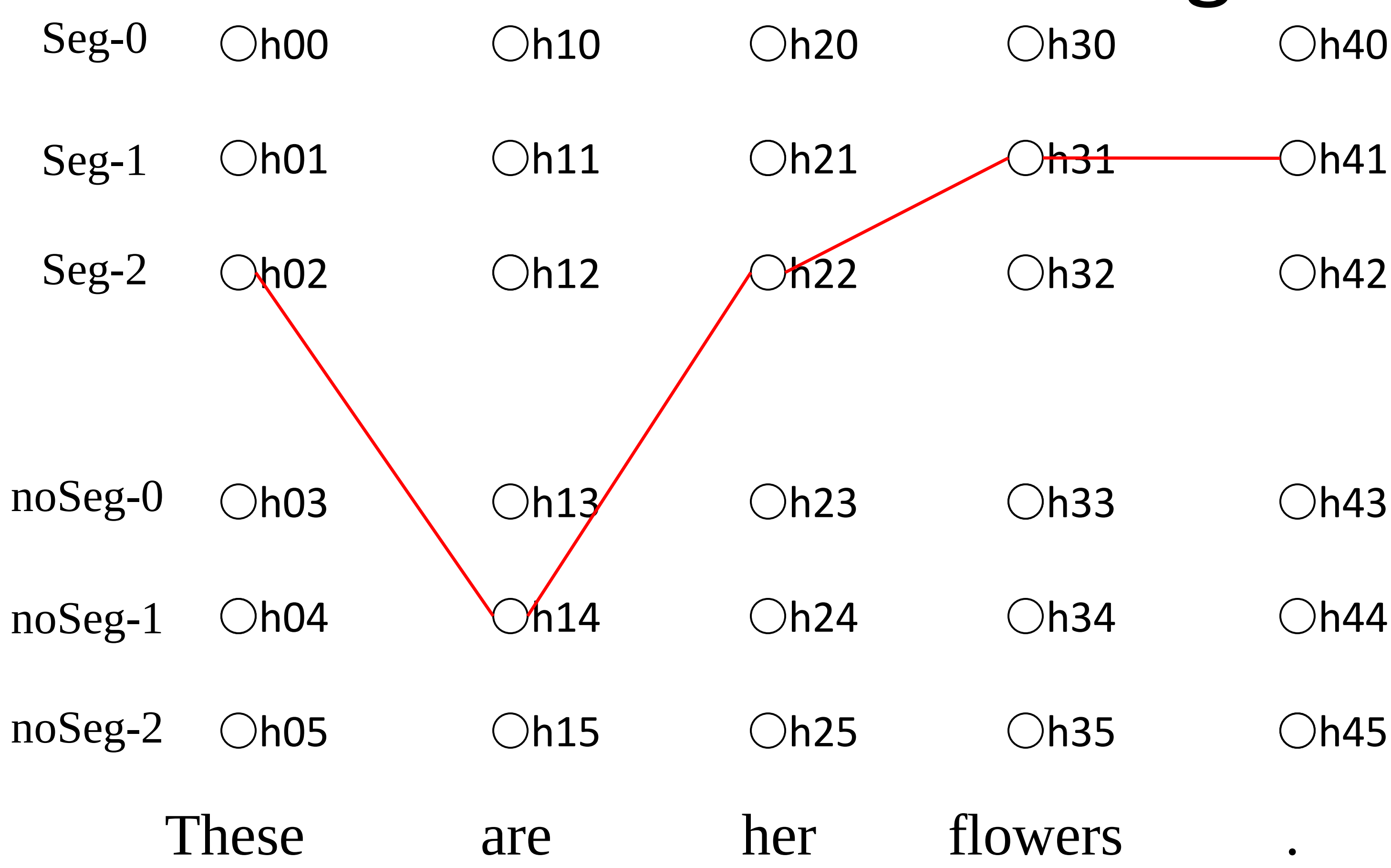
(1) Admissible heuristics for A* search

Seg-0	○h00	○h10	○h20	○h30	○h40
Seg-1	○h01	○h11	○h21	○h31	○h41
Seg-2	○h02	○h12	○h22	○h32	○h42
noSeg-0	○h03	○h13	○h23	○h33	○h43
noSeg-1	○h04	○h14	○h24	○h34	○h44
noSeg-2	○h05	○h15	○h25	○h35	○h45
	These	are	her	flowers	.

