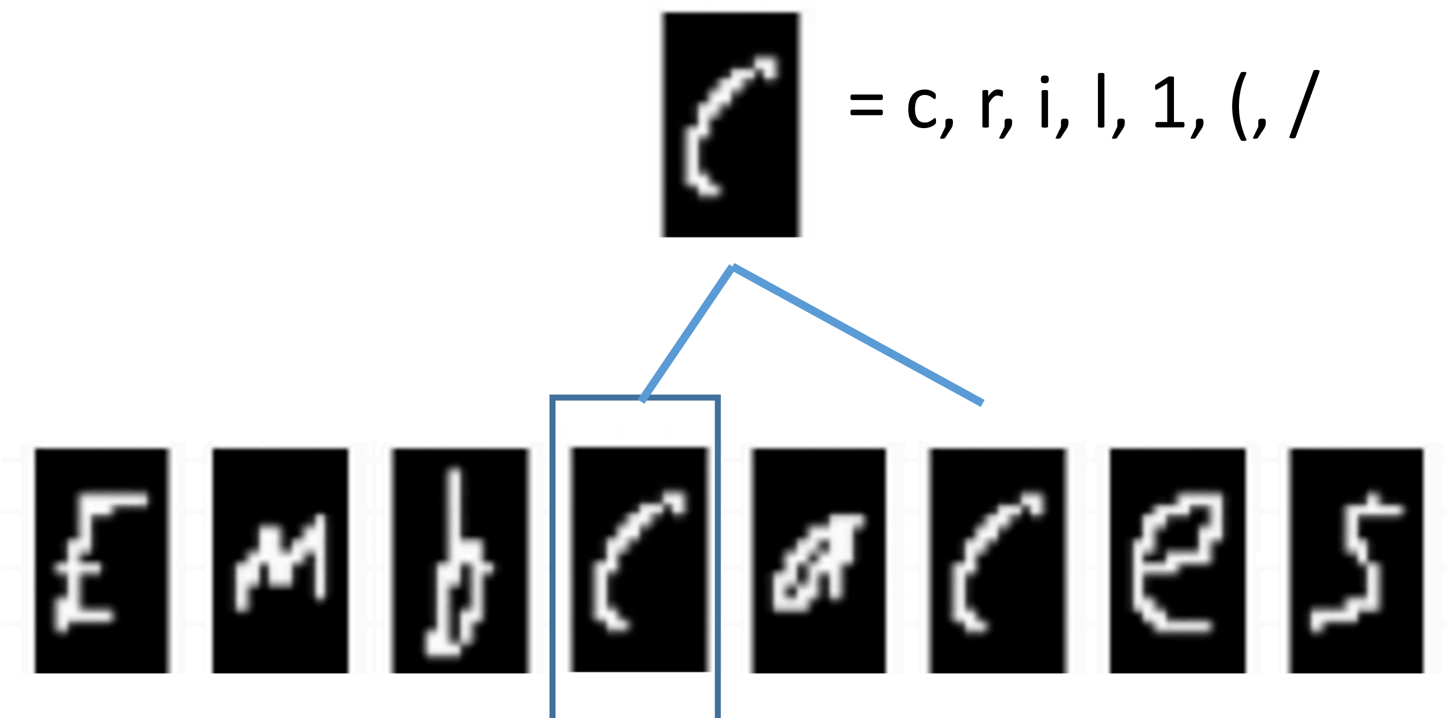
Structured Predictions:
Practical Advancements and Applications in Natural Language Processing

Kai-Wei Chang
UCLA

References: http://kwchang.net/talk/sp.html

Why is structure important? Hand written recognition example

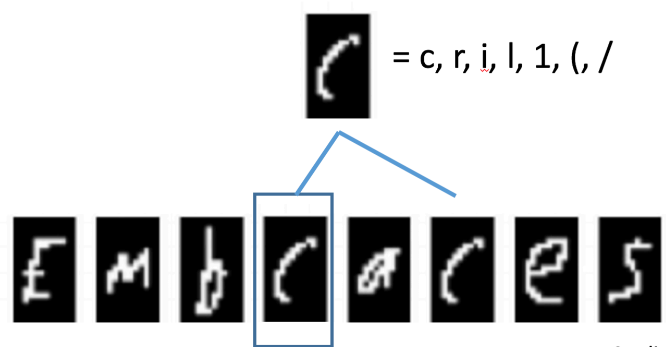
- What is this English letter?



Credit: Ben Taskar

Why is structure important? Hand written recognition example

❖ What is this English letter?



Credit: Ben Taskar

Part-of-speech (POS) tagging:

A	♍♋♦	chases	a	mouse
Det	??	Verb	Det	Noun
	Noun			

Q: [Chris] = [Mr. Robin] ?

Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh.** As **a boy**, **Chris** lived in a pretty home called **Cotchfield Farm**. When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

P.S. In fact, Alan Alexander Milne is the author.

Slide modified from Dan Roth

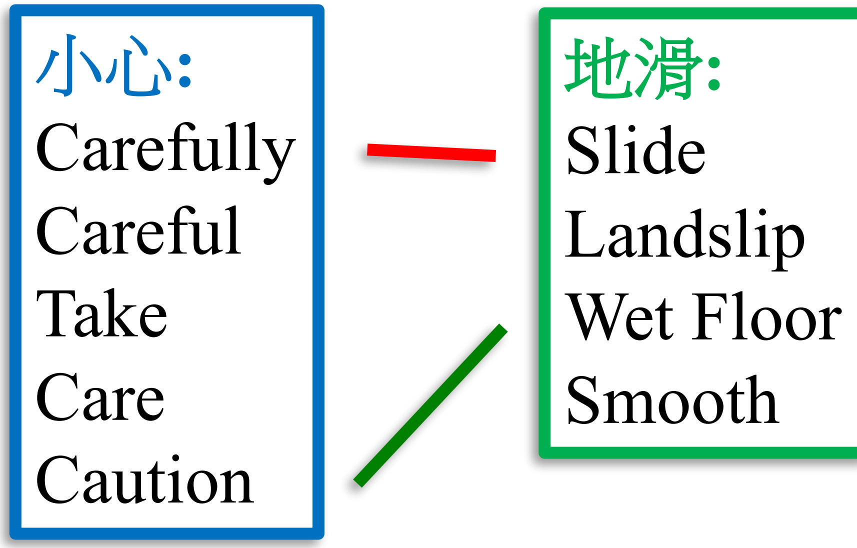
Complex Decision Structure

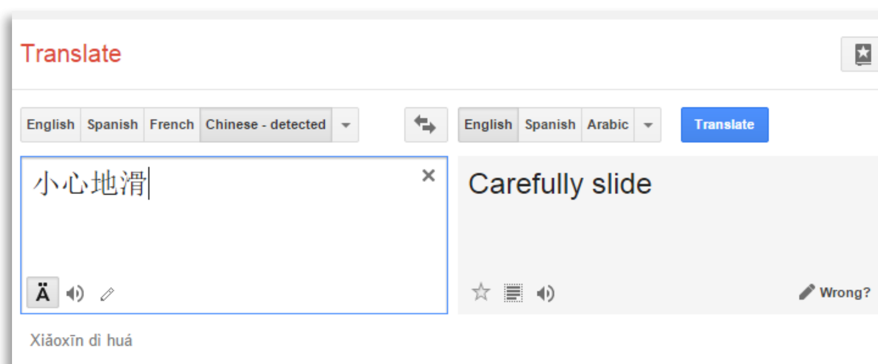
Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh.** As **a boy**, **Chris** lived in a pretty home called **Cotchfield Farm**. When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

P.S. In fact, Alan Alexander Milne is the author.

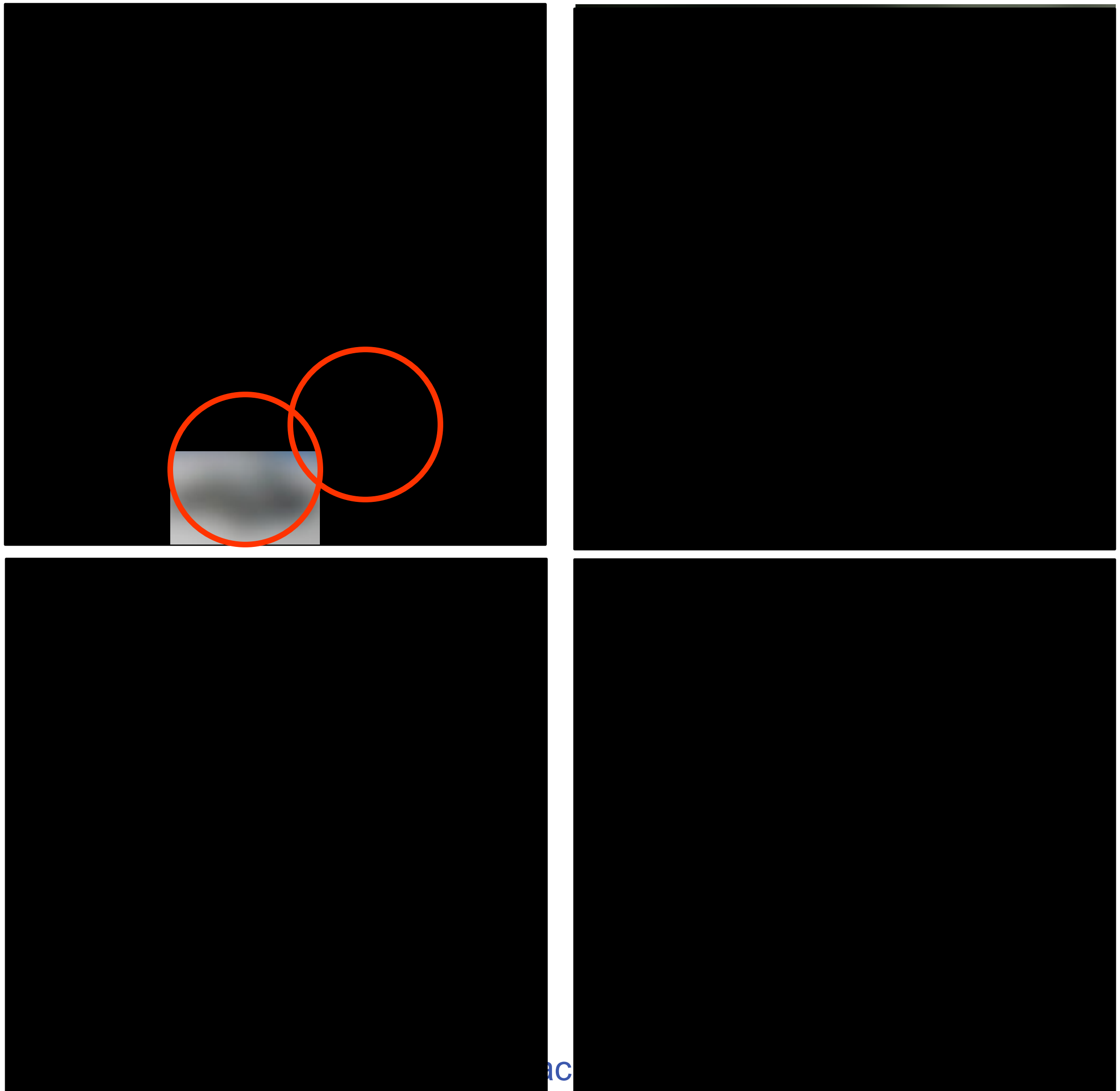
Challenges--language is compositional

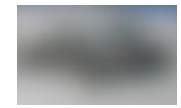
Challenges--language is compositional





Visual recognition





Credit: Dhruv Batra

Human body recognition

Joint Inference with General Constraint Structure
[Roth&Yih'04,07,….]

Recognizing Entities and Relations

Joint inference gives good improvement

other	0.05
per	**0.85**
loc	0.10

other	0.10
per	**0.60**
loc	0.30

other	0.05
per	0.50
loc	**0.45**

Bernie's wife, Jane, is a native

E_1 E_2

R_{12} R_{23}

Key Questions:

How to learn the model(s)?

What is the source of the knowledge?

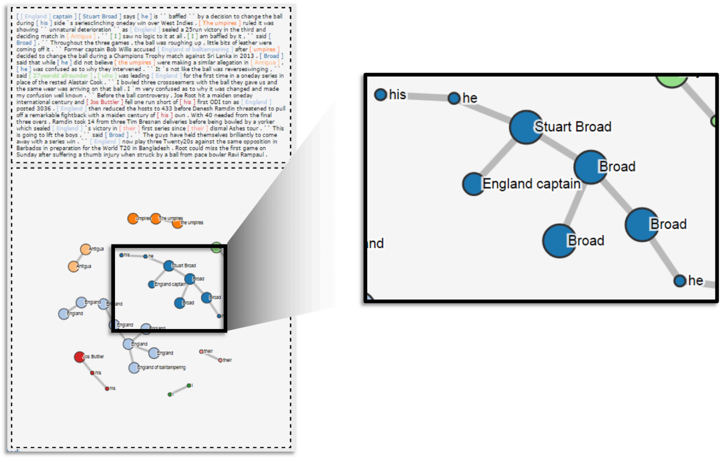
How to guide the global inference?

irrelevant	0.05
spouse_of	**0.45**
born_in	0.50

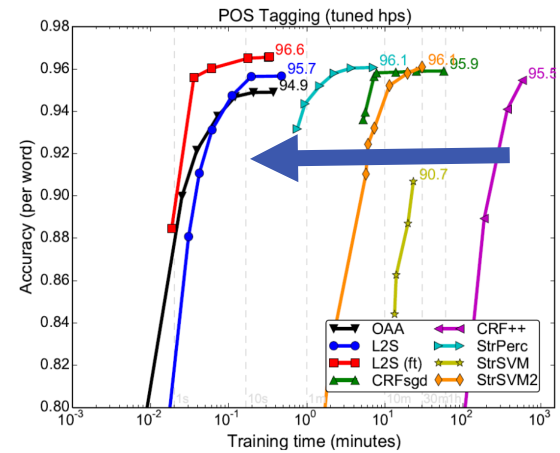
irrelevant	0.10
spouse_of	0.05
born_in	**0.85**

Models could be learned separately/jointly; constraints may come up only at decision time.

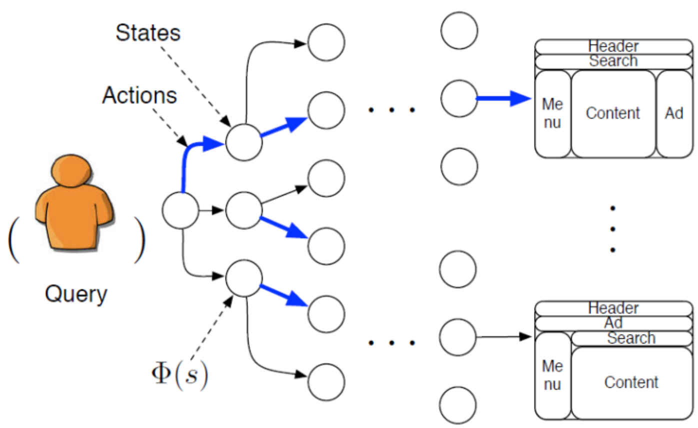
Structured Prediction Models



How to model?



Training/test/dev speed



Learning signals

activity	cooking
agent	woman
food	vegetable

Fairness (data biases)

Outline

- Part 1: 14:40-15:50

  From binary to structured prediction

  -- Introduction to CRF

  -- Constrained conditional model

- Part 2: 16:00-17:30

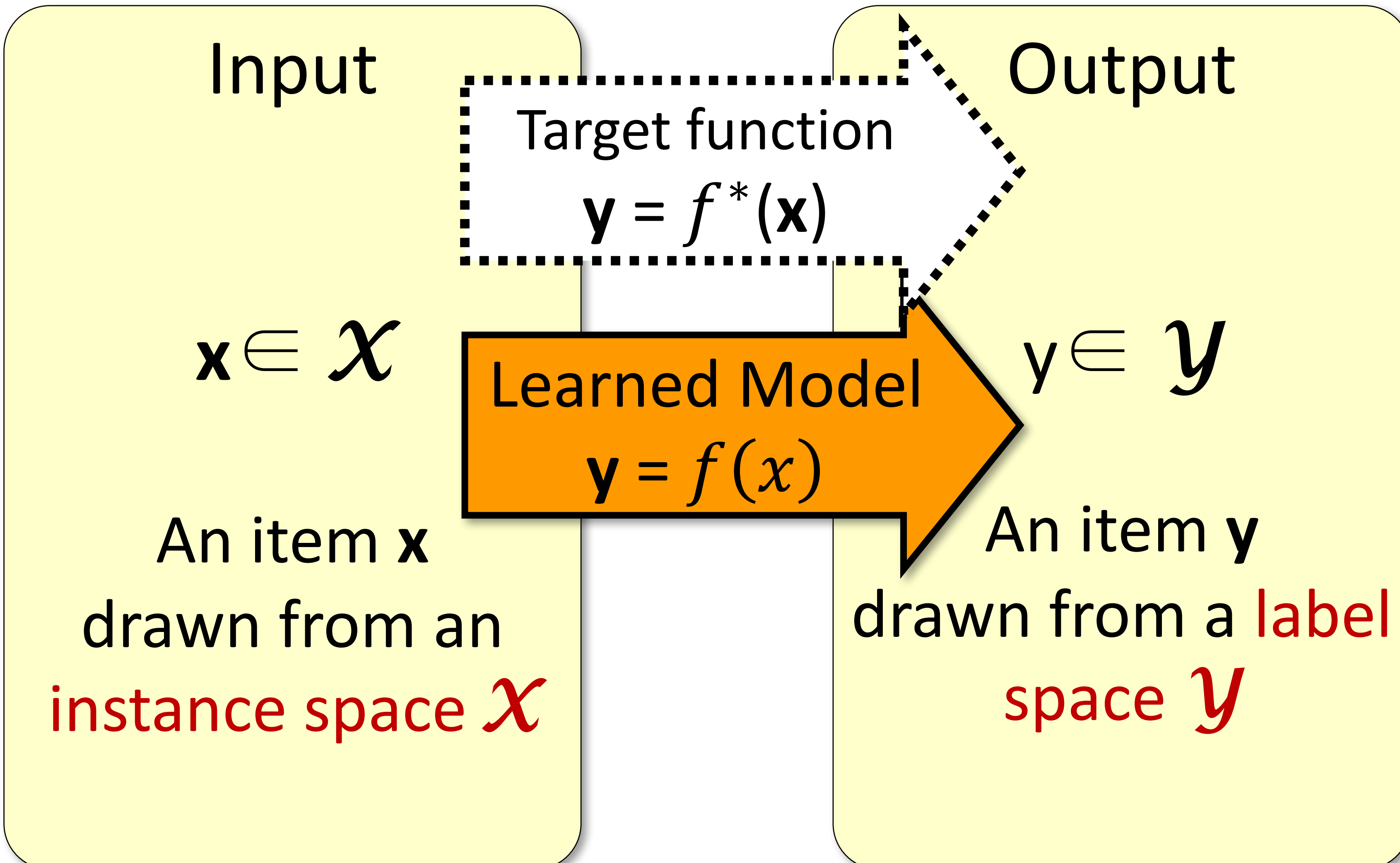
  Advanced topics

  -- Efficient inference/learning

  -- Learning from Indirect supervision signal

  -- Reducing human biases in structured models

Supervised learning



Supervised learning

y is represented in output space (label space)
Different kinds of output:

- Binary classification:
  $y \in \{-1,1\}$
- Multiclass classification:
  $y \in \{1,2,3, \ldots K\}$
- Regression:
  $y \in R$
- Structured output
  $y \in \{1,2,3, \ldots K\}^N$

Output

y $\in \mathcal{Y}$

An item **y** drawn from a label space $\mathcal{Y}$

Combinatorial optimization problem

$$\hat{y} = \text{argmax}_{\boldsymbol{y}\in\boxed{\mathcal{Y}}} \, f(\boldsymbol{y}; \boxed{\boldsymbol{w}}, \boxed{\boldsymbol{x}})$$

input

model parameters

output space

- Inference/Test: given $\boldsymbol{w}, \boldsymbol{x}$, solve argmax
- Learning/Training: find a good $\boldsymbol{w}$

Binary Linear Classifiers

$$\hat{y} = \operatorname{argmax}_{\boldsymbol{y} \in \mathcal{Y}} f(\boldsymbol{y}; \boldsymbol{w}, \boldsymbol{x})$$

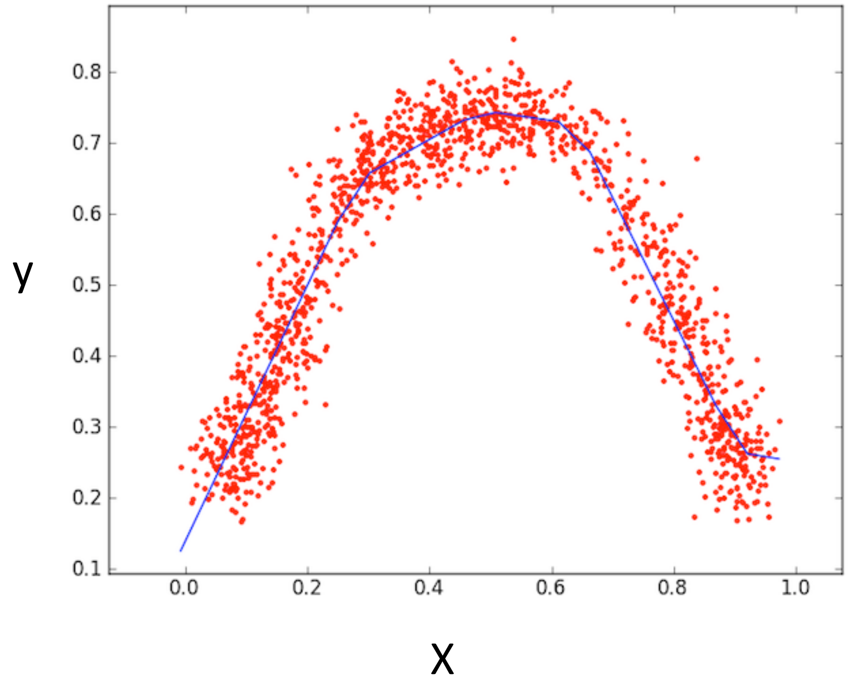
- $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}, \mathcal{Y} = \{-1,1\}$
- $f(y; \boldsymbol{w}, \boldsymbol{x}) \stackrel{\text{def}}{=} y(\boldsymbol{w}^\top \boldsymbol{x} + \boldsymbol{b}) = y(\sum_i w_i x_i + \boldsymbol{b})$
- $\operatorname{argmax}_{\boldsymbol{y} \in \mathcal{Y}} f(\boldsymbol{y}; \boldsymbol{w}, \boldsymbol{x}) = \begin{cases} 1, \boldsymbol{w}^\top \boldsymbol{x} + b \geq 0 \\ -1, \boldsymbol{w}^\top \boldsymbol{x} + b < 0 \end{cases}$

$$= \operatorname{sgn}(\boldsymbol{w}^\top \boldsymbol{x} + \boldsymbol{b})$$

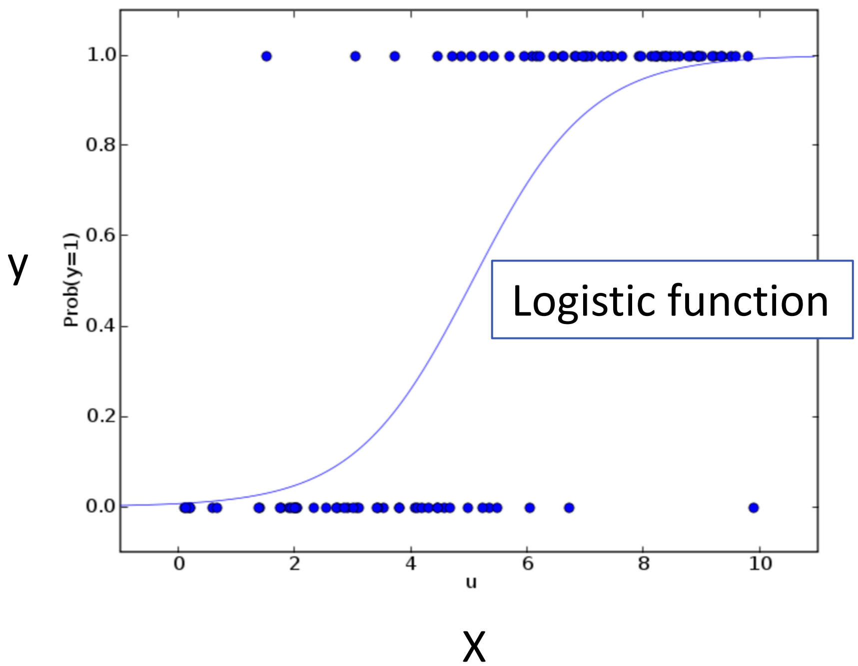
(break ties arbitrarily)

Recap: Logistic Regression

Regression



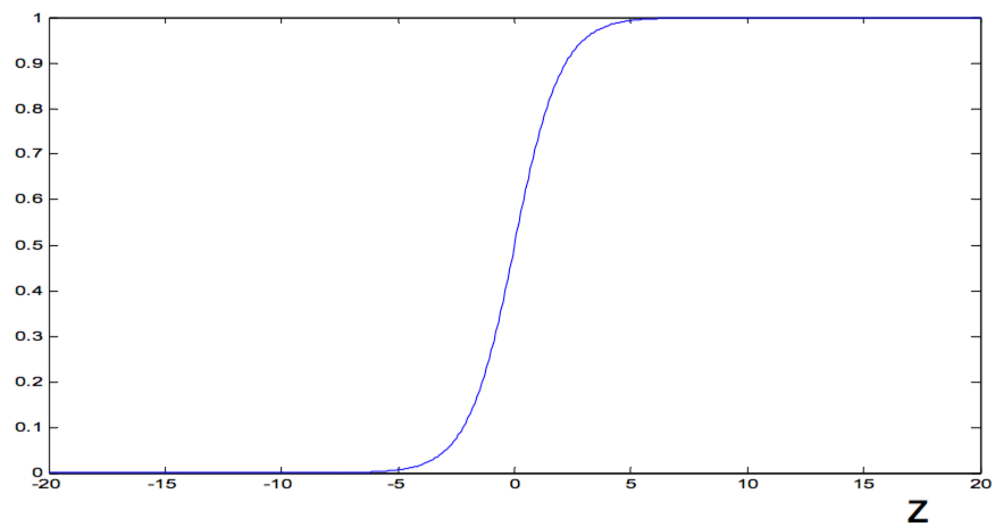
Logistic regression



logistic function or sigmoid function

- When $z \to \infty$ what is $\sigma(z)$?
- When $z \to -\infty$ what is $\sigma(z)$?
- When $z = 0$ what is $\sigma(z)$?

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$



Probabilistic Interpretation

Assume labels are generated using the following probability distribution:

$$P(y = 1|\mathbf{x}, \mathbf{w}) = \frac{e^{\mathbf{w}^T\mathbf{x}}}{1 + e^{\mathbf{w}^T\mathbf{x}}} = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}$$

$$P(y = -1|\mathbf{x}, \mathbf{w}) = \frac{1}{1 + e^{\mathbf{w}^T\mathbf{x}}}$$

$$P(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\mathbf{w}^T\mathbf{x}}}{1+e^{\mathbf{w}^T\mathbf{x}}} = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}}$$

$$P(y=-1|\mathbf{x},\mathbf{w}) = \frac{1}{1+e^{\mathbf{w}^T\mathbf{x}}}$$

$$\Downarrow$$

$$P(y|\mathbf{x},\mathbf{w}) = \frac{1}{1+\exp\left(-y\mathbf{w}^T\mathbf{x}\right)}$$

$$P(y \mid x, w) = \sigma(yw^T x)$$

$$\boxed{\sigma(z) = \frac{1}{1+e^{-z}}}$$

How to make prediction?

Predict y=1 if P(y=1|x,w) > P(y= -1|x, w)

$$P(y=1|\mathbf{x},\mathbf{w}) = \frac{e^{\mathbf{w}^T\mathbf{x}}}{1+e^{\mathbf{w}^T\mathbf{x}}} = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}}$$

$$P(y=-1|\mathbf{x},\mathbf{w}) = \frac{1}{1+e^{\mathbf{w}^T\mathbf{x}}}$$

$$\log\frac{P(y=1\mid x,w)}{P(y=-1\mid x,w)} = w^T x$$

- The decision boundary?

$$w^T x > 0$$

How to learn models
-- Maximum likelihood estimation

- Which bag of words more likely generate:

  aaaDaaaKoaaaa

Maximum likelihood estimation

- Probabilistic model assumption:

$$P(y|\mathbf{x}, \mathbf{w}) = \frac{1}{1 + \exp\left(-y\mathbf{w}^T\mathbf{x}\right)}$$

- The log-likelihood of seeing a dataset D = {(x , y)} if the true weight vector was w:

$$\log P(D|\mathbf{w}) = -\sum \log\left(1 + \exp(-y\mathbf{w}^T\mathbf{x}\right)$$

$$\boxed{\begin{array}{c} P(D|w) = \Pi_i\, P(y_i|x_i, w) \\ \Rightarrow \log P(D|w) = \sum_i \log P(y_i|x_i, w) \end{array}}$$

Minimizing negative log-likelihood

- Log likelihood

$$\log P(D|\mathbf{w}) = -\sum \log\left(1 + \exp(-y\mathbf{w}^T\mathbf{x}\right)$$

- Logistic regression

$$\min_{\boldsymbol{w}} \quad \sum_i \log(\, 1 + e^{-\mathrm{y_i}(\mathbf{w}^{\mathrm{T}}\mathbf{x}_{\mathrm{i}})})$$

- Let's add some prior (Gaussian Prior)

$$\min_{\boldsymbol{w}} \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_i \log(\, 1 + e^{-\mathrm{y_i}(\mathbf{w}^{\mathrm{T}}\mathbf{x}_{\mathrm{i}})})$$

(multi-class) log-linear model

Partition function

- Assumption:

$$P(y|x,w) = \frac{\exp\left(w_y^T x\right)}{\sum_{y' \in \{1,2,\ldots K\}} \exp\left(w_{y'}^T x\right)}$$

- This is a valid probability assumption. Why?

This often called soft-max

Softmax

- Softmax: let s(y) be the score for output y
  here s(y)=$w_y^T x$, but it can be computed by other metric.

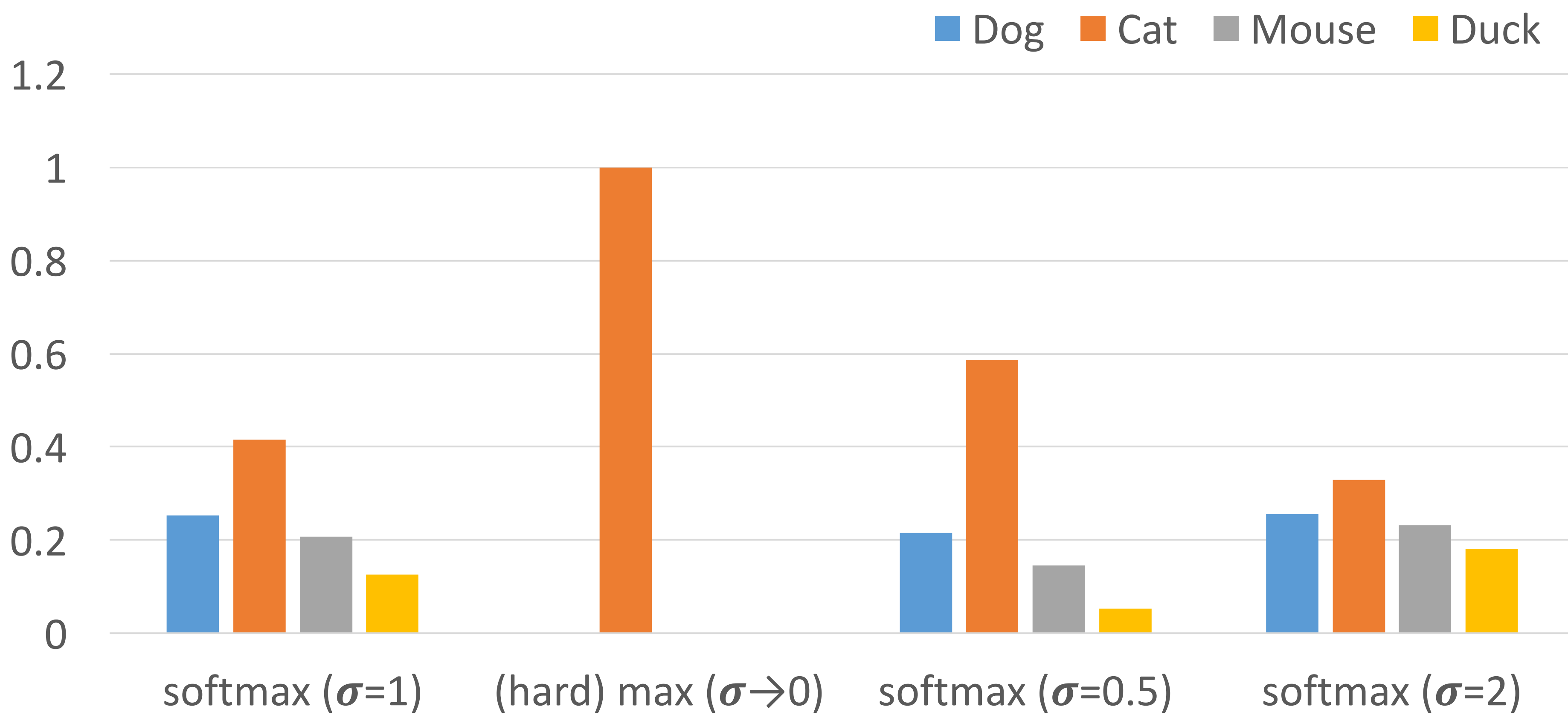
$$P(y) = \frac{\exp(s(y))}{\sum_{y\prime \in \{1,2,\ldots K\}} \exp(s(y))}$$

- We can control the peakedness of the distribution

$$P(y|\sigma) = \frac{\exp(s(y)/\sigma)}{\sum_{y\prime \in \{1,2,\ldots K\}} \exp(s(y)/\sigma)}$$

Temperature (example)

$$P(y|\sigma) = \frac{\exp(s(y)/\sigma)}{\sum_{y' \in \{1,2,\ldots K\}} \exp(s(y)/\sigma)}$$



Maximum log-likelihood estimation

- Training can be done by maximum log-likelihood estimation i.e. $\max_{w} \log P(D|w)$

D=$\{(x_i, y_i)\}$

$$P(D|w) = \Pi_i \frac{\exp\left(w_{y_i}^T x_i\right)}{\sum_{y' \in \{1,2,\ldots K\}} \exp(w_{y'}^T x_i)}$$

$$\log P(D|w) = \sum_i [w_{y_i}^T x_i - \log \sum_{y' \in \{1,2,\ldots K\}} \exp(w_{y'}^T x_i)]$$

Maximum a posteriori

$D=\{(x_i, y_i)\}$

$P(w|D) \propto P(w)P(D|w)$

$\mathrm{m}ax_w \; -\frac{1}{2}\sum_y w_y^T w_y + \; C\sum_i [w_{y_i}^T x_i - \log\sum_{y' \in \{1,2,\ldots K\}} \exp(w_{y'}^T x_i)]$

or

$\min\limits_w \; \frac{1}{2}\sum_y w_y^T w_y + \; C\sum_i [\log\sum_{y' \in \{1,2,\ldots K\}} \exp\left(w_{y'}^T x_i\right) - w_{y_i}^T x_i]$

(multi-class) log-linear model

- Assumption:

Partition function

$$P(y|x,w) = \frac{\exp\left(w_y^T x\right)}{\sum_{y' \in \{1,2,\ldots K\}} \exp\left(w_{y'}^T x\right)}$$

- Another way to write this (with Kesler construction) is

$$P(y|x,w) = \frac{\exp(w^T \phi(x,y))}{\sum_{y' \in \{1,2,\ldots K\}} \exp(w^T \phi(x,y'))}$$

Kesler construction

Assume we have a multi-class problem with K class and n features.

$$w_i^T x$$

models:

$w_1, w_2, \ldots w_K,$

$w_k \in R^n$

- Input:

  $x \in R^n$

$$w^T \phi(x, i)$$

$$w = \begin{bmatrix} w_1 \\ \vdots \\ w_y \\ \vdots \\ w_n \end{bmatrix}_{nK \times 1} \qquad \phi(x, y) = \begin{bmatrix} 0_n \\ \vdots \\ x \\ \vdots \\ 0_n \end{bmatrix}_{nK \times 1}$$

x in y^{th} block;
Zeros elsewhere

$$w^T \phi(x, y) = w_y^T x$$

log-linear model

Partition function

- Assumption:

$$P(y|x,w) = \frac{\exp(w^T\phi(x,y))}{\sum_{y' \in \{1,2,\ldots K\}} \exp(w^T\phi(x,y'))}$$

- Learning:

$$\min_w \ \frac{1}{2}w^T w + \ C\sum_i [\log \sum_{y' \in \{1,2,\ldots K\}} \exp(w^T\phi(x_i,y')) - w^T\phi(x_i,y_i)]$$

How can we predict?

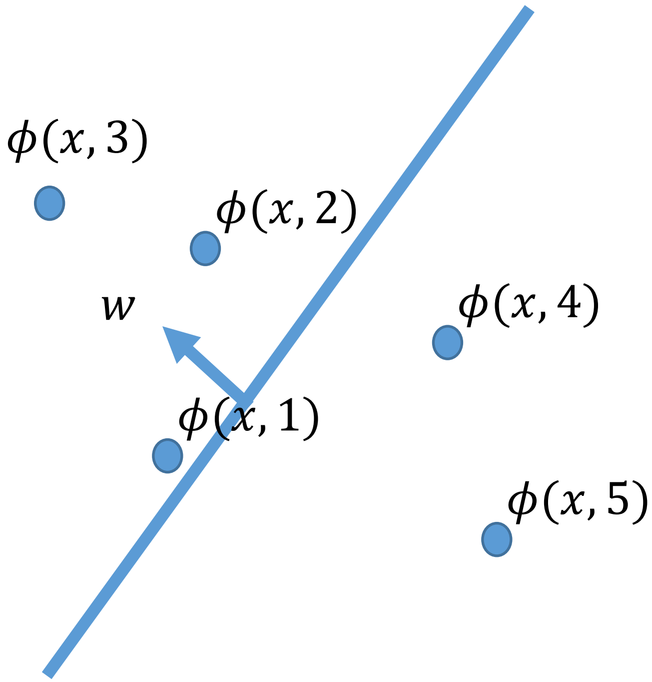
$$\mathrm{argmax}_y \;\; P(y \mid x, w)$$

$$\mathrm{argmax}_y \;\; w^T \phi(x, y)$$

$$w = \begin{bmatrix} w_1 \\ \vdots \\ w_y \\ \vdots \\ w_n \end{bmatrix}_{nK\times 1} \qquad \phi(x,y) = \begin{bmatrix} 0_n \\ \vdots \\ x \\ \vdots \\ 0_n \end{bmatrix}_{nK\times 1}$$

$$P(y|x,w) = \frac{\exp(w^T\phi(x,y))}{\sum_{y'\in\{1,2,\ldots K\}} \exp(w^T\phi(x,y'))}$$

For input an input x,
the model predict label is 3



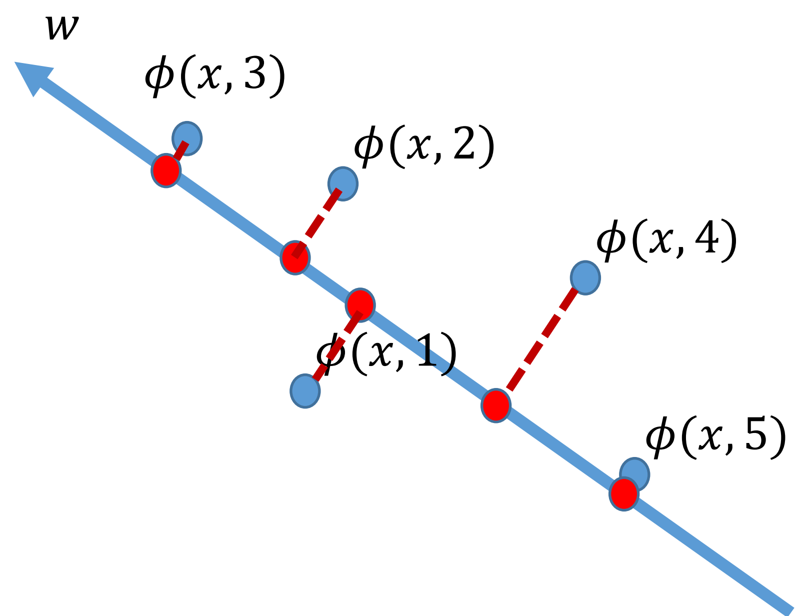
This is equivalent to
$\mathrm{argmax}_{y\in\{1,2,\ldots K\}} \; w_y^T \; x$

How can we predict?