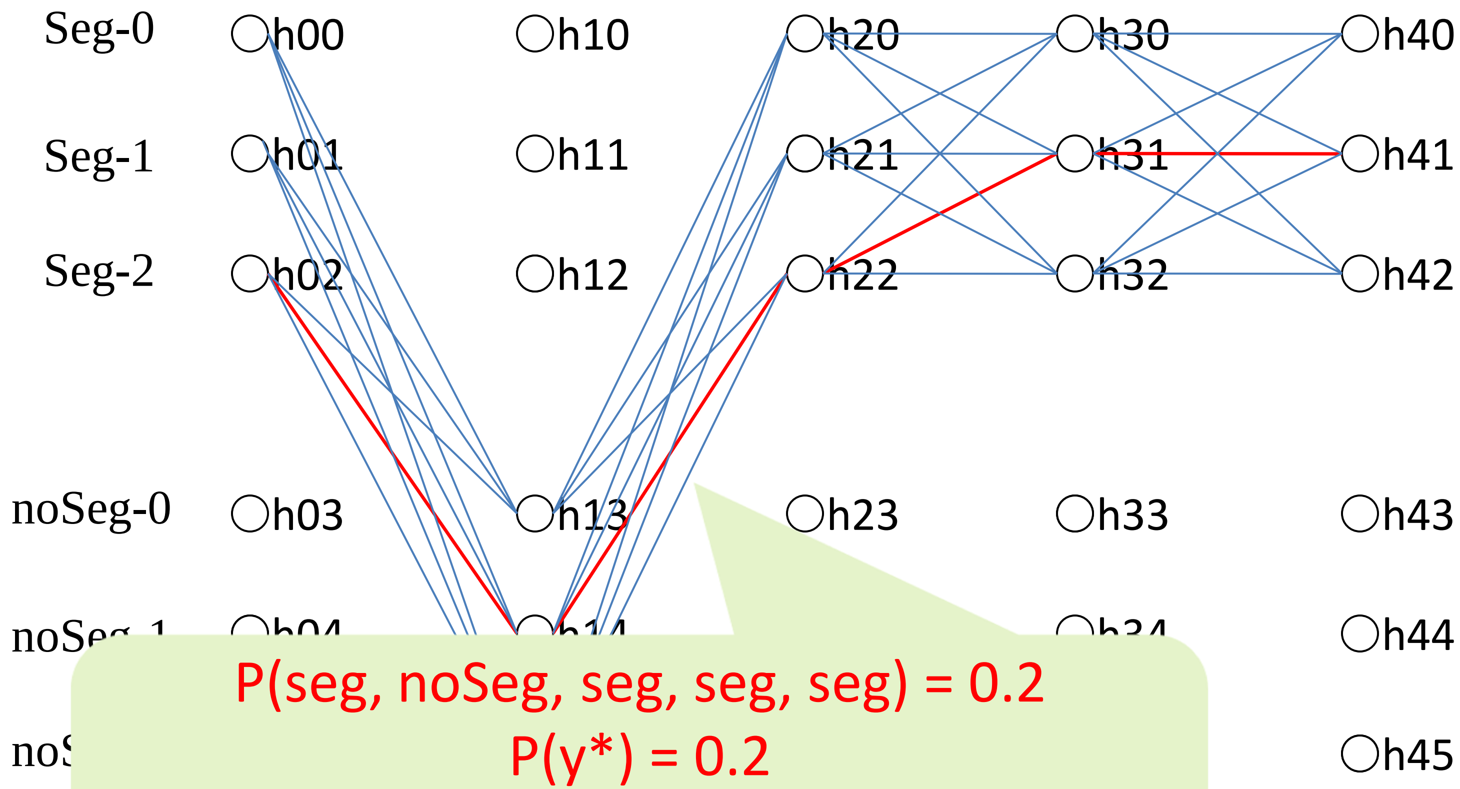
(2) Find 1st latent path h1: A* search



(3) Get y1 & P(y1): Forward-Backward algo.