For input an input x,
the model predict label is 3

$$\operatorname{argmax}_y \; w^T \phi(x, y)$$

$$w = \begin{bmatrix} w_1 \\ \vdots \\ w_y \\ \vdots \\ w_n \end{bmatrix}_{nK\times 1} \qquad \phi(x, y) = \begin{bmatrix} 0_n \\ \vdots \\ x \\ \vdots \\ 0_n \end{bmatrix}_{nK\times 1}$$



Combinatorial optimization problem

$$\hat{y} = \mathrm{argmax}_{\boldsymbol{y}\in\boxed{\mathcal{Y}}}\, f(\boldsymbol{y}; \boxed{\boldsymbol{w}}, \boxed{\boldsymbol{x}})$$

input

model parameters

output space

- Inference/Test: given $\boldsymbol{w}, \boldsymbol{x}$, solve argmax
- Learning/Training: find a good $\boldsymbol{w}$

Structured Prediction

- Input: $x \in X$
- Truth: $y^* \in Y(x)$
- Predicted: $h(x) \in Y(x)$
- Loss: $loss(y, y^*)$

I	can	can	a	can
Pro	Md	Vb	Dt	Nn
Pro	Md	Md	Dt	Vb
Pro	Md	Md	Dt	Nn
Pro	Md	Nn	Dt	Md
Pro	Md	Nn	Dt	Vb

Goal: make joint prediction to minimize a joint loss

find $h \in H$ such that $h(\mathrm{x}) \in Y(X)$
minimizing $E_{(x,y)\sim D}\left[loss\left(y, h(x)\right)\right]$ based on N
samples $(x_n, y_n)\sim D$

General log-linear model

- Assumption:

$$P(y|x,w) = \frac{\exp(w^T\phi(x,y))}{\sum_{y'\in Y}\exp(w^T\phi(x,y'))}$$

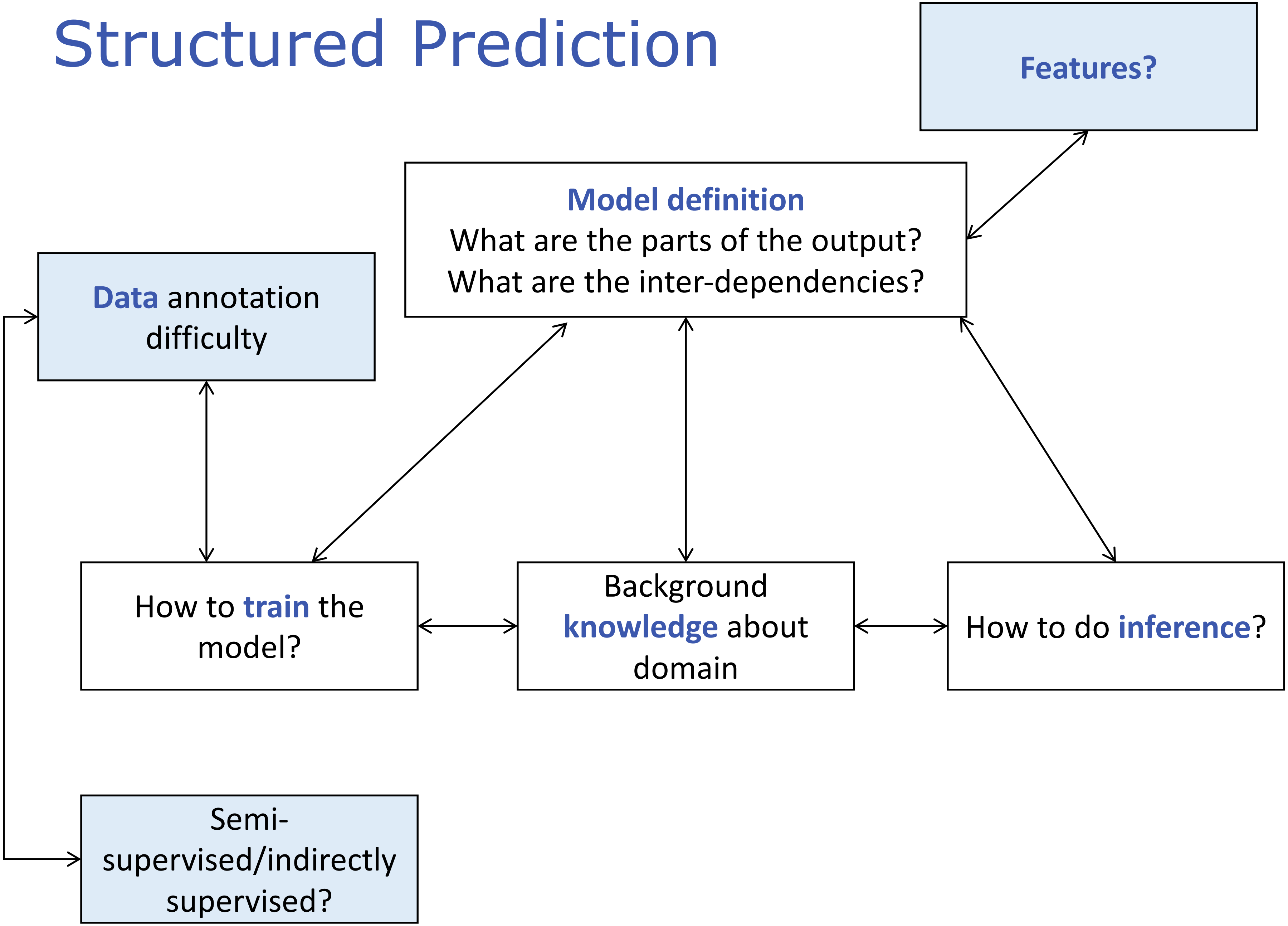
Partition function

Summation over exponentially large output space

Challenges with structured output

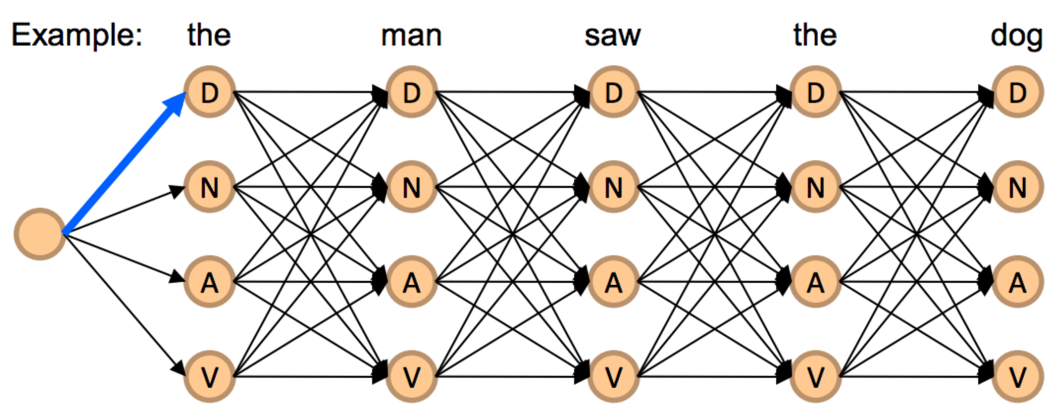
- We cannot train a separate weight vector for each possible inference outcome (why?)
  - For multi-class we train one weight vector for each class
- We cannot enumerate all possible structures for inference
  - Inference for multiclass was easy

Structured Prediction

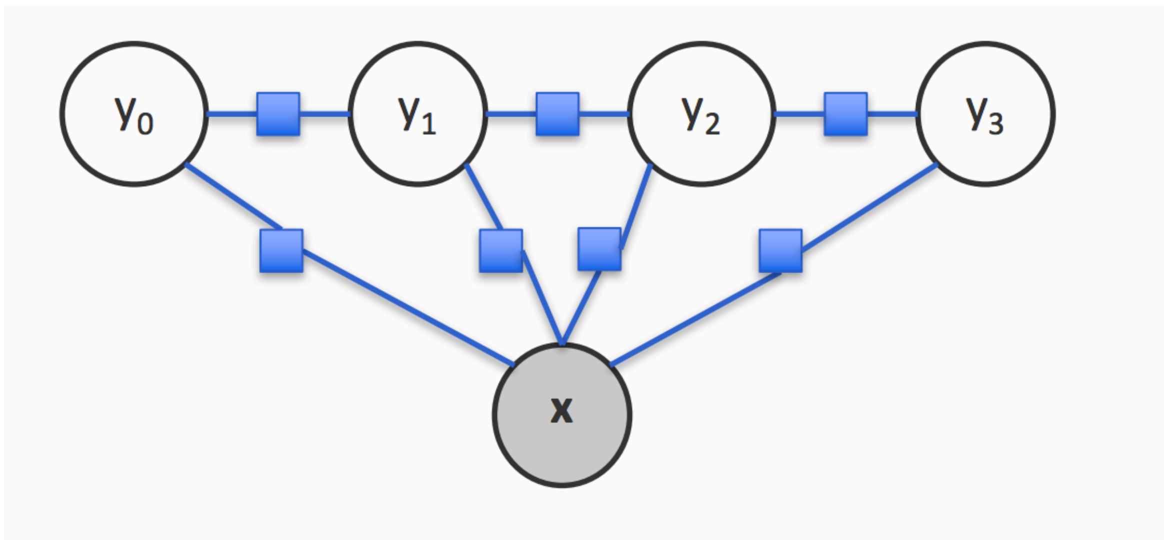


Deal with combinatorial output

- Decompose the output into parts that are labeled
- Define a graph to represent (independent assumption)
  - how the parts interact with each other
  - These labeled interacting parts are scored; the total score for the graph is the sum of scores of each part
  - an inference algorithm to assign labels to all the parts

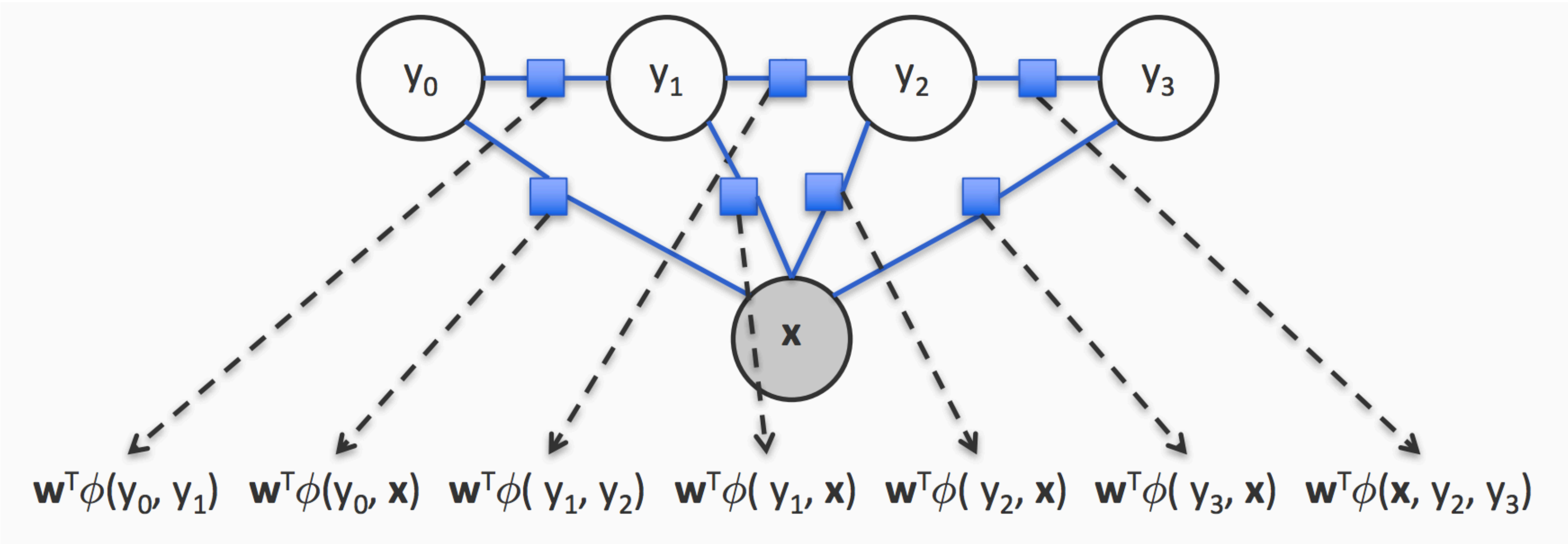


Conditional Random Field: Factor graph



Each node is a random variable
We observe some nodes and need to assign the rest
Each factor is associated with a score

Conditional Random Field: Factor graph



Each node is a random variable

We observe some nodes and need to assign the rest

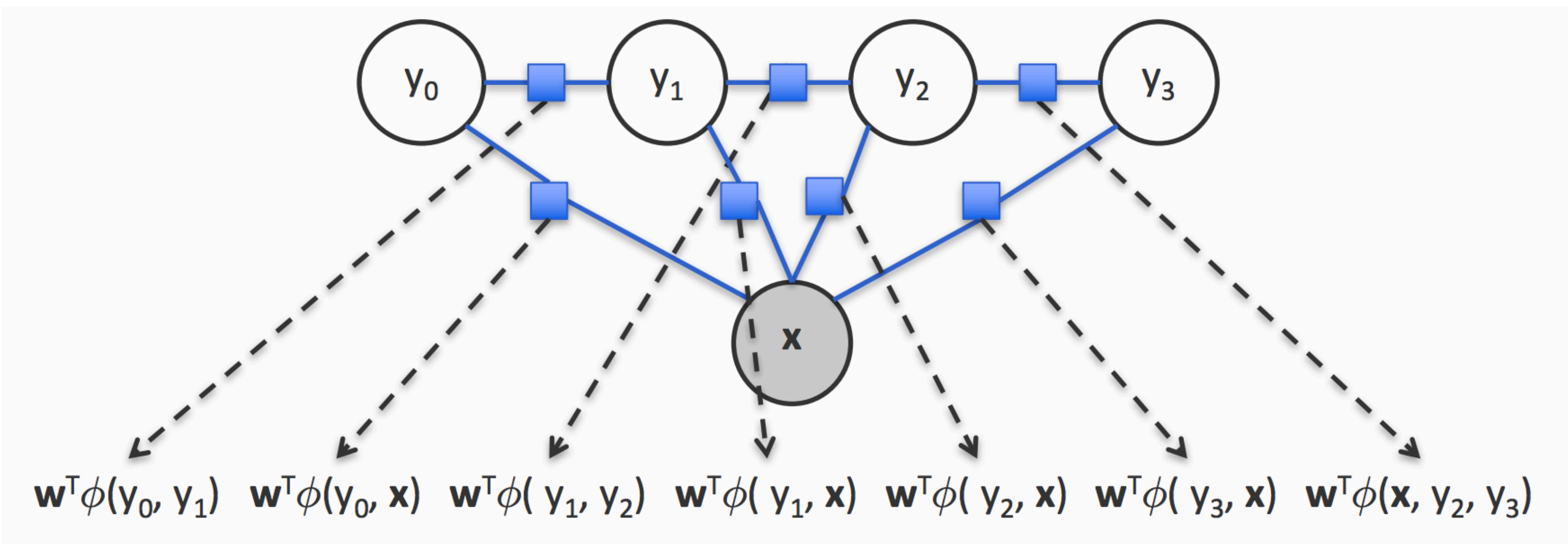
Each factor is associated with a score

Conditional Random Field for sequences

$$P(\boldsymbol{y} \mid \boldsymbol{x}) = \frac{1}{Z} w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_0) \prod_i \exp(\boldsymbol{w}^T \phi(y_i, y_{i-1}) + w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_i))$$

Z: Normalizing constant,
sum over all sequences

$$Z = \sum_y w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_0) \prod_i \exp(\boldsymbol{w}^T \phi(y_i, y_{i-1}) + w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_i))$$

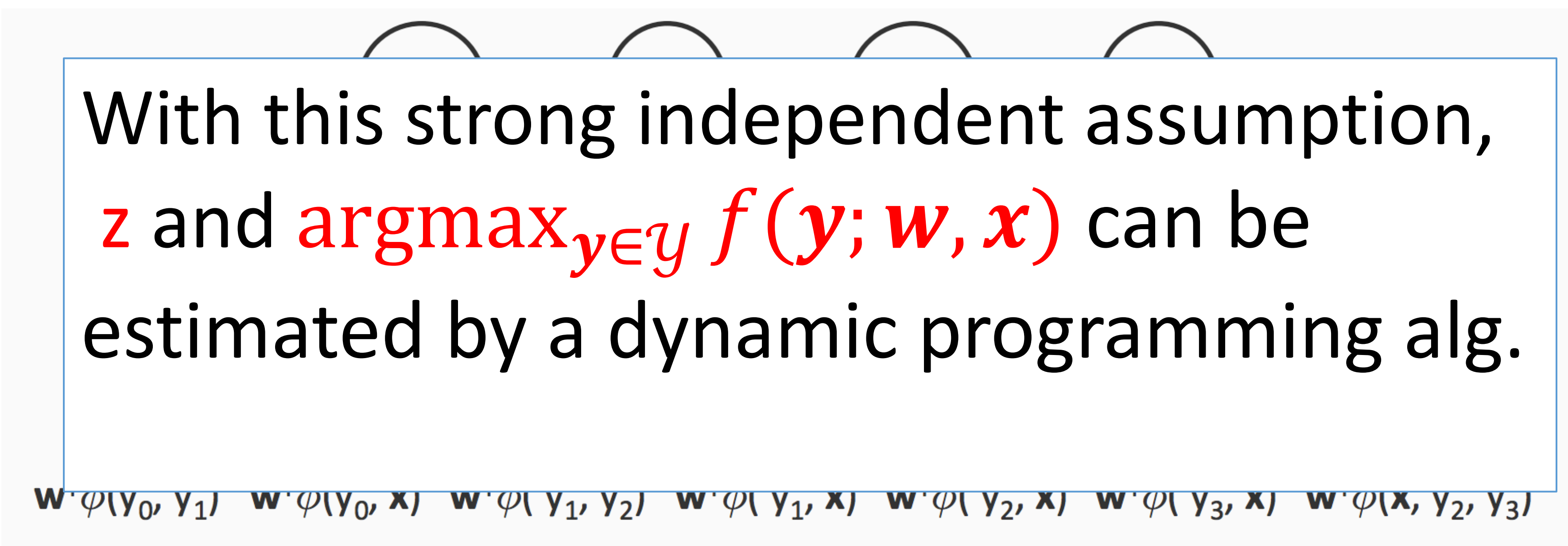


Conditional Random Field for sequences

$$P(\boldsymbol{y} \mid \boldsymbol{x}) = \frac{1}{Z} w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_0) \prod_i \exp(\boldsymbol{w}^T \phi(y_i, y_{i-1}) + w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_i))$$

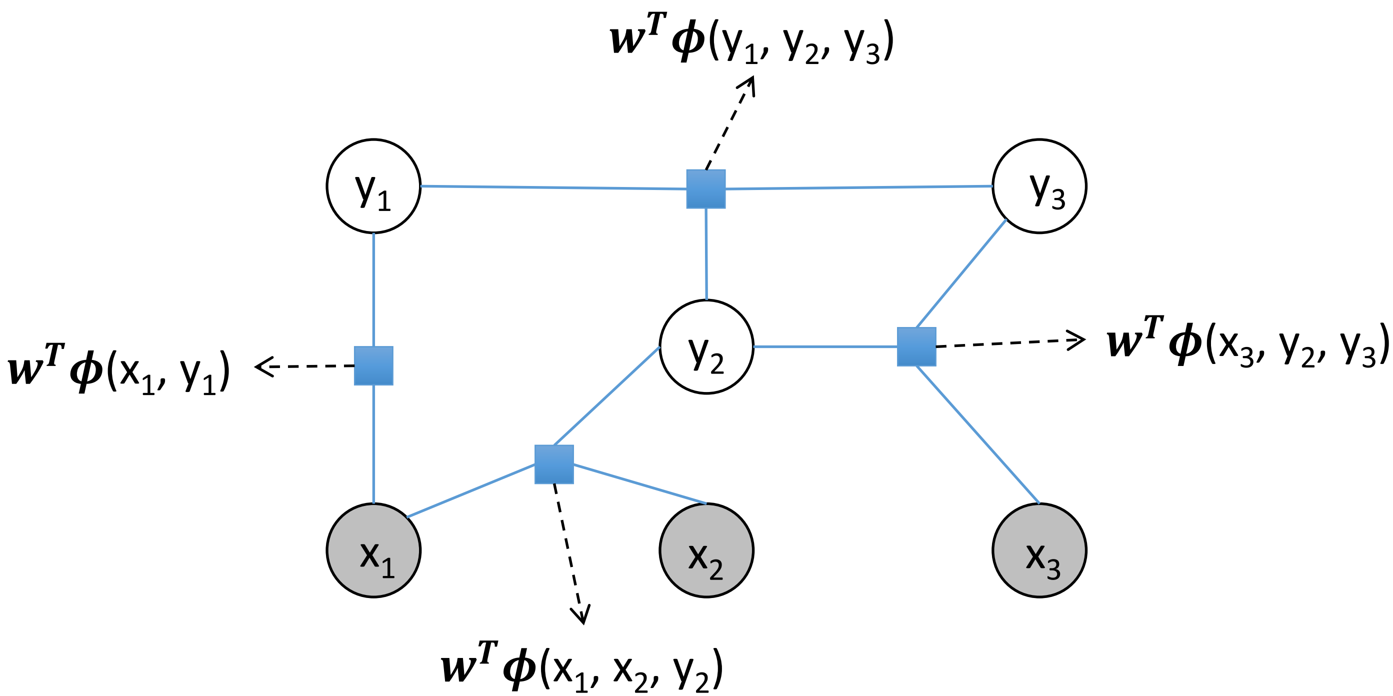
Z: Normalizing constant,
sum over all sequences

$$Z = \sum_y w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_0) \prod_i \exp(\boldsymbol{w}^T \phi(y_i, y_{i-1}) + w^T \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}_i))$$



With this strong independent assumption, z and $\mathrm{argmax}_{y \in \mathcal{Y}} f(\boldsymbol{y}; \boldsymbol{w}, \boldsymbol{x})$ can be estimated by a dynamic programming alg.

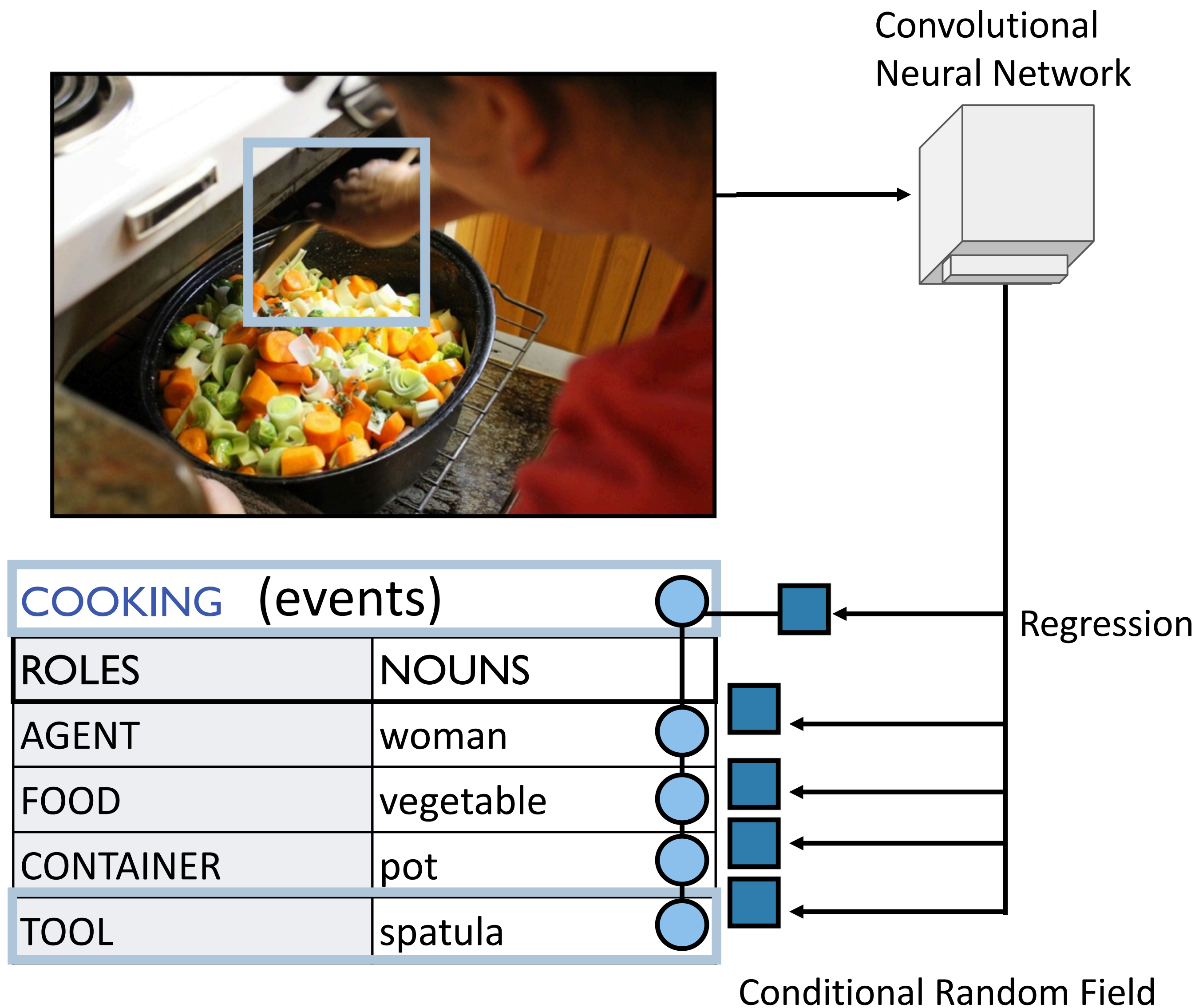
General CRFs



$$P(\mathbf{y}|\mathbf{x}) = \frac{\exp\left(\mathbf{w}^T\phi(\mathbf{x},\mathbf{y})\right)}{\sum_{\hat{\mathbf{y}}}\exp\left(\mathbf{w}^T\phi(\mathbf{x},\hat{\mathbf{y}})\right)}$$

$$\boldsymbol{\phi}(\mathbf{x}, \mathbf{y}) = \boldsymbol{\phi}(x_1, y_1) + \boldsymbol{\phi}(y_1, y_2, y_3) + \boldsymbol{\phi}(x_3, y_2, y_3) + \boldsymbol{\phi}(x_1, x_2, y_2)$$

CRF can be incorporated with deep learning



Yatskar et al. CVPR ’16, Yang et al. NAACL ’16, Gupta and Malik arXiv ’16

Learning in log-linear model

- Assumption:

Partition function

$$P(y|x,w) = \frac{\exp(w^T\phi(x,y))}{\sum_{y' \in Y} \exp(w^T\phi(x,y'))}$$

- Learning:

Summation over exponentially large output space

$$\min_w \ \frac{1}{2}w^T w + \ C\sum_i [\log \sum_{y' \in Y} \exp(w^T\phi(x_i,y')) - w^T\phi(x_i,y_i)]$$

Computational questions

1. Learning: Given a training set $\{<\mathbf{x}_i, \mathbf{y}_i>\}$
   - Train via maximum likelihood (typically regularized)

$$\max_{\mathbf{w}} \sum_i \log P(\mathbf{y}_i|\mathbf{x}_i, \mathbf{w}) = \max_{\mathbf{w}} \sum_i \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i) - \log Z_{\mathbf{w}}(\mathbf{x}_i)$$

   - Need to compute partition function during training

$$Z_{\mathbf{w}}(\mathbf{x}_i) = \sum_{\mathbf{y}} \exp(\mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}))$$

Computational questions

1. Learning: Given a training set {<$\mathbf{x}_i$, $\mathbf{y}_i$>}
   - Train via maximum likelihood (typically regularized)

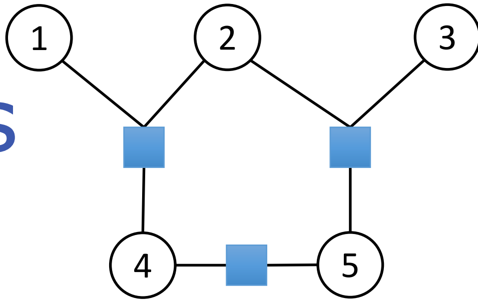
   $$\max_{\mathbf{w}} \sum_i \log P(\mathbf{y}_i|\mathbf{x}_i, \mathbf{w}) = \max_{\mathbf{w}} \sum_i \mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}_i) - \log Z_{\mathbf{w}}(\mathbf{x}_i)$$

   - Need to compute partition function during training

   $$Z_{\mathbf{w}}(\mathbf{x}_i) = \sum_{\mathbf{y}} \exp(\mathbf{w}^T \phi(\mathbf{x}_i, \mathbf{y}))$$

2. Prediction: $\max_{\mathbf{y}} \mathbf{w}^T \phi(\mathbf{x}, \mathbf{y})$
   - Go over all possible assignments to the **y**'s
   - Find the one with the highest probability/score

Inference in graphical models



In general, compute probability of a subset of states

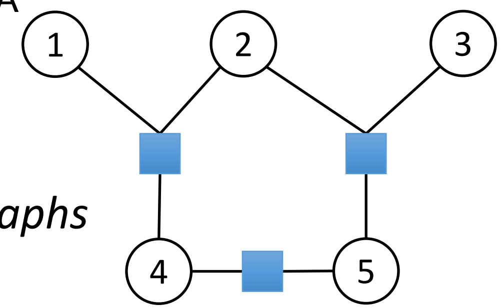
- $P(\mathbf{x}_A)$, for some subsets of random variables $\mathbf{x}_A$

- Exact inference
  - Variable elimination
    - Marginalize by summing out variables in a “good” order i
  - Belief propagation (exact only for graphs without loops)
    - Nodes pass messages to each other about their estimate of what the neighbor’s state should be
  - Generally efficient for trees, sequences (and maybe other graphs too)
- “Approximate” inference

Inference in graphical models

In general, compute probability of a subset of states

- $P(\mathbf{x}_A)$, for some subsets of random variables $\mathbf{x}_A$

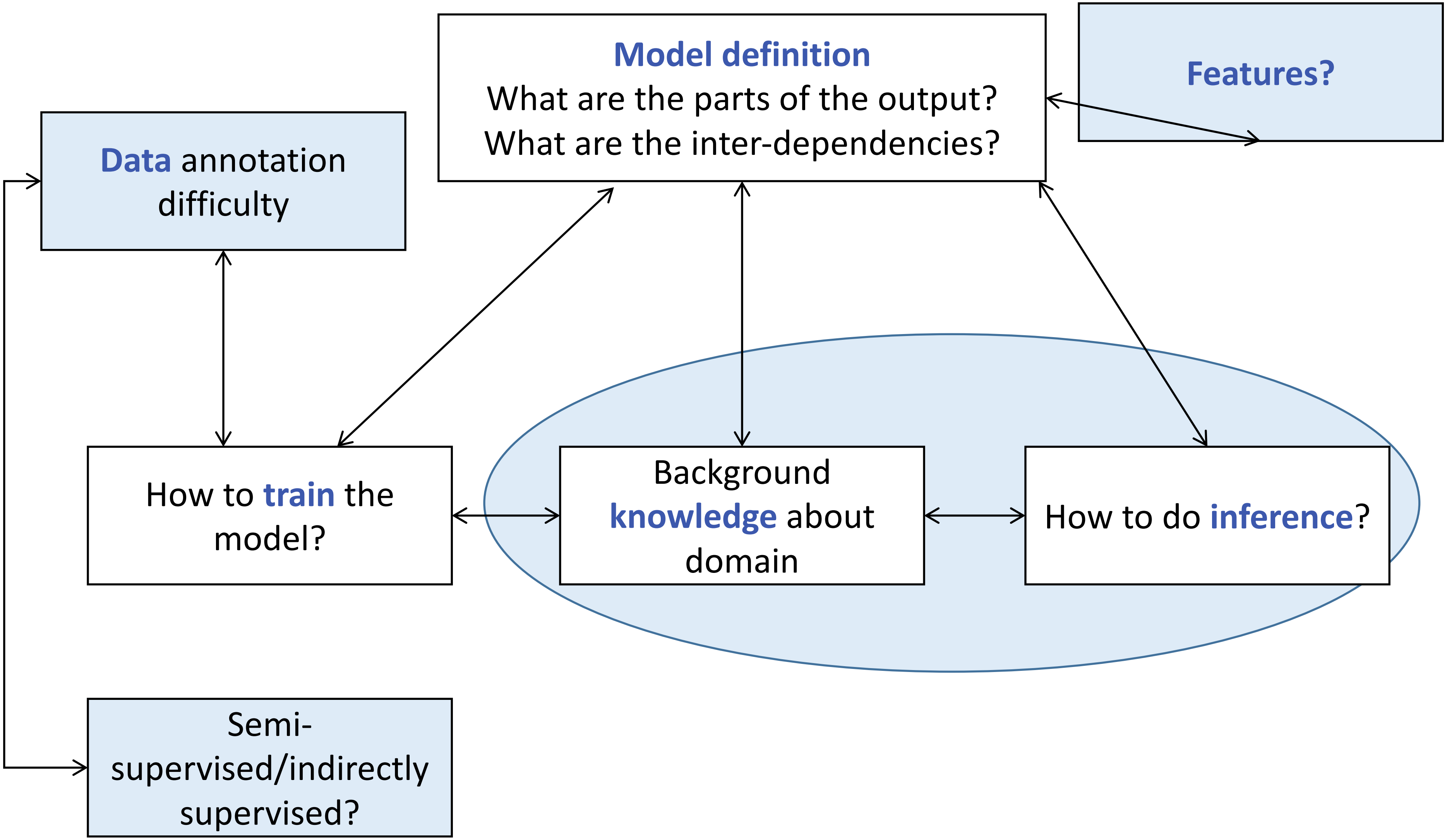


- Exact inference

  NP-hard in general, works for simple graphs

- "Approximate" inference
  - **M**arkov **C**hain **M**onte **C**arlo
    - Gibbs Sampling/Metropolis-Hastings
  - Variational algorithms
    - Frame inference as an optimization problem, perturb it to an approximate one and solve the approximate problem
  - Loopy Belief propagation
    - Run BP and hope it works!

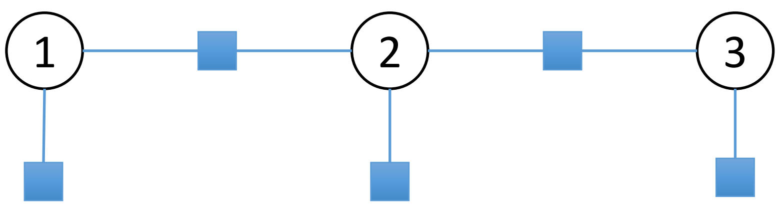
Constrained Conditional Model



Consistency of outputs

Or: How to introduce knowledge into prediction

Suppose we have a sequence labeling problem where the outputs can be one of A or B



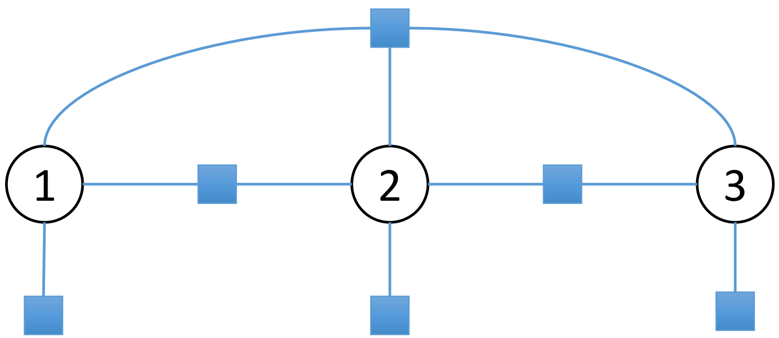
Consistency of outputs

Or: How to introduce knowledge into prediction

Suppose we have a sequence labeling problem where the outputs can be one of A or B

We want to add a condition:

There should be no more than one B in the output



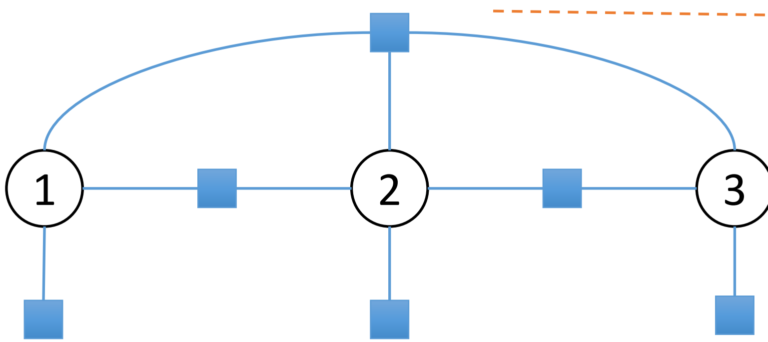
Consistency of outputs

Or: How to introduce knowledge into prediction

Suppose we have a sequence labeling problem where the outputs can be one of A or B

We want to add a condition:

There should be no more than one B in the output



Potential function that ensures this condition

y1	y2	y3	f
A	A	A	0
A	A	B	0
A	B	A	0
A	B	B	-1
B	A	A	0
B	A	B	-1
B	B	A	-1
B	B	B	-1

Should we learn what we can write down easily? Especially for such large, computationally cumbersome factors

Entity Relation Extraction task

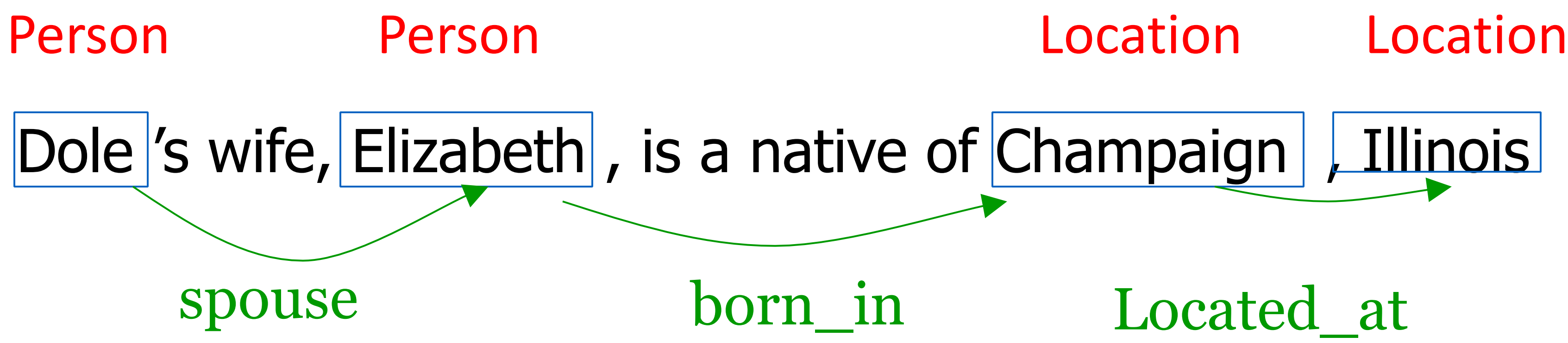
Dole 's wife, Elizabeth , is a native of Champaign , Illinois .

Person Person Location Location

spouse born_in Located_at

- Consistency constraint: A spouse relation can only hold between two person entities and cannot hold between two location entities

ILP Formulation for Entity Relation Task



Dole

PER	0.5
LOC	0.3
ORG	0.2

Elizabeth

PER	0.6
LOC	0.1
ORG	0.3

Dole-Elizabeth

spouse	0.7
born_in	0.1
Located_at	0.1
No-relation	0.1

ILP Formulation

Dole

PER	0.5	y1
LOC	0.3	y2
ORG	0.2	y3

Elizabeth

PER	0.6	y4
LOC	0.1	y5
ORG	0.3	y6

Dole-Elizabeth

spouse	0.7	y7
born_in	0.1	y8
Located_at	0.1	y9
No-relation	0.1	y10

maximize

$0.5y1 + 0.3y2 + 0.2y3 +$

$0.6y4 + 0.1y5 + 0.3y6 +$

$0.7y7 + 0.1y8 + 0.1y9 + 0.1y10$

subj to $yi \in \{0,1\}$

$y1 + y2 + y3 = 1$

$y4 + y5 + y6 = 1$

$y7 + y8 + y9 + y10 = 1$

$2y7-y1-y4<=0$

A spouse relation can only hold between two person entities

Amortized Inference for ILP

- We can write the ILP as

$$\max_{\mathbf{y}} \mathbf{c}\mathbf{y}$$

$$A\mathbf{y} \leq \mathbf{b}$$

$$\mathbf{y}_i \in \{0,1\}$$

- Inference problems discussed in previous sections can be represented as 0-1 ILPs.

maximize

$0.5y1 + 0.3y2 + 0.2y3 +$

$0.6y4 + 0.1y5 + 0.3y6 +$

$0.7y7 + 0.1y8 + 0.1y9 + 0.1y10$

subj to $yi \in \{0,1\}$

$y1 + y2 + y3 = 1$

$y4 + y5 + y6 = 1$

$y7 + y8 + y9 + y10 = 1$

$2y7-y1-y4<=0$

Inference with constraints

- Combinatorial optimization problems can be often written as integer linear programs (ILP)
  - The conversion is not always trivial
  - Allows injection of "knowledge" in the form of constraints

- Different ways of solving ILPs
  - Commercial solvers: CPLEX, Gurobi, etc
  - Specialized solvers if you know something about your problem
    - Lagrangian relaxation, amortized inference, etc
  - Can approximate to linear programs and hope for the best