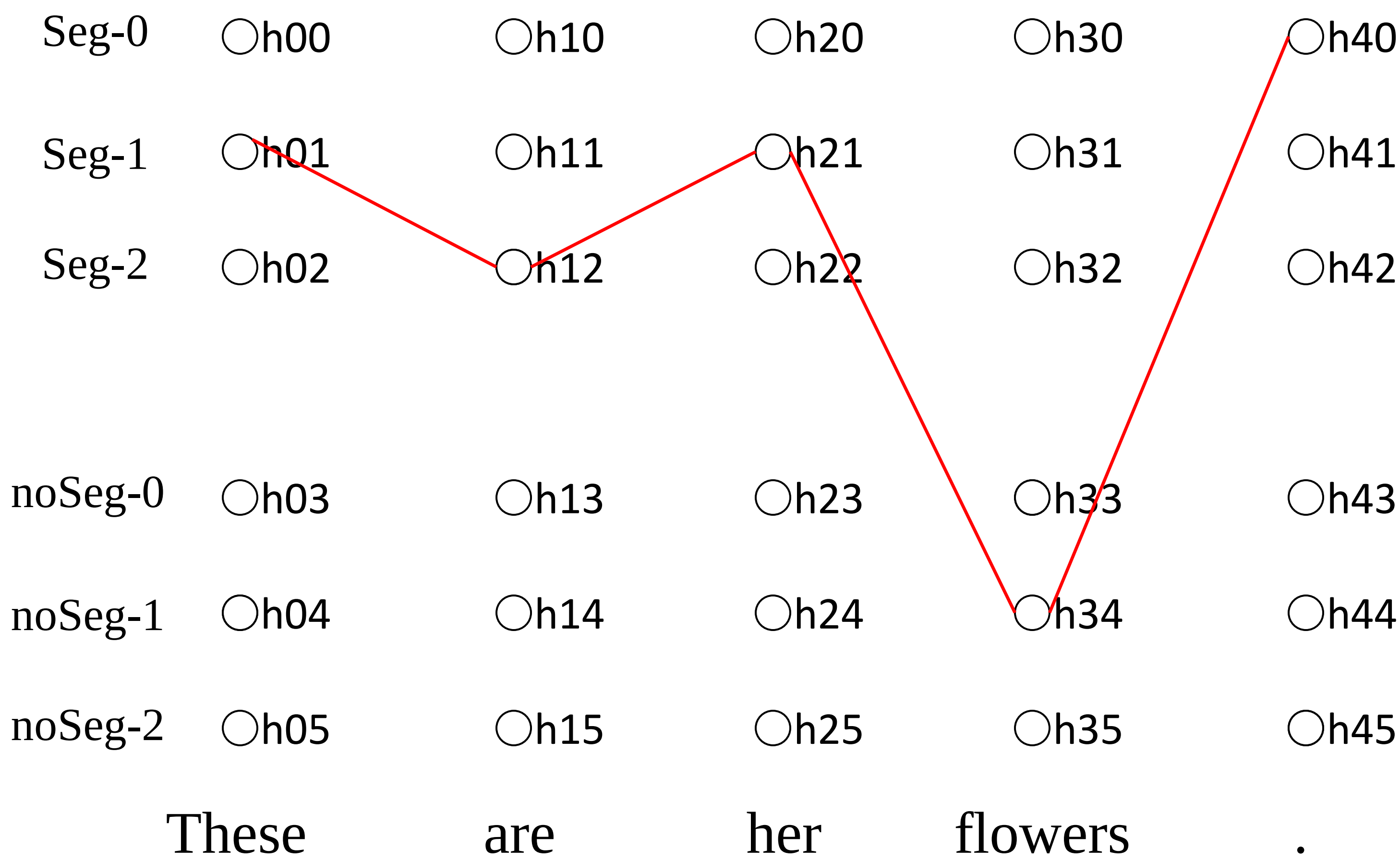


(3) Get y1 & P(y1): Forward-Backward algo.

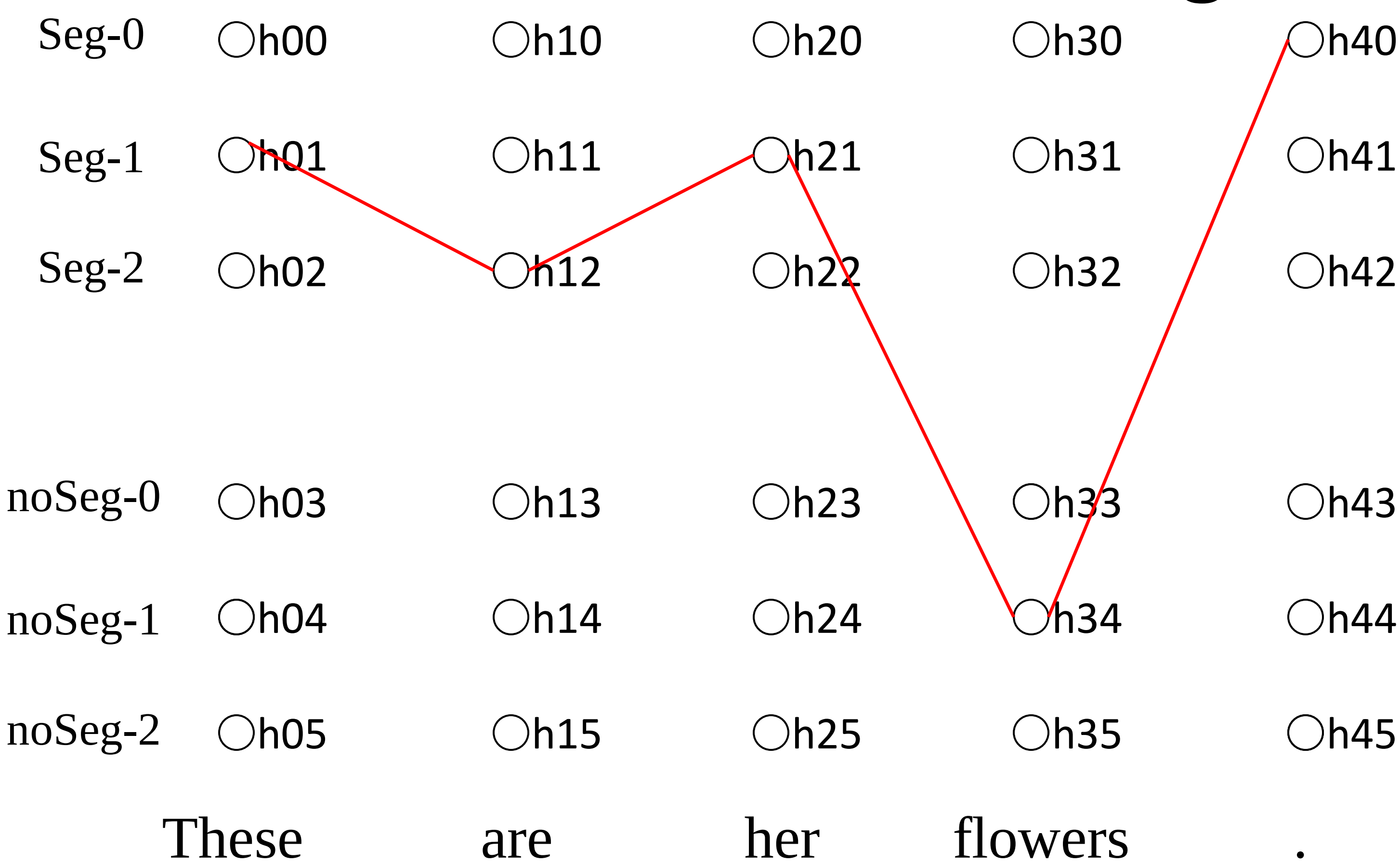


P(seg, noSeg, seg, seg, seg) = 0.2
P(y*) = 0.2
P(unknown) = 1 - 0.2 = 0.8
P(y*) > P(unknown) ?