Integer linear programming

- In general

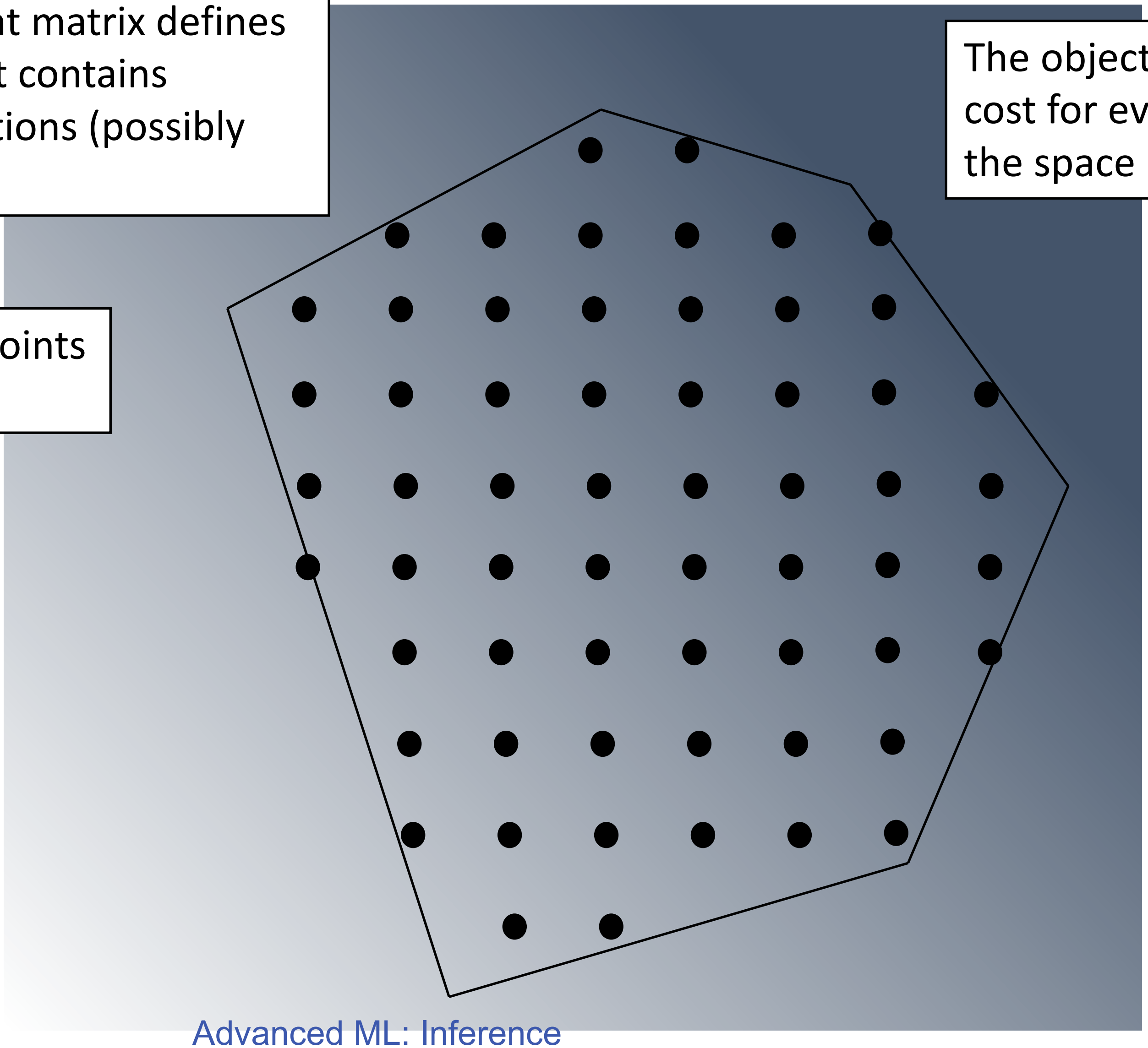
$$\begin{aligned} \max \quad & \mathbf{c}^T\mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0 \\ & \boxed{\text{Each } x_i \text{ is an integer.}} \end{aligned}$$

Geometry of integer linear programming

The constraint matrix defines polytope that contains allowed solutions (possibly not closed)

The objective defines cost for every point in the space

Only integer points allowed

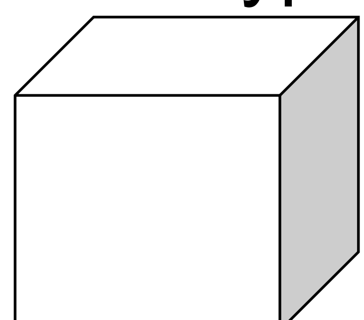


0-1 integer linear programming

- In general

$$
\begin{aligned}
\max \quad & \mathbf{c}^T \mathbf{x} \\
\text{subject to} \quad & A\mathbf{x} \leq \mathbf{b} \\
& \mathbf{x} \geq 0 \\
& \boxed{\mathbf{x} \in \{0, 1\}^n}
\end{aligned}
$$

- An instance of integer linear programs
  - Still NP-hard
- Geometry: We are only considering points that are vertices of the Boolean hypercube



Back to structured prediction

$$\hat{y} = \operatorname{argmax}_{\boldsymbol{y} \in \boxed{\mathcal{Y}}} f(\boldsymbol{y}; \boldsymbol{w}, \boldsymbol{x})$$

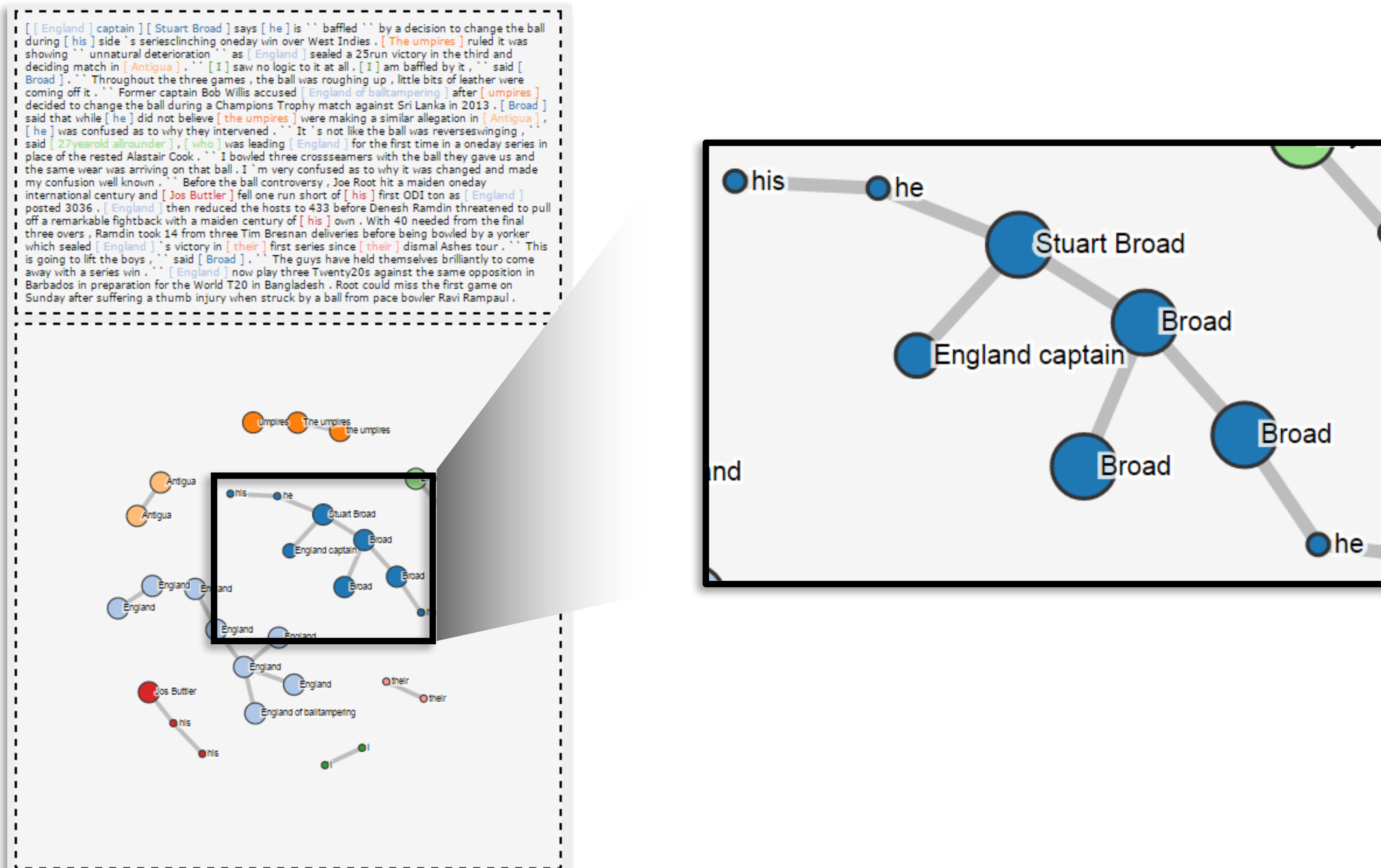
output space
+ expert constraints

- General idea: Frame the argmax problem as a 0-1 integer linear program
  - Allows addition of arbitrary constraints

Example application:
Co-reference Resolution

Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh**. As a **boy**, **Chris** lived in a pretty home called **Cotchfield Farm**. When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

Co-reference Resolution Demo



Co-reference Resolution

- Learn a pairwise similarity score function (local predictor)

  Example features:
  - same sub-string?
  - positions in the paragraph
  - other 30+ feature types
- Key components:
  - Pairwise classification
  - Clustering (jointly or not?)

Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh**. As a **boy**, **Chris** lived in a pretty home called **Cotchfield Farm**. When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

Approaches

Approach1: Decoupling Approach(CoNLL11)

Learn a pairwise similarity function

Cluster based on this function

Approach2: Latent Structured Learning (ICML 14, EMNLP 13)

Update the similarity function

Cluster based on this function.

Baseline: Decoupling Approach

A heuristic to learn the model [Soon+ 01, Bengtson+ 08,CoNLL11]

❖ Decouple learning and inference:

Learn a pairwise similarity function

Cluster based on this function

Decoupling Approach-Learning

As a **boy**, **Chris**$_1$ lived in a pretty home called **Cotchfield Farm.** When **Chris**$_2$ was three years old, **his father**$_3$ wrote a poem about **him**$_4$. The poem was printed in a magazine for others to read. **Mr. Robin**$_5$ then wrote a book

Positive Samples

(Chris$_1$, him$_4$)

(Chris$_2$, him$_4$)

(Chris$_1$, Chris$_2$)

(his father$_3$, Mr. Robin$_5$)

Negative Samples

(Chris$_1$, his father$_3$)

(Chris$_2$, his father$_3$)

(him$_4$, his father$_3$)

(Chris$_1$, Mr. Robin$_5$)

(Chris$_2$, Mr. Robin$_5$)

(him$_4$, Mr. Robin$_5$)

Greedy Best-Left-Link Clustering

[Bill Clinton], recently elected as the **[President of the USA]**, **has been invited by the [Russian President], [Vladimir Putin], to visit [Russia]. [President Clinton] said that [he] looks forward to strengthening** ties between **[USA]** and **[Russia]**.

Greedy Best-Left-Link Clustering

[Bill Clinton], recently elected as the **[President of the USA]**, has been invited by the **[Russian President]**, **[Vladimir Putin]**, **to visit [Russia]. [President Clinton] said that [he] looks forward to strengthening** ties between **[USA]** and **[Russia]**.

Greedy Best-Left-Link Clustering

[Bill Clinton], recently elected as the **[President of the USA]**, has been invited by the **[Russian President]**, **[Vladimir Putin]**, to visit **[Russia]**. **[President Clinton]** said that **[he]** looks forward to strengthening ties between **[USA]** and **[Russia]**.

Greedy Best-Left-Link Clustering

[Soon+ 01, Bengtson+ 08, CoNLL11]

[Bill Clinton], recently elected as the **[President of the USA]**, has been invited by the **[Russian President]**, **[Vladimir Putin]**, to visit **[Russia]**. **[President Clinton]** said that **[he]** looks forward to strengthening ties between [USA] and **[Russia]**.

[Bill Clinton]

[President of the USA]

[President Clinton]

[he]

[Russian President]

[Vladimir Putin]

[Russia].

[Russia].

Best Left-Linking Forest

Can we do better?

- Decoupling may lose information

Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh**. As a **boy**, **Chris** lived in a pretty home called **Cotchfield Farm.** When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

Structured Learning Approach

Update the similarity function

Cluster based on this function.

Learn the similarity function while clustering

Attempt: All-Links Clustering

[Mccallum+ 04, CoNLL 11]

- Define a global scoring function:

  Attempt: using all within-cluster pairs:

  - Inference problem is too hard

Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh**. As a **boy**, **Chris** lived in a pretty home called **Cotchfield Farm.** When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

Latent Left-Linking Model (L3M)

[ICML 14, EMNLP 13]

Score (a clustering C)
= Score (the best left-linking forest that is consistent with C)
= $\sum$ Score of edges in the forests

Christopher Robin is alive and well. **He** is the same person that you read about in the book, **Winnie the Pooh**. As a **boy**, **Chris** lived in a pretty home called **Cotchfield Farm.** When **Chris** was three years old, **his father** wrote a poem about **him**. The poem was printed in a magazine for others to read. **Mr. Robin** then wrote a book

Linguistic Constraints

- Must-link constraints:
  - E.g., SameProperName, …
- Cannot-link constraints:
  - E.g., ModifierMismatch, …

[Bill Clinton], recently elected as the [President of the USA], has been invited by the [Russian President], [Vladimir Putin], to visit [Russia]. [President Clinton] said that [he] looks forward to strengthening ties between [USA] and [Russia].

- Clustering with constraints[(Basu+08, Zhi+14)]

Inference in L3M [ICML 14, EMNLP 13]

- Represented using an ILP formulation[Scott+ 2004/2007]
- Inference can be done using a greedy heuristics.

$y_{i,j} = 1 \iff i,j$ is an edge in the forest

$$\arg\max_{\boldsymbol{y}} \quad \textstyle\sum_c S_{i,j} \boxed{y_{i,j}}$$
$$s.t \quad \boxed{A\boldsymbol{y} \le b;} \; y_{i,j} \in \{0,1\}$$

- Modeling constraints
- Linguistic constraints

Log linear model: Probabilistic L3M

[ICML 14, EMNLP 13a]

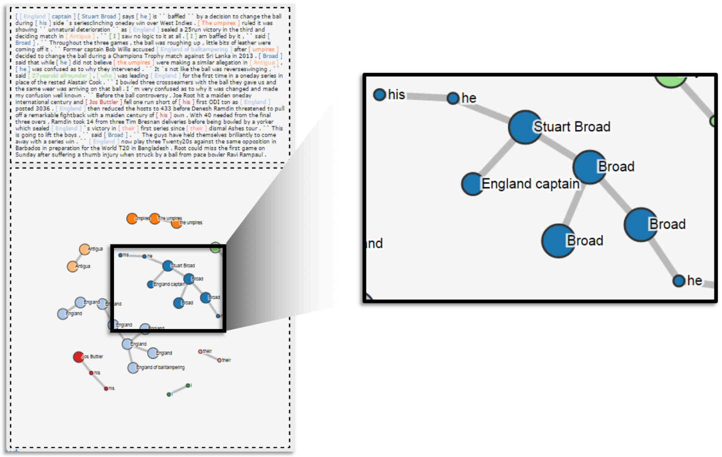
❖Define a log-linear model

Pr [a clustering C]
= ∑ Pr [forests that are consistent with C]
= ∑Π Pr [edges in the forest]
= $\Pi_i \sum_{j\in e(i)}$ Pr [edge(j,i)]
Pr [edge(j,i)] ~ $\exp(\boldsymbol{w} \cdot \phi(j,i)/\gamma)$ (γ: a parameter)

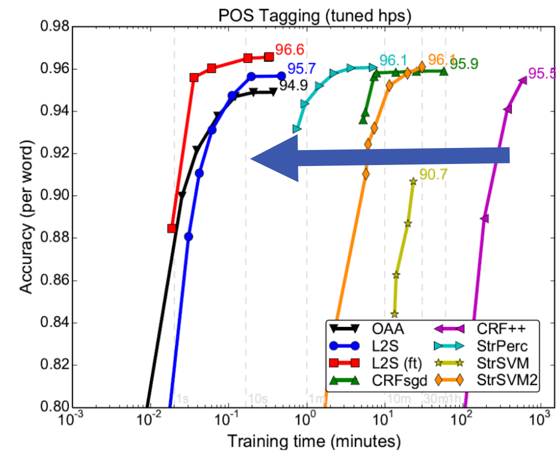
❖Regularized Maximum Log-Likelihood Estimation:

$$\min_{\mathbf{w}} \ \mathrm{LL}(\mathbf{w}) = \beta||\mathbf{w}||^2 + \sum_d \log Z_d(\boldsymbol{w}) - \sum_d \sum_i \log\left(\sum_{j<i} \exp(\boldsymbol{w} \cdot \phi(i,j)/\gamma) C_d(i,j)\right)$$

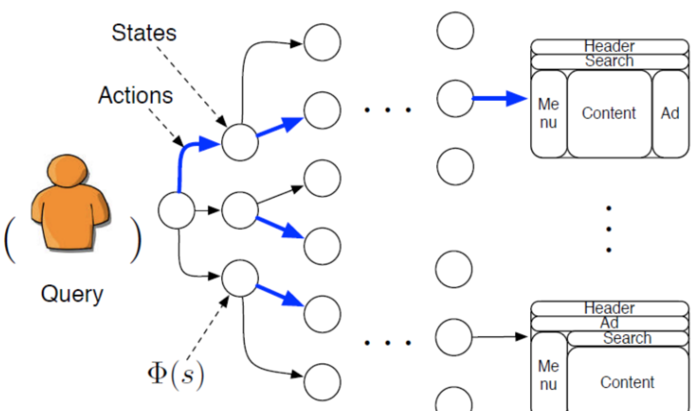
Structured Prediction Models



How to model?



Training/test/dev speed



Learning signals

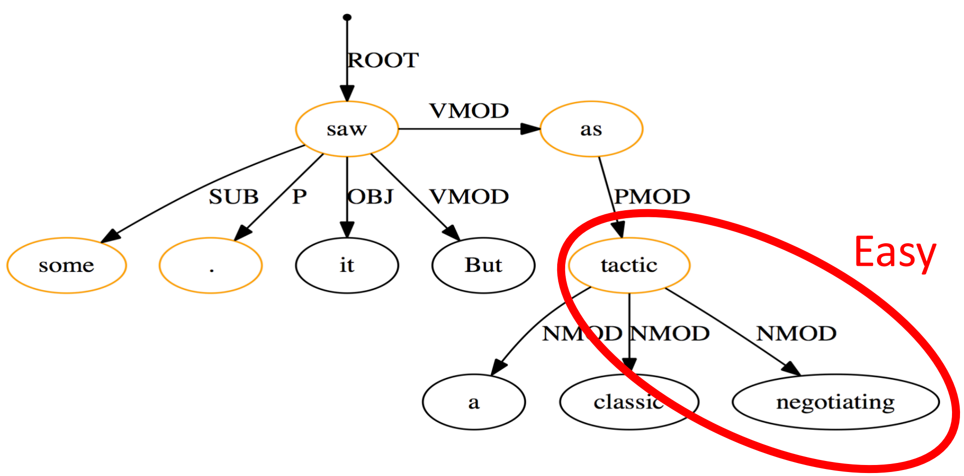
activity	cooking
agent	woman
food	vegetable

Fairness (data biases)

Idea 1: Adaptive feature selection [AAAI 17]

- **Observation:** some decisions are simpler than the others
- **Idea:** adaptively generate computationally costly features during test-time

But some saw it as a classic negotiating tactic



Idea 2: Amortized inference

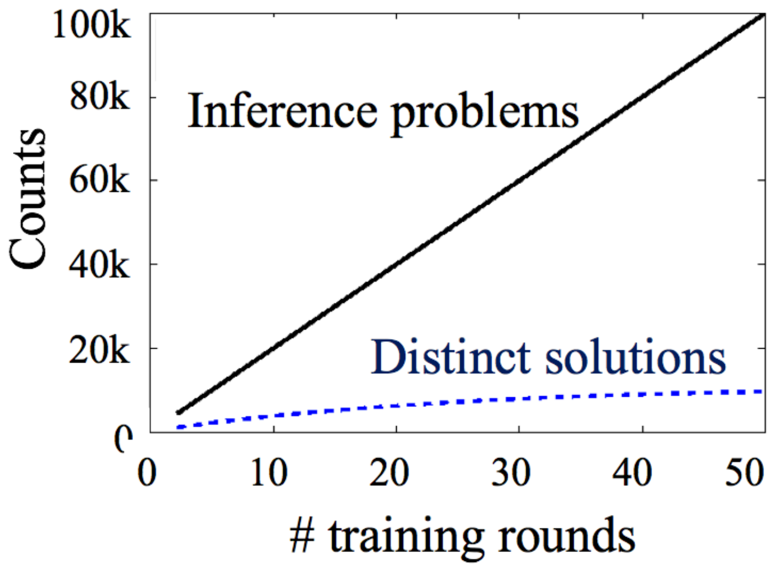
- Observation: Many inference problems share the same solution