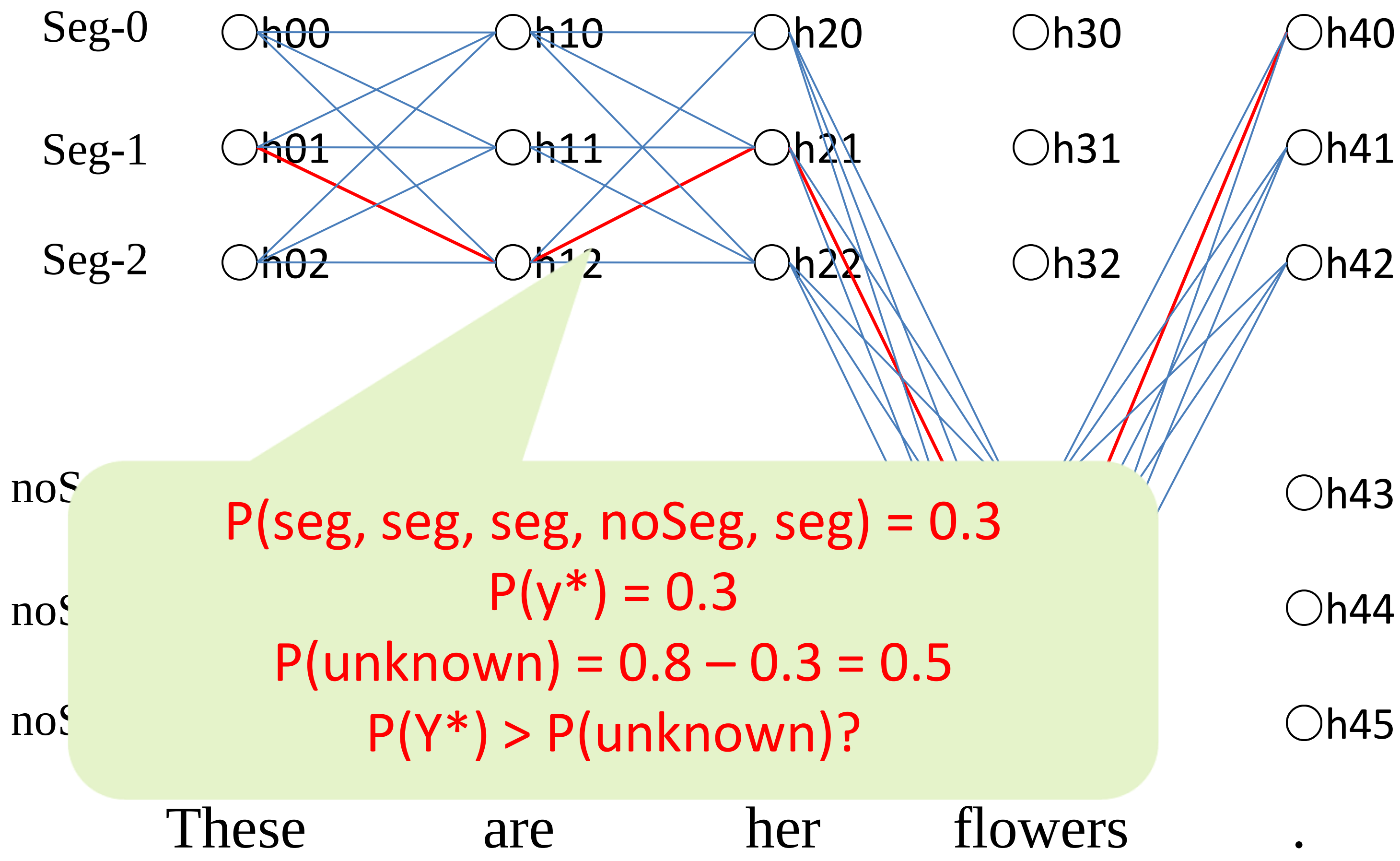
(4) Find 2nd latent path h2: A* search



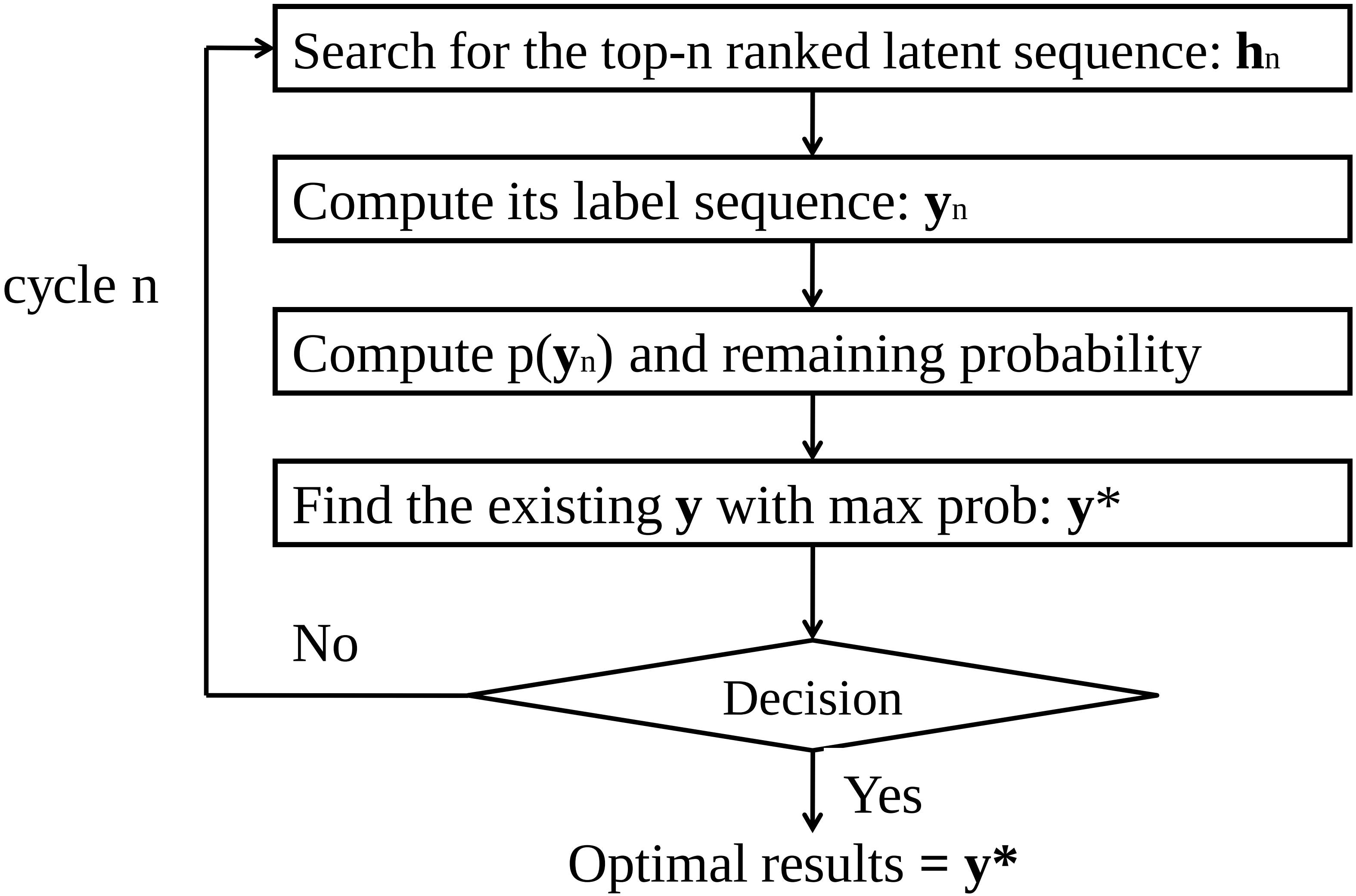