S1	POS
He	Pronoun
is	VerbZ
reading	VerbG
a	Det
book	Noun

POS	S2
Pronoun	She
VerbZ	is
VerbG	watching
Det	a
Noun	movie

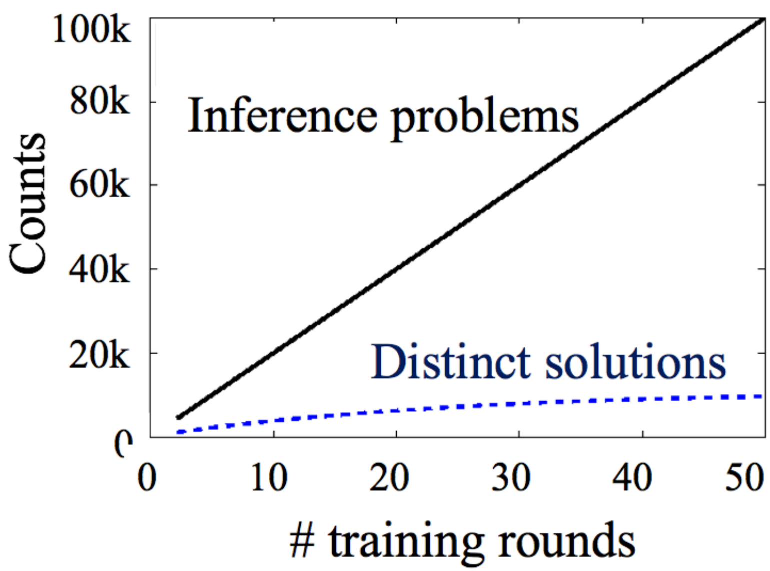
Idea 2: Amortized inference

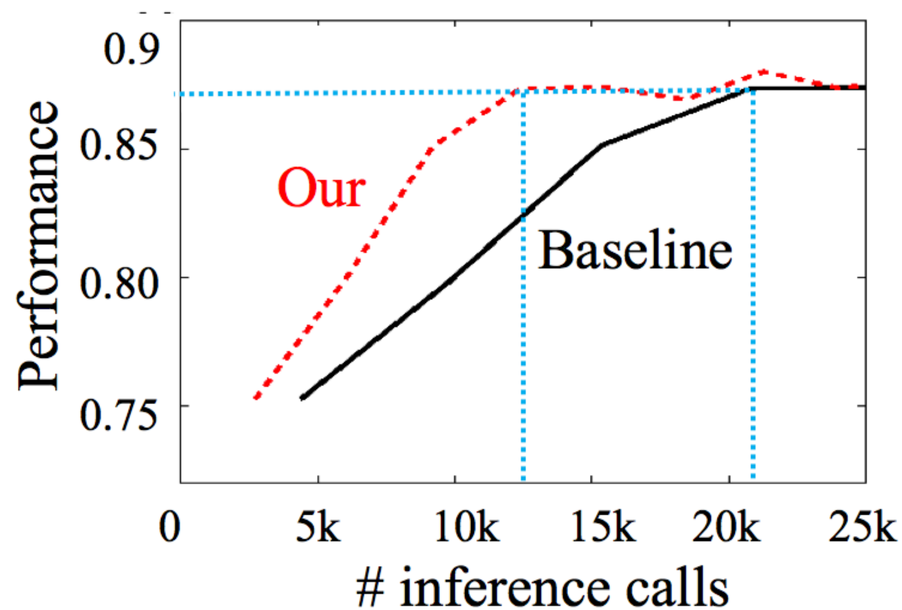
- Idea: Exploit this redundancy by caching old inference solutions [AAAI 15]



Idea 2: Amortized inference

- Idea: Exploit this redundancy by caching old inference solutions [AAAI 15]





Amortized inference – key components

- A general inference framework

  … to represent inference problems

- A condition

  … to check if two problems have the same solution

```
If CONDITION(problem cache, new problem)
   then (no need to call the solver)
        SOLUTION(new problem) = old solution
Else
        Call base solver and update cache
End
```

0.04 ms

2 ms

Solution Methods

- Assume a graphical structure; optimize
  - Use within various structured predictions algorithms (e.g., CRF, Structured Perceptron, M3N, Structured SVM) [Lafferty+ 01, Collins02, Taskar04]
  - See our AAAI16 tutorial (https://goo.gl/TF7cGj)
- Learning to search approaches
  - Assume the complex decision is incrementally constructed by a sequence of decisions
  - E.g., LASO, dagger, Searn, transition-based methods
  - See our NAACL15 tutorials (http://hunch.net/~l2s)

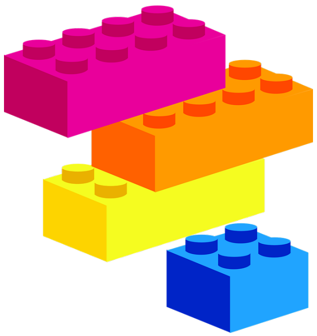
Libraries for Structured Predictions

- Illinois-SL: graph-based structured prediction
  - Support various algorithms; parallel ⇒ very fast
- Vowpal-Wabbit: credit assignment compiler
  - A general online learning library
  - Support search-based structured prediction

Learning to search approaches: Credit Assignment Compiler [NIPS16]

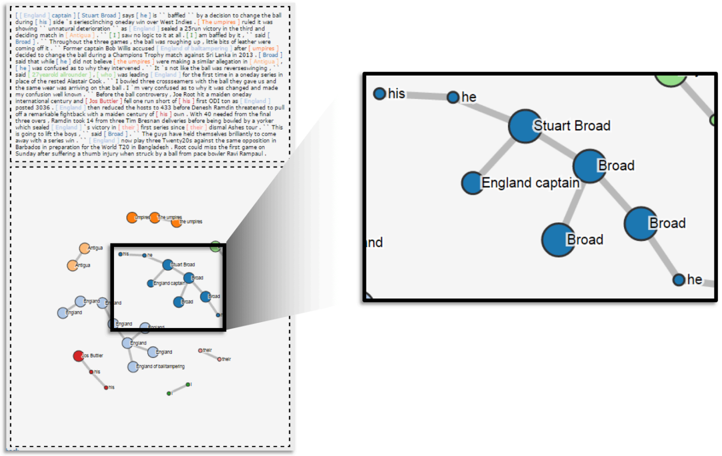
Sequential_RUN(*examples*)

```
for i = 1 to len(examples) do
    prediction ← predict(examples[i], examples[i].label)
    loss(prediction ≠ examples[i].label)
end for
```

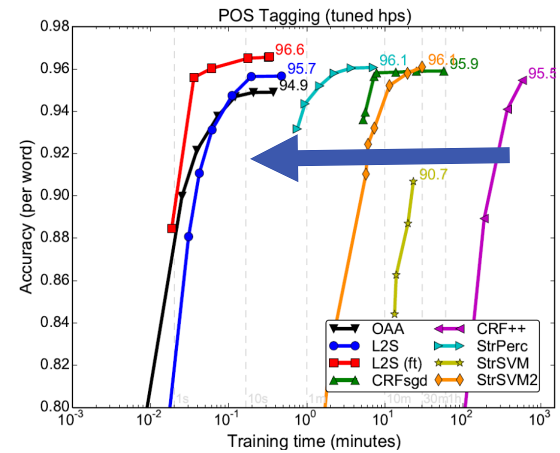


- Write the decoder, providing some side information for training
- Library translates this piece of program with data to the update rules of model
- Applied to dependency parsing, Name entity recognition, relation extraction, POS tagging…

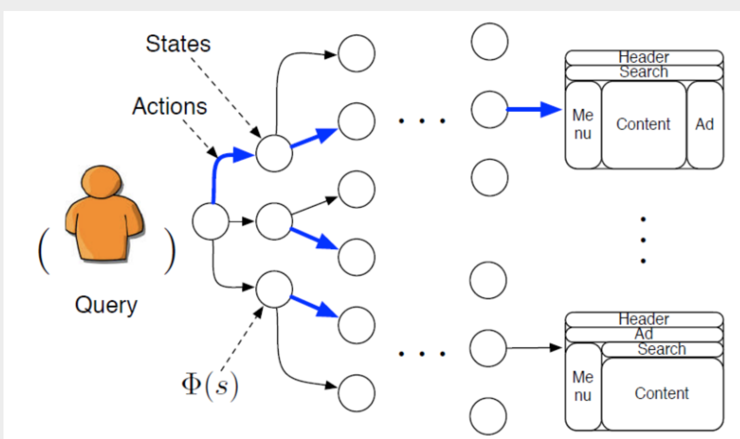
Outline



NLP Applications



Training/test speed



Learning signals

Query

activity	cooking
agent	woman
food	vegetable

Fairness (data biases)

Learning with Implicit/Partial Supervision

[ICML15, EMNLP16, AAAI workshop17]

- Consider algebra word problem

Maria is now four times as old as Kate. Four years ago, Maria was six times as old as Kate. Find their ages now.

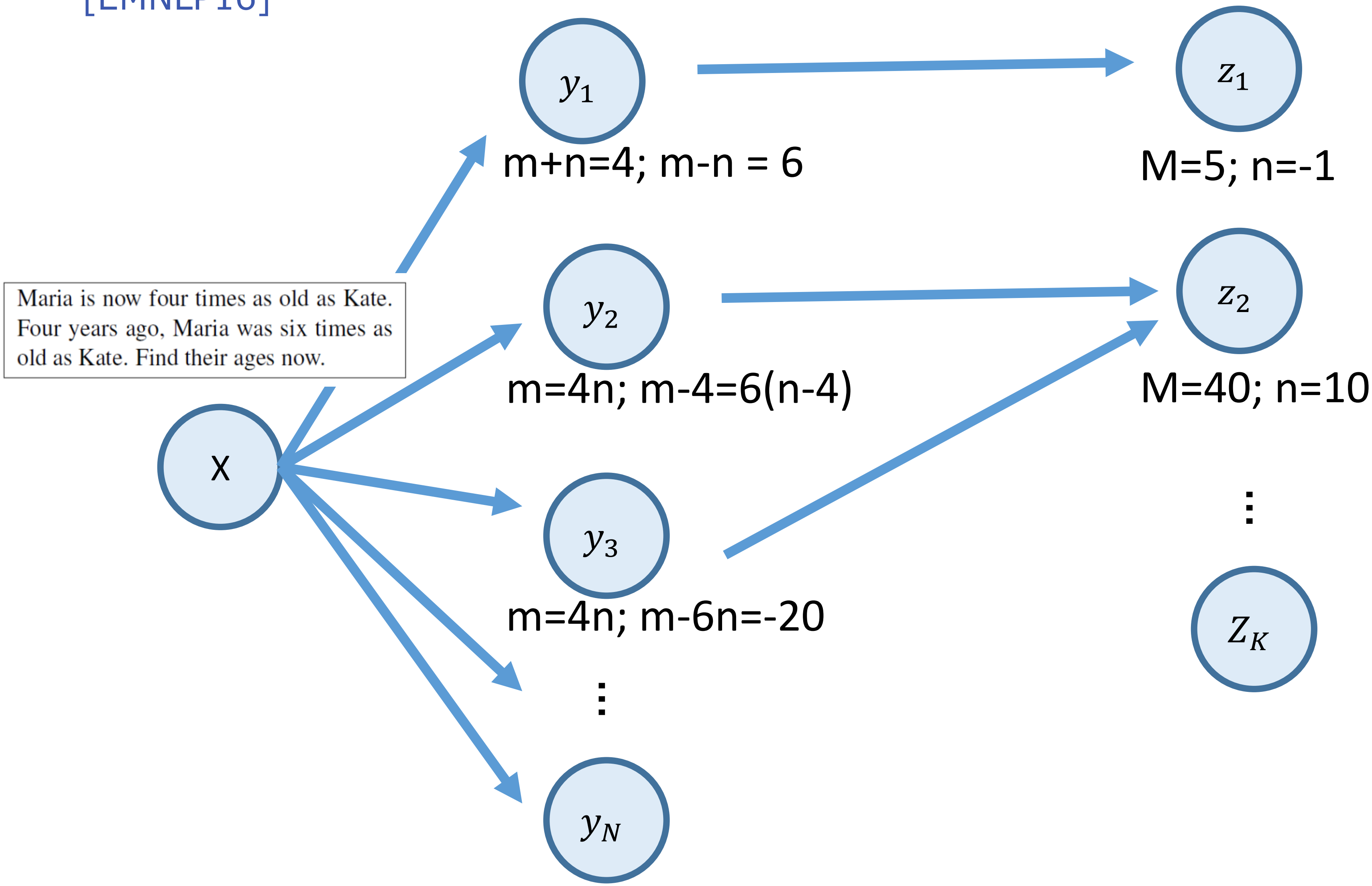
- Build semantic parser to translate question to an equation system

$$m = 4 \times n \text{ and } m - 4 = 6 \times (n - 4).$$

- Then answer can be derived: m=40, n=10

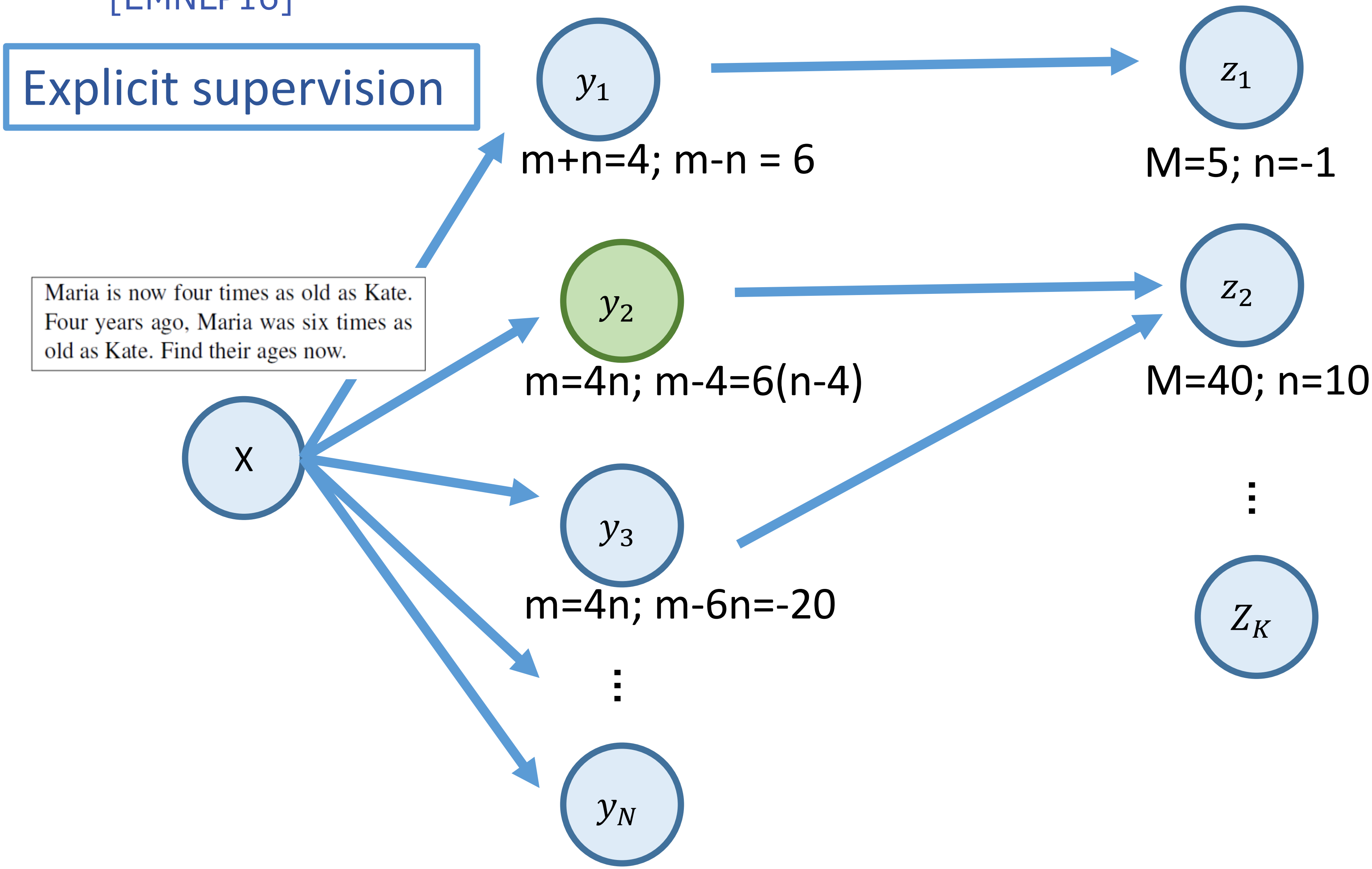
Learning with Implicit/Partial Supervision

[EMNLP16]



Learning with Implicit/Partial Supervision

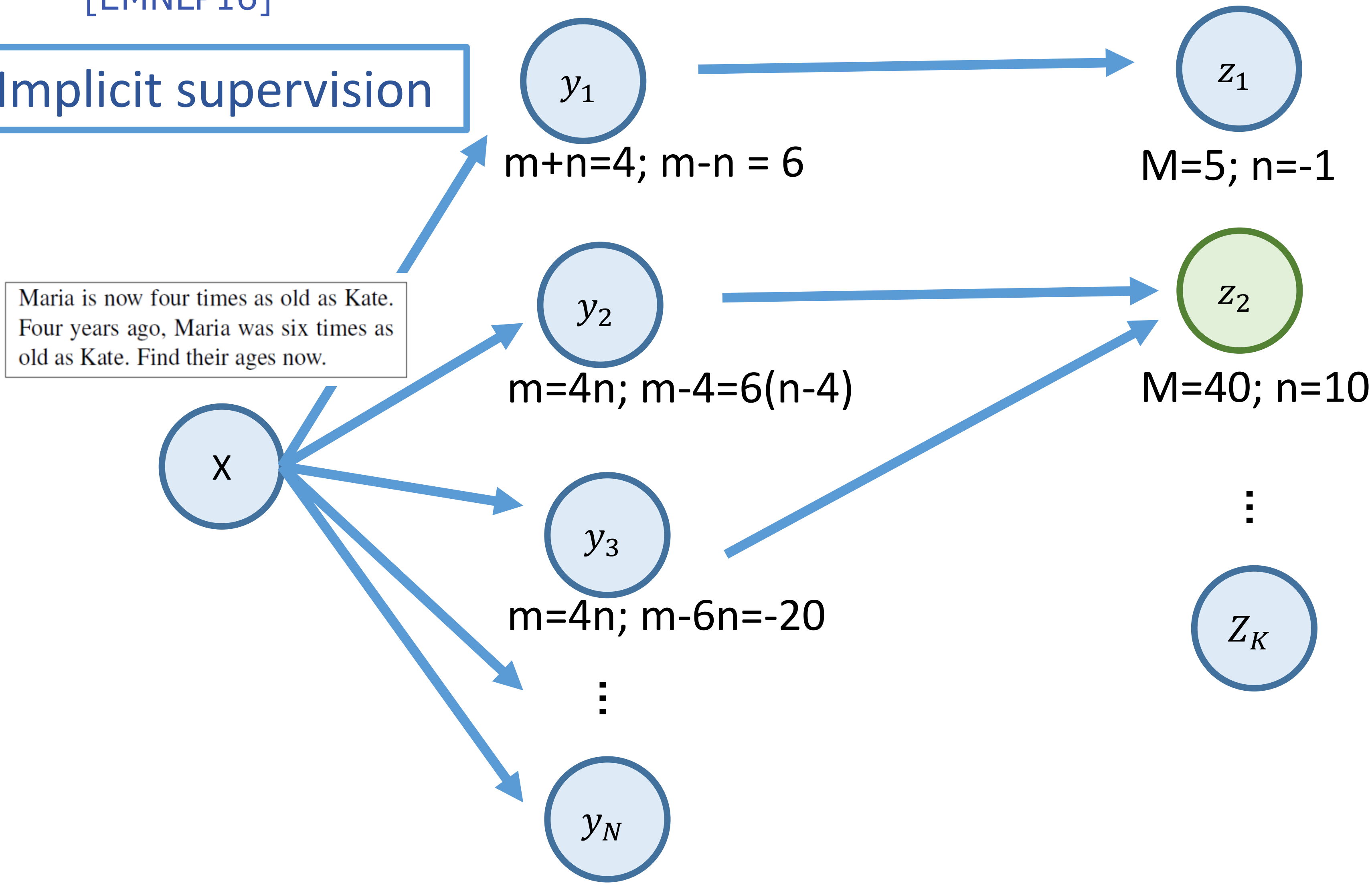
[EMNLP16]



Learning with Implicit/Partial Supervision

[EMNLP16]

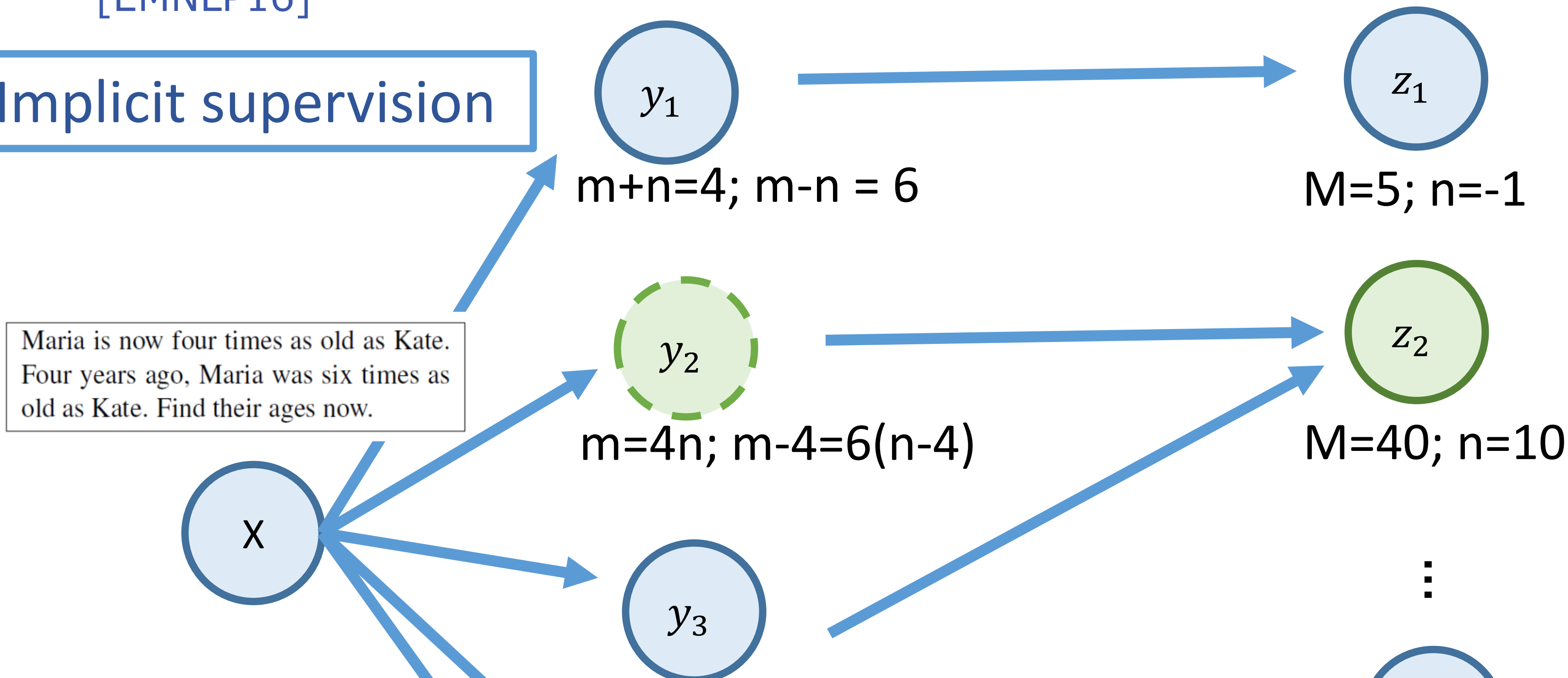
Implicit supervision



Learning with Implicit/Partial Supervision

[EMNLP16]

Implicit supervision



Key idea: Use model to guide exploration on implicit supervision
⇒ Infer plausible equation systems

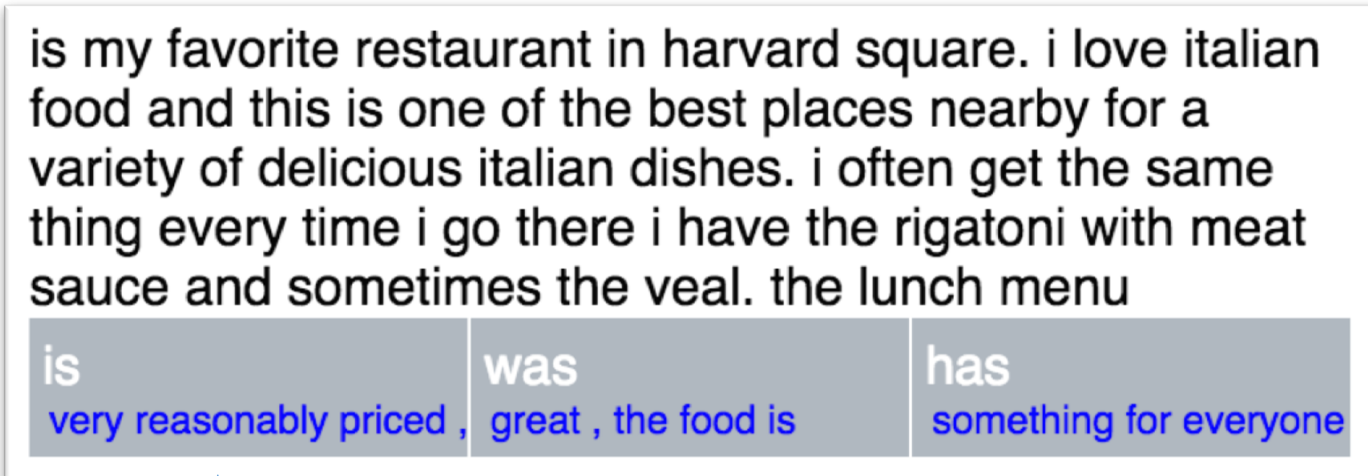
Our approach: Use the best equation system in the beam whose solution matches the implicit signal

Structured Contextual Bandit Setting

[ICML15, AAAI workshop17, IJCNLP 17]

w/ Akshay Krishnamurthy, Alekh Agarwal, Hal Daume; III, John Langford
w/ Kenneth Arnold, Adam Kalai

- Loss of only a single structured label can be observed



Blue words preview the phrase that would be inserted with repeated taps

Prediction ≠ Suggestion, especially in writing

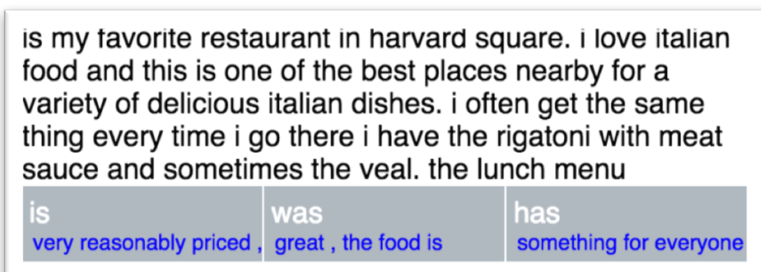
- Baseline: an N-gram model doesn't work
- Accurate predictions may be poor suggestions
  - E.g., most frequent phrases in Yelp:
    this place is great!; Love it, love it, love it!; I love this place
  - E.g., most frequent short email replies [Kannan+ 16]:
    I love you; Thanks; sounds good
- Writers prefer suggestions occur less frequently but more creative/enthusiastic/informative:
  - e.g., this was truly a wonderful experience

Key ideas

1. Discriminative language model
   - Allow to add features
     ⇒ Tune the model to favor phrases with some properties

Key ideas

1. Discriminative language model
2. Counterfactual learning
   - Cannot get full feedback
     - Only candidates presented to users are annotated
     - How can we estimate the model parameter unbiasedly?

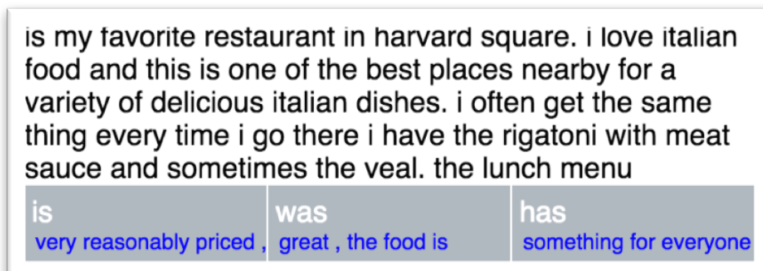


5% 3% 7%

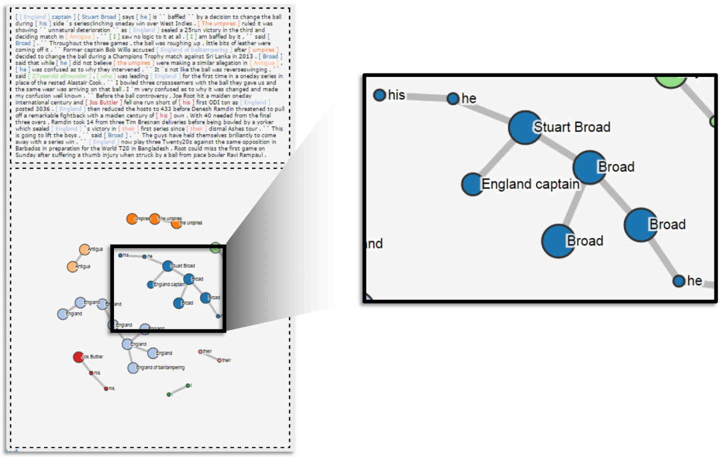
is very delicious	2%
has a drink option	1%
is not worthwhile	6%
is good.	8%
…	
…	

Counterfactual learning

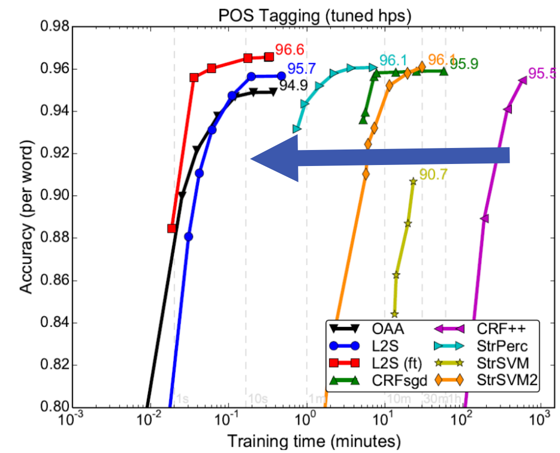
- Collect behavior data using a reference model (suggested phrase, reward, probability, context)
  - Reference model: a tri-gram language model with Kneser-Ney smoothing & temperature parameter
  - reward = # words accepted



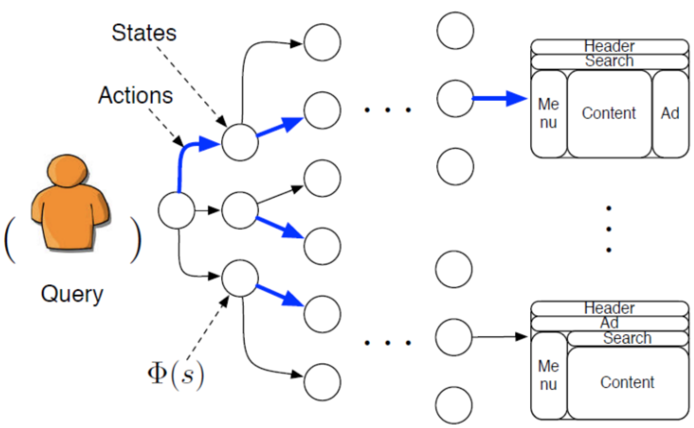
Outline



NLP Applications



Training/test speed



Learning signals

activity	cooking
agent	woman
food	vegetable

Fairness (data biases)

Select photo

× The photo you want to upload does not meet our criteria because:

- Subject eyes are closed

Please refer to the technical requirements.
You have 9 attempts left.

Check the photo requirements.

Subject eyes closed

After your tenth attempt you will need to start again and re-enter the CAPTCHA security check.

Reference number: 20161206-81

Filename: Untitled.jpg

If you wish to contact us about the photo, you must provide us with the reference number given above.

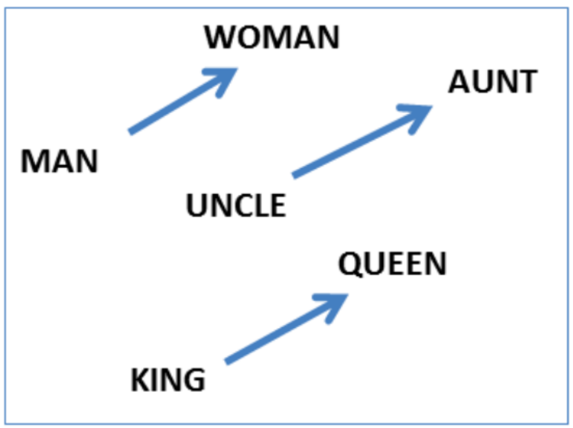
Please print this information for your records.

A screenshot of New Zealand man Richard Lee's passport photo rejection notice, supplied to Reuters December 7, 2016. Richard Lee/Handout via REUTERS

https://www.reuters.com/article/us-newzealand-passport-error/new-zealand-passport-robot-tells-applicant-of-asian-descent-to-open-eyes-idUSKBN13W0RL

Word Embeddings can be Dreadfully Sexist [nips16, **reported by NPR, MIT tech review**]

- $v_{man} - v_{woman} + v_{uncle} \sim v_{aunt}$



he: ___	she: ___
uncle	aunt
lion	
surgeon	
architect	
beer	
professor	

We use Google w2v embedding trained from the news

Project occupations in gender direction

Extreme *he* occupations

1. maestro	2. skipper	3. protege
4. philosopher	5. captain	6. architect
7. financier	8. warrior	9. broadcaster
10. magician	11. figher pilot	12. boss

she

he

Extreme *she* occupations

1. homemaker	2. nurse	3. receptionist
4. librarian	5. socialite	6. hairdresser
7. nanny	8. bookkeeper	9. stylist
10. housekeeper	11. interior designer	12. guidance counselor

Quora | Ask or Search Quora | Ask Question | Read

Natural Language Processing · Data · Machine Learning · Seeking Question

Where can I find some pre-trained word vectors for natural language processing/understanding?

Answer · Request · Follow 36 · Comment 1 · Share · Downvote

4 Answers

Volkan Cirik, worked on word {vectors|embedding}
Written May 9, 2014

You can use Koc AI-Lab's repository ai-ku/wvec to induce word embeddings.
That link should point you out pre-trained word embeddings for WSJ corpus.
I already have pre-trained embeddings for RCV1 and Wikipedia corpora for different dimensions. They are not available for public yet, but I can provide word embeddings if you want.

Other word embeddings are :
Mikolov's Works : word2vec (the code is pretty fast)
Turian's Work : Word representations for NLP
Dhillon's Work : Eigenwords
Huang's Work : Improving Word Representations Via Global Context And Multiple Word Prototypes

Google

word2vec resume

Scholar

About 93 results (0.02 sec)

Articles

Case law

My library

Any time

Since 2016

Since 2015

Since 2012

Custom range...

Sort by relevance

Sort by date

include patents

include citations

Create alert

Machine Learned **Resume**-Job Matching Solution

Y Lin, H Lei, PC Addo, X Li - arXiv preprint arXiv:1607.07657, 2016 - arxiv.org

... We use LDA to classify **resumes** into 32 and 64 topics respectively. ... each Chinese phrase as a word and each list of phrases as a sentence, after **word2vec** training, each ... In this paper, we have considered the **resume**-job matching problem and pro- posed a solution by using ...

Cite Save

[PDF] SKILL: A System for Skill Identification and Normalization.

M Zhao, F Javed, F Jacob, M McNair - AAAI, 2015 - pdfs.semanticscholar.org

... ThiS dictionary capacitateS 90% of noiSe exhibited in **reSume** SkillS SectionS. ... iS initiated firSt for the input queY ry (aka, Seed Skill phraSeS from **reSumeS**) for proper ... implement and produce highly precise and relevant skills recognition system, we utilize **word2vec** (Mikolov et ...

Cited by 4 Related articles All 3 versions Cite Save More

Word2Vec vs DBnary ou comment (ré) concilier représentations distribuées et réseaux lexico-sémantiques? Le cas de l'évaluation en traduction automatique

C Servan, Z Elloumi, H Blanchon, L Besacier - TALN 2016, 2016 - hal.archives-ouvertes.fr

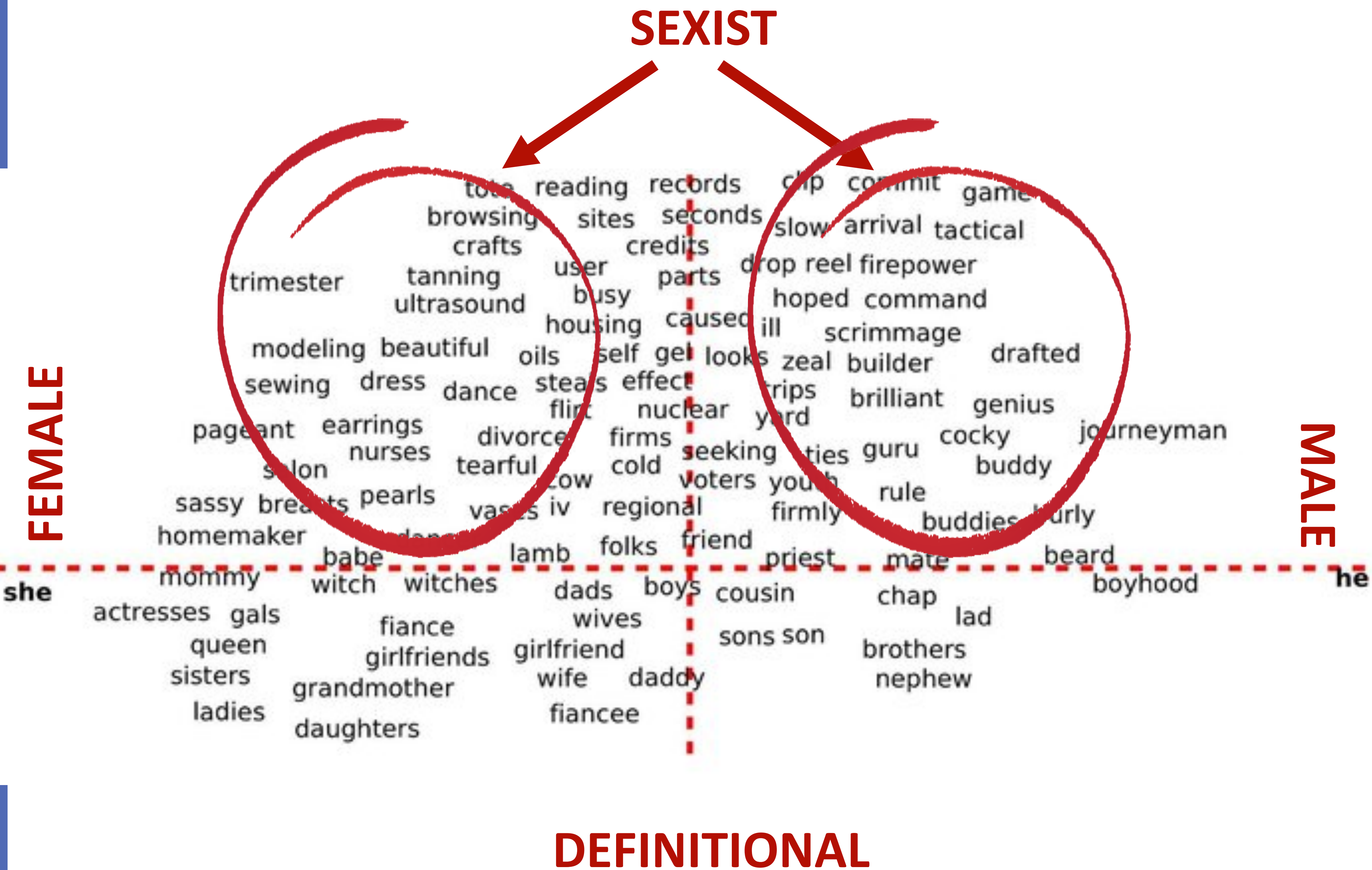
... Page 2. **Word2Vec** vs DBnary ou comment (ré)concilier représentations ... **RÉSUMÉ** Cet article présente une approche associant réseaux lexico-sémantiques et représentations distribuées de mots appliquée à l'évaluation de la traduction automatique. ...

Cite Save

Macau: Large-scale skill sense disambiguation in the online recruitment domain

Q Luo, M Zhao, F Javed, F Jacob - Big Data (Big Data), 2015 ..., 2015 - ieeexplore.ieee.org

... Contexts are extracted from either skill section(s) of **resumes** or requirement section(s) of job postings. We used a popular tool **word2vec** [12] with parameter



(related [Schmidt '15])

~~SEXIST~~

tote
browsing
tanning
scrimmage
dress
sewing
brilliant
nurse
cocky
genius
homemaker

FEMALE

MALE

DEFINITIONAL

(related [Schmidt '15])

Debiasing algorithm (Soft version)

Find a linear transformation T of the gender-neutral words to reduce the gender component while not moving the words too much.

$W =$ matrix of all word vectors.