(5) Get y2 & P(y2): Forward-backward algo.



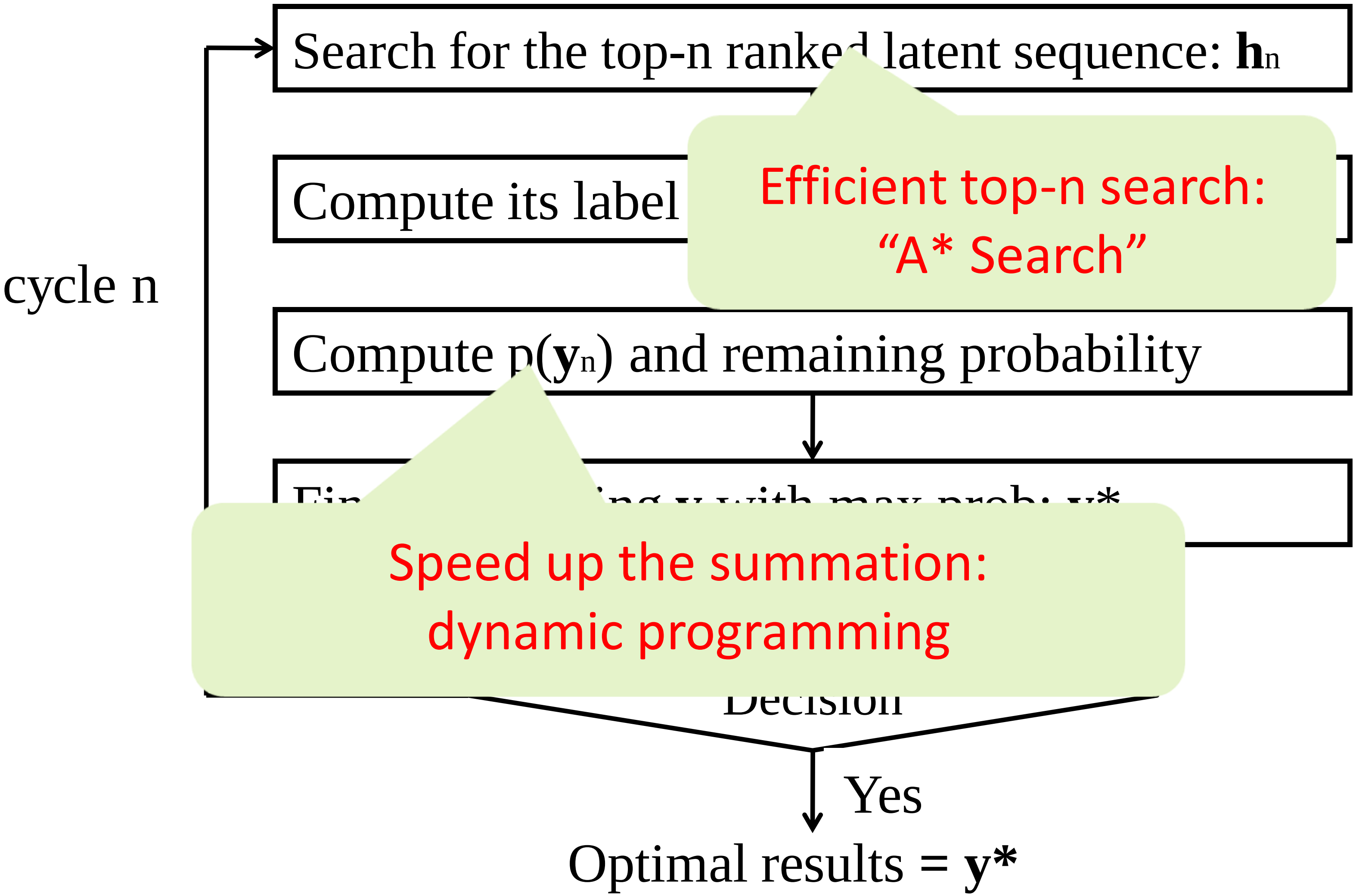
(5) Get y2 & P(y2): Forward-backward algo.



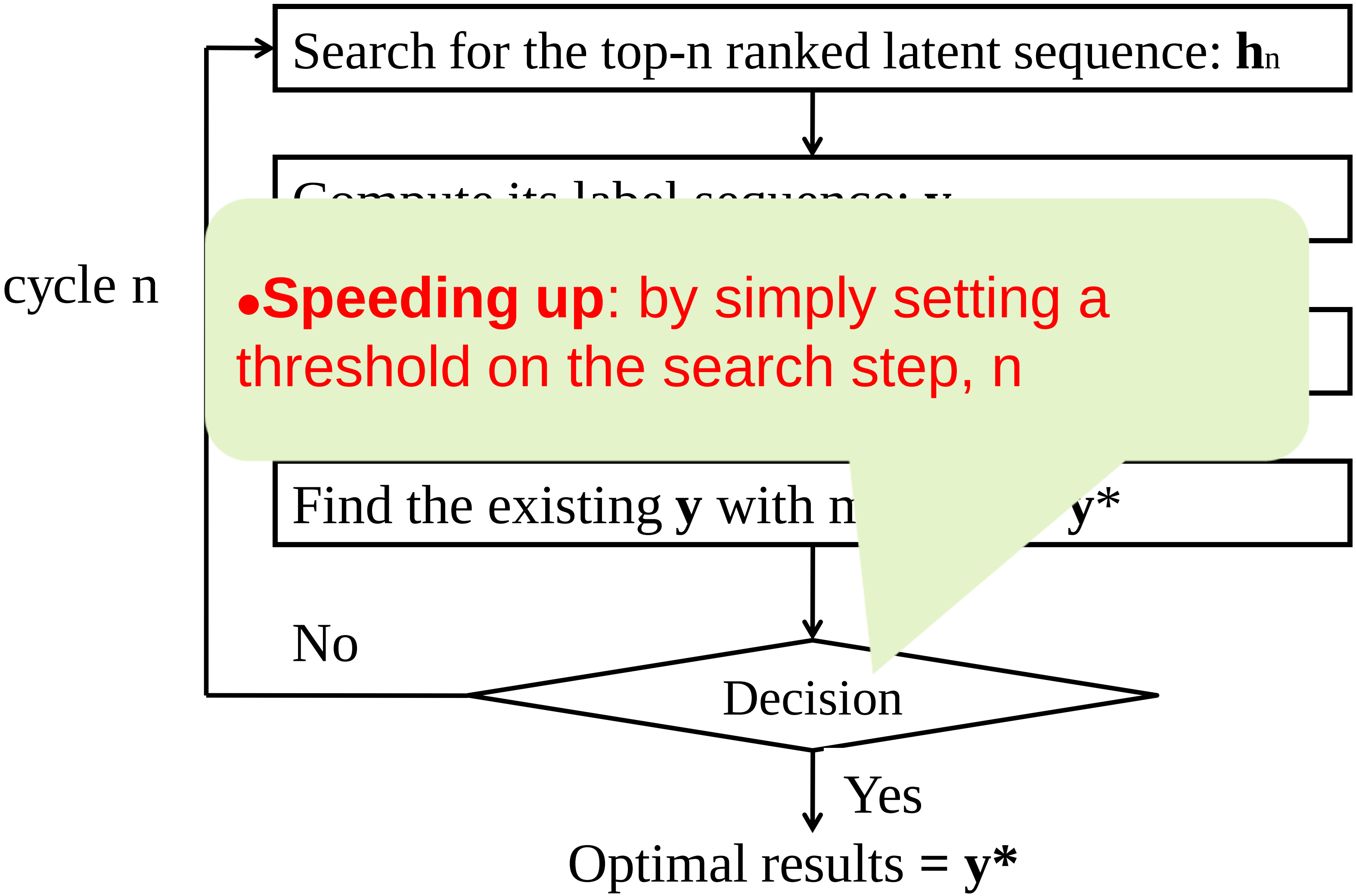
Data flow: the inference algo.



Key: make this exact method as fast as previous approx. methods!



Key: make this exact method as fast as previous approx. methods!



- **Speeding up**: by simply setting a threshold on the search step, n

Conclusions

- Inference on LDCRFs is an NP-hard problem (even for linear-chain latent dynamics)!
- Proposed an exact inference method on LDCRFs.
- The proposed method achieves good accuracies yet with fast speed.