$N =$ matrix of neutral word vectors.

$$\min_T ||\underbrace{(TW)^T(TW) - W^TW}_{\text{don't move too much}}||_F^2 + \lambda||\underbrace{(TN)^T(TB)}_{\text{minimize gender component}}||_F^2$$

Human Bias in Structured Prediction Models

[EMNLP 17*] w/ Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez

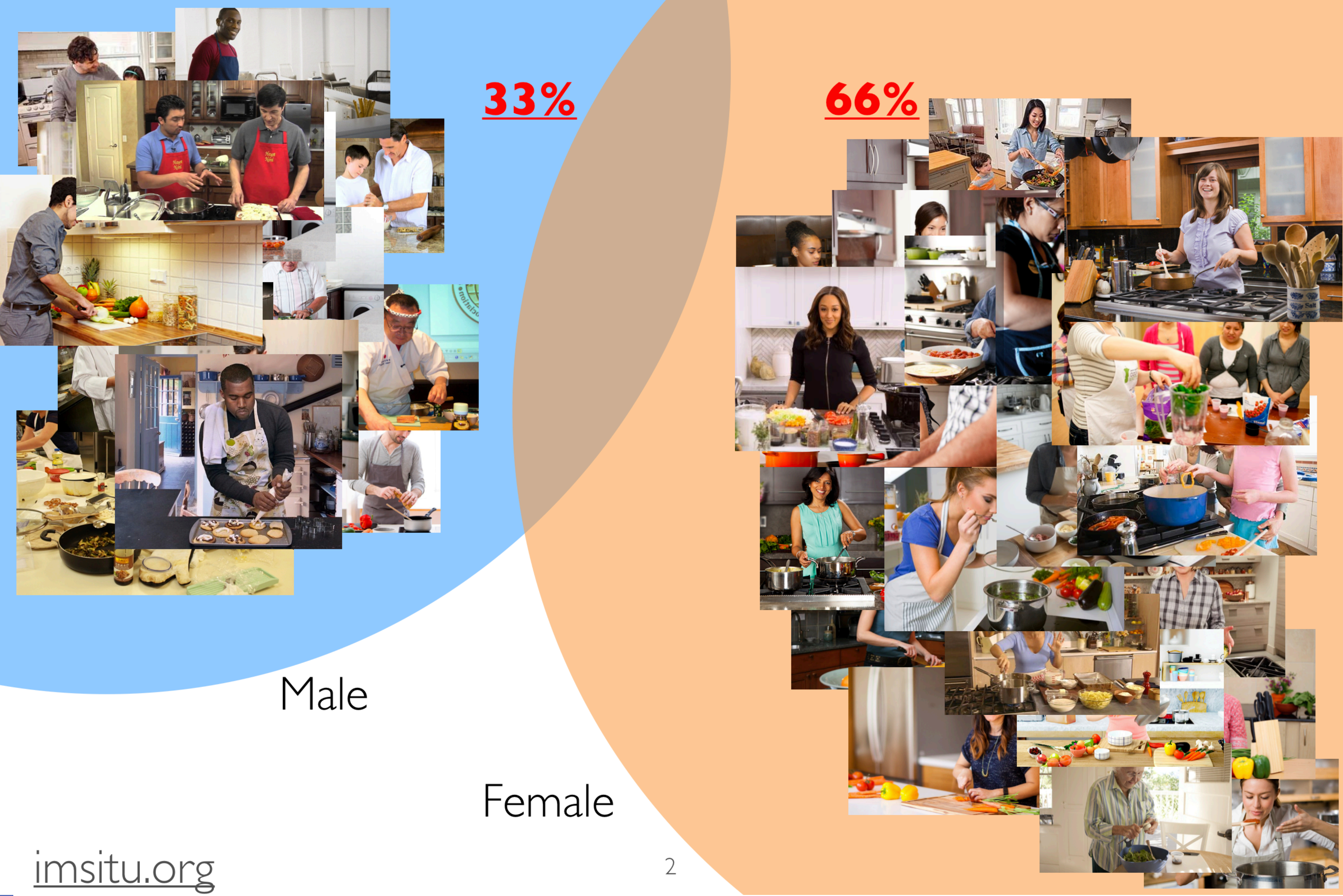
What's the agent for this image?

Cooking	
Role	Object
agent	?
food	vegetable
container	bowl
tool	knife
place	kitchen

An example from a vSRL (visual Semantic Role Labeling) system

*Best Long Paper Award at EMNLP 17

Dataset Gender Bias



Model Bias After Training

16%

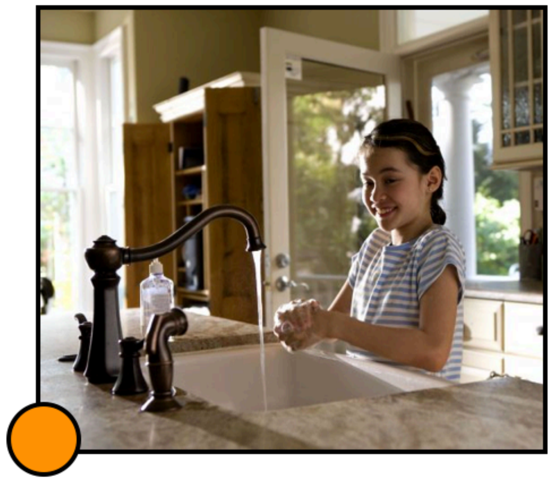
84%

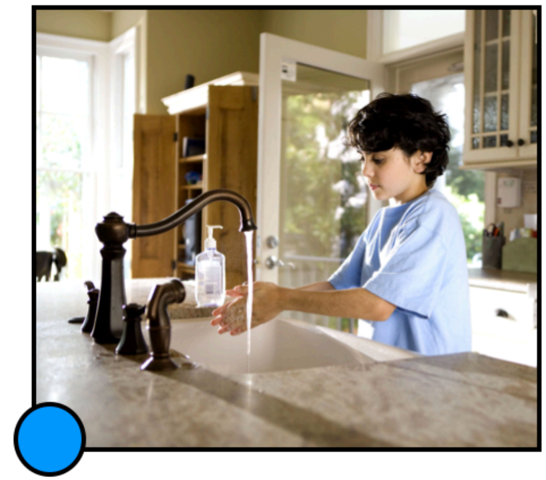
Male

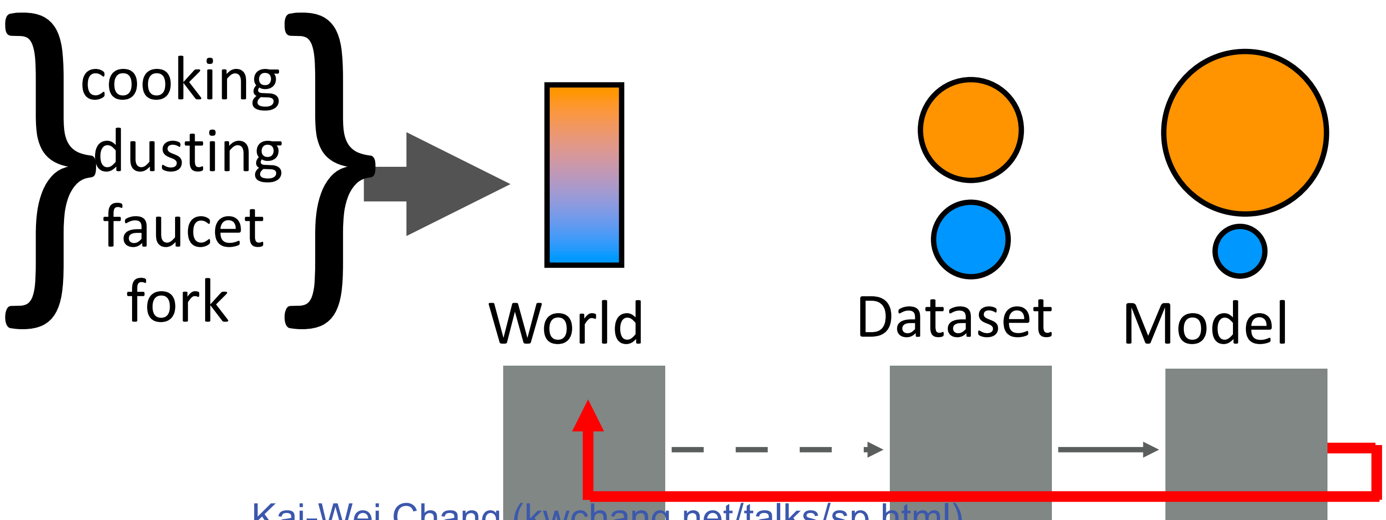
Female

imsitu.org

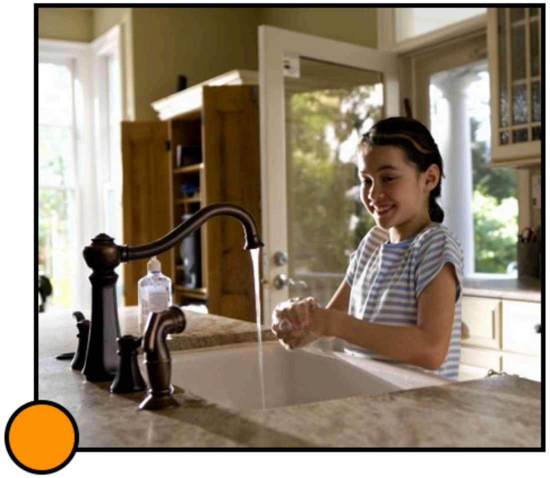
Algorithmic Bias in Grounded Setting



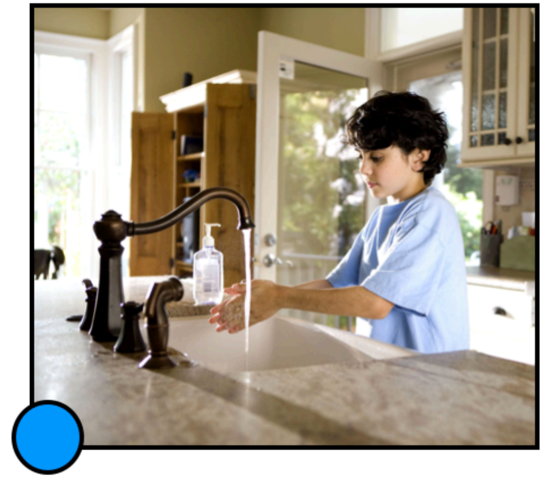




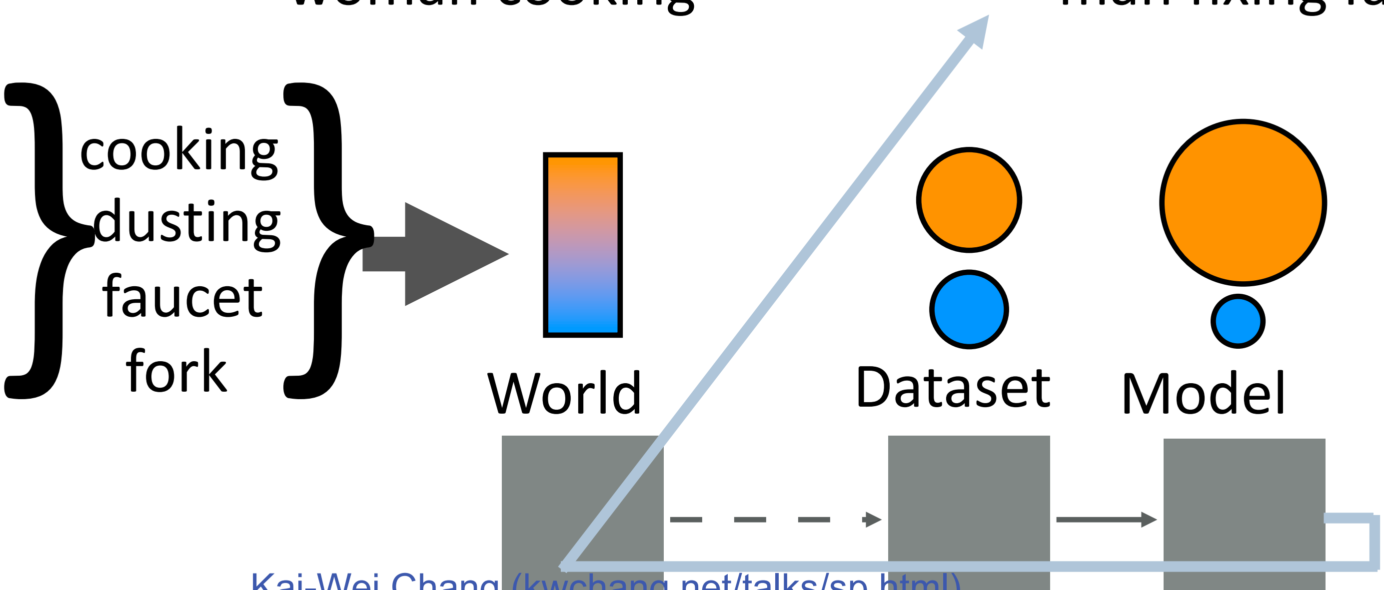
Algorithmic Bias in Grounded Setting



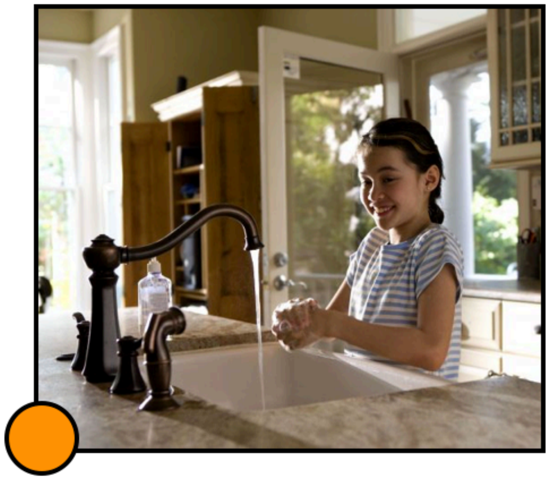
woman cooking

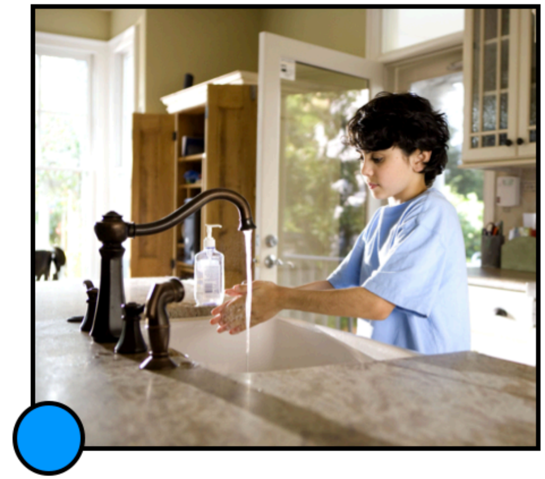


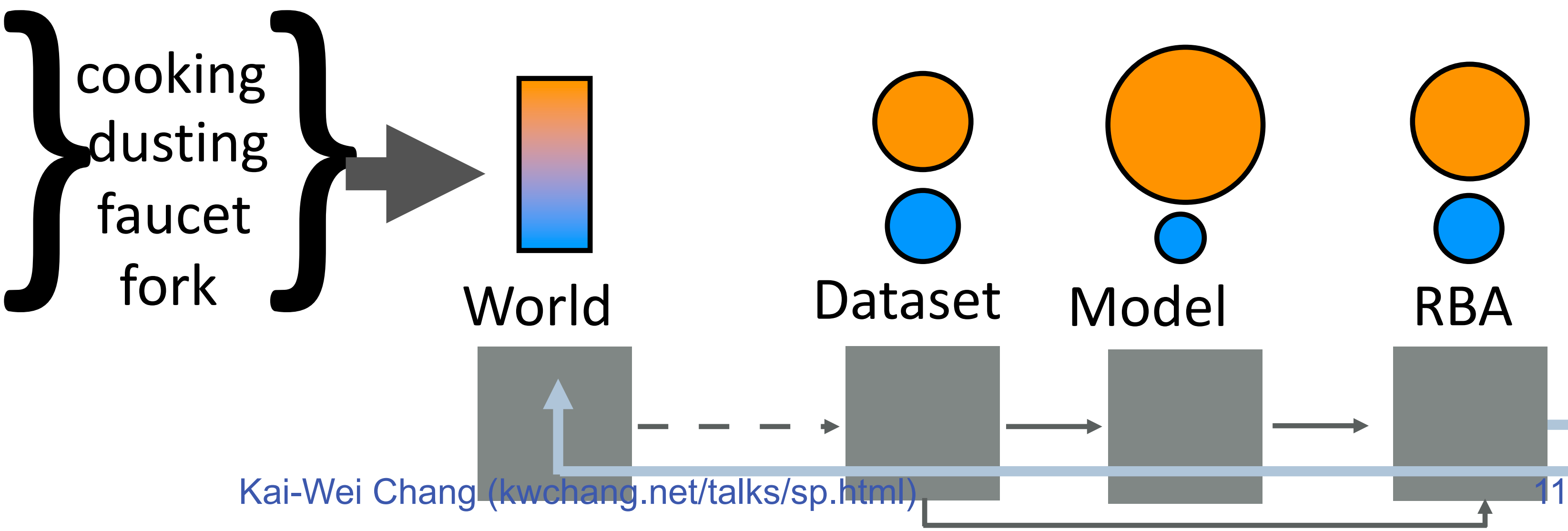
man fixing faucet



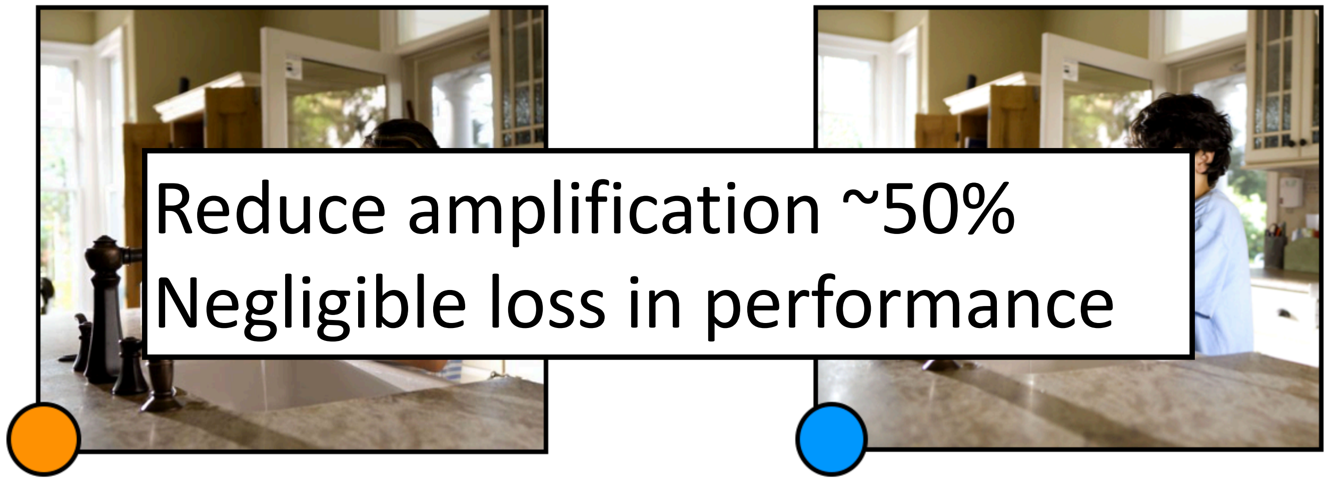
Algorithmic Bias in Grounded Setting

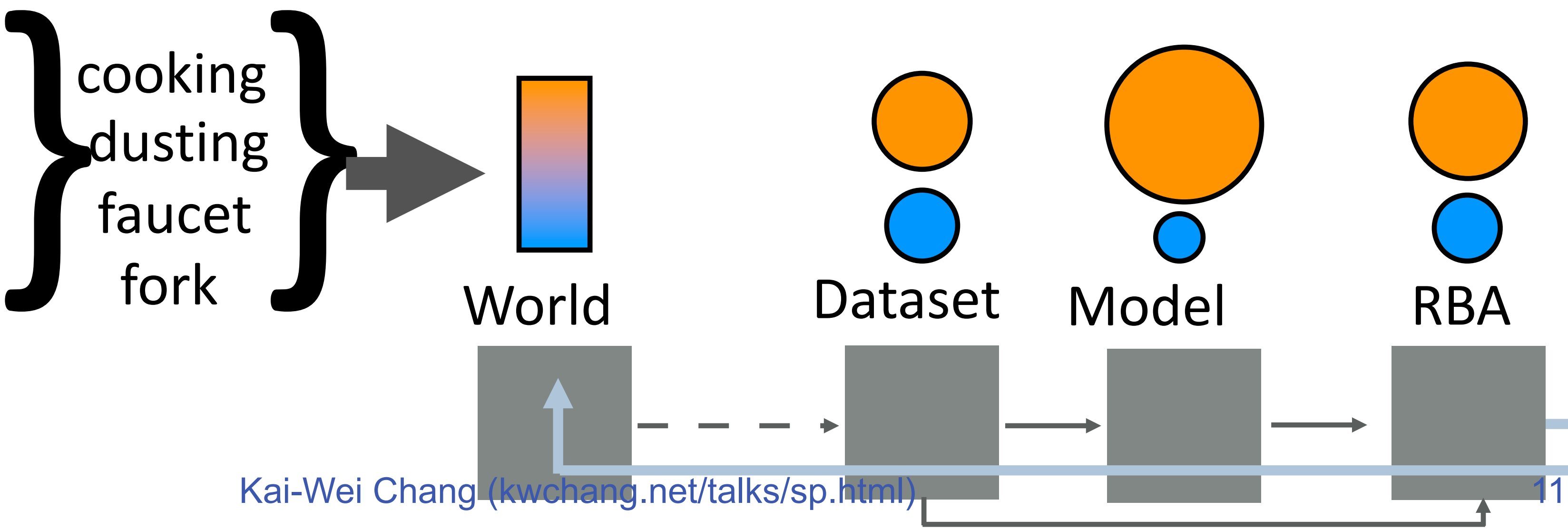