Latent variable perceptron for structured classification

Xu Sun (孫 栩)
University of Tokyo

2010.06.16

A new model for fast training

[Sun+ IJCAI 09]

Conditional latent variable model:

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}} \cancel{\sum_{\boldsymbol{h}:\mathrm{Proj}(\boldsymbol{h})=\boldsymbol{y}}} P(\boldsymbol{h} \mid \boldsymbol{x}, \boldsymbol{\theta})$$

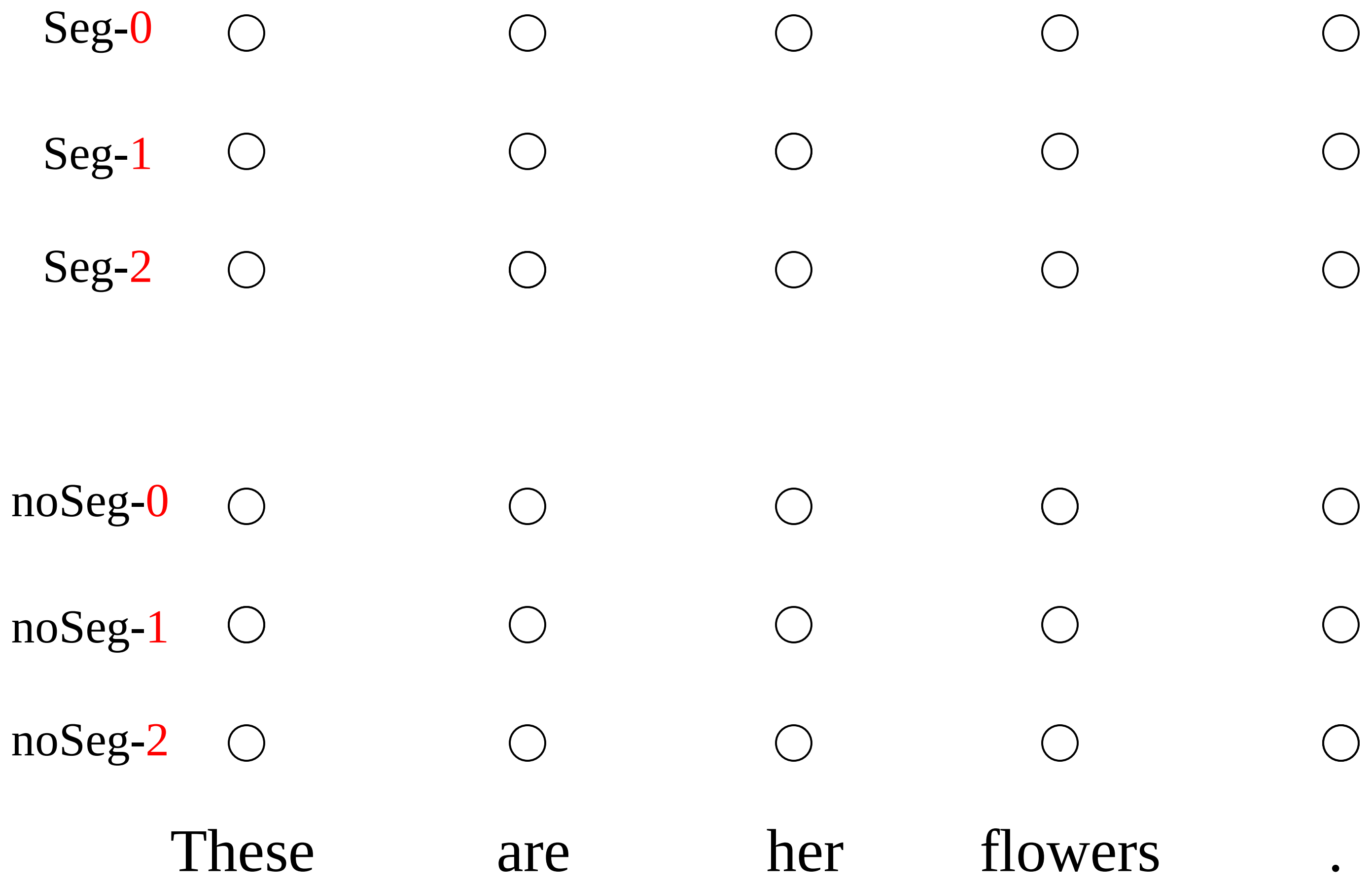
Normally, batch training
(do weight update after go over all samples)

Our proposal, a new model (*Sun et al., 2009*) :

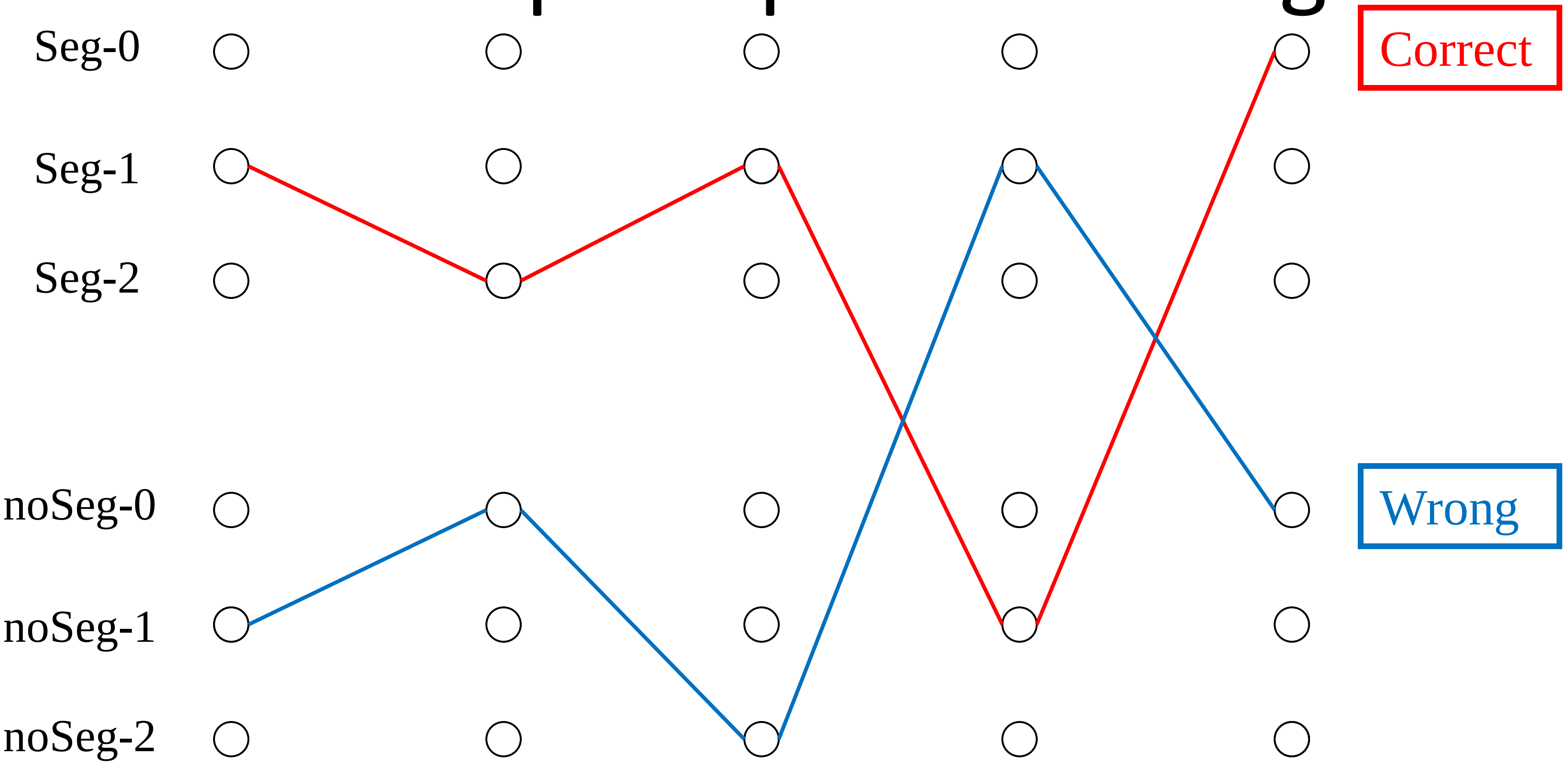
$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}} P'(\boldsymbol{h} \mid \boldsymbol{x}, \boldsymbol{\theta})$$

Online training
(do weight update on each sample)

Our proposal:
latent perceptron training



Our proposal:
latent perceptron training



$$\boldsymbol{\theta}^{i+1} = \boldsymbol{\theta}^{i} + \boxed{\mathbf{f}[\arg\max_{\mathbf{h}} F(\mathbf{h} \mid \mathbf{y}_i^{*}, \mathbf{x}_i, \boldsymbol{\theta}^{i}), \mathbf{x}_i]} - \boxed{\mathbf{f}[\arg\max_{\mathbf{h}} F(\mathbf{h} \mid \mathbf{x}_i, \boldsymbol{\theta}^{i}), \mathbf{x}_i]}$$

Convergence analysis: separability
[Sun+ IJCAI 09]

- With latent variables, is data space still separable? Yes

Theorem 1. *Given the latent feature mapping* $\mathbf{m} = (m_1, \ldots, m_n)$, *for any sequence of training examples* $(\mathbf{x}_i, \mathbf{y}_i^*)$ *which is separable with margin* δ *by a vector* $\mathbf{U}$ *represented by* $(\alpha_1, \ldots, \alpha_n)$ *with* $\sum_{i=1}^{n} {\alpha_i}^2 = 1$, *the examples then will also be latently separable with margin* $\overline{\delta}$, *and* $\overline{\delta}$ *is bounded below by*

$$\overline{\delta} \geq \delta / T,$$

where $T = (\sum_{i=1}^{n} m_i {\alpha_i}^2)^{1/2}$.

Convergence

[Sun+ IJCAI 09]

- Is latent perceptron training convergent?

Yes

Theorem 2. *For any sequence of training examples* $(\mathbf{x}_i, \mathbf{y}_i^*)$ *which is separable with margin* δ, *the number of mistakes of the latent perceptron algorithm in Figure 1 is bounded above by*

$$number\ of\ mistakes \leq 2T^2M^2/\delta^2$$

Comparison to traditional perceptron:

$$number\ of\ mistakes \leq R^2 / \delta^2$$

A difficult case: inseparable data
[Sun+ IJCAI 09]

- Are errors tractable for inseparable data?

#mistakes per iteration is up-bounded

Theorem 3. *For any training sequence* $(\mathbf{x}_i, \mathbf{y}_i^*)$, *the number of mistakes made by the latent perceptron training algorithm is bounded above by*

$$number\ of\ mistakes \leq \min_{\overline{\mathbf{U}}, \overline{\delta}} (\sqrt{2}M + D_{\overline{\mathbf{U}}, \overline{\delta}})^2 / \overline{\delta}^2$$

Summarization: convergence analysis

- Latent perceptron is convergent
  - By adding any latent variables, a separable data will still be separable
  - Training is not endless (will stop on a point)
  - Converge speed is fast (similar to traditional perceptron)
  - Even for a difficult case (inseparable data), mistakes are tractable (up-bounded on #mistake-per-iter)

References & source code

- X. Sun, T. Matsuzaki, D. Okanohara, J. Tsujii. Latent variable perceptron for structured classification. In *IJCAI 2009*.
- X. Sun & J. Tsujii. Sequential labeling with latent variables. In *EACL 2009*.

- Souce code (Latent-dynamic CRF, LDI inference, Latent-perceptron) can be downloaded from my homepage:

  http://www.ibis.t.u-tokyo.ac.jp/XuSun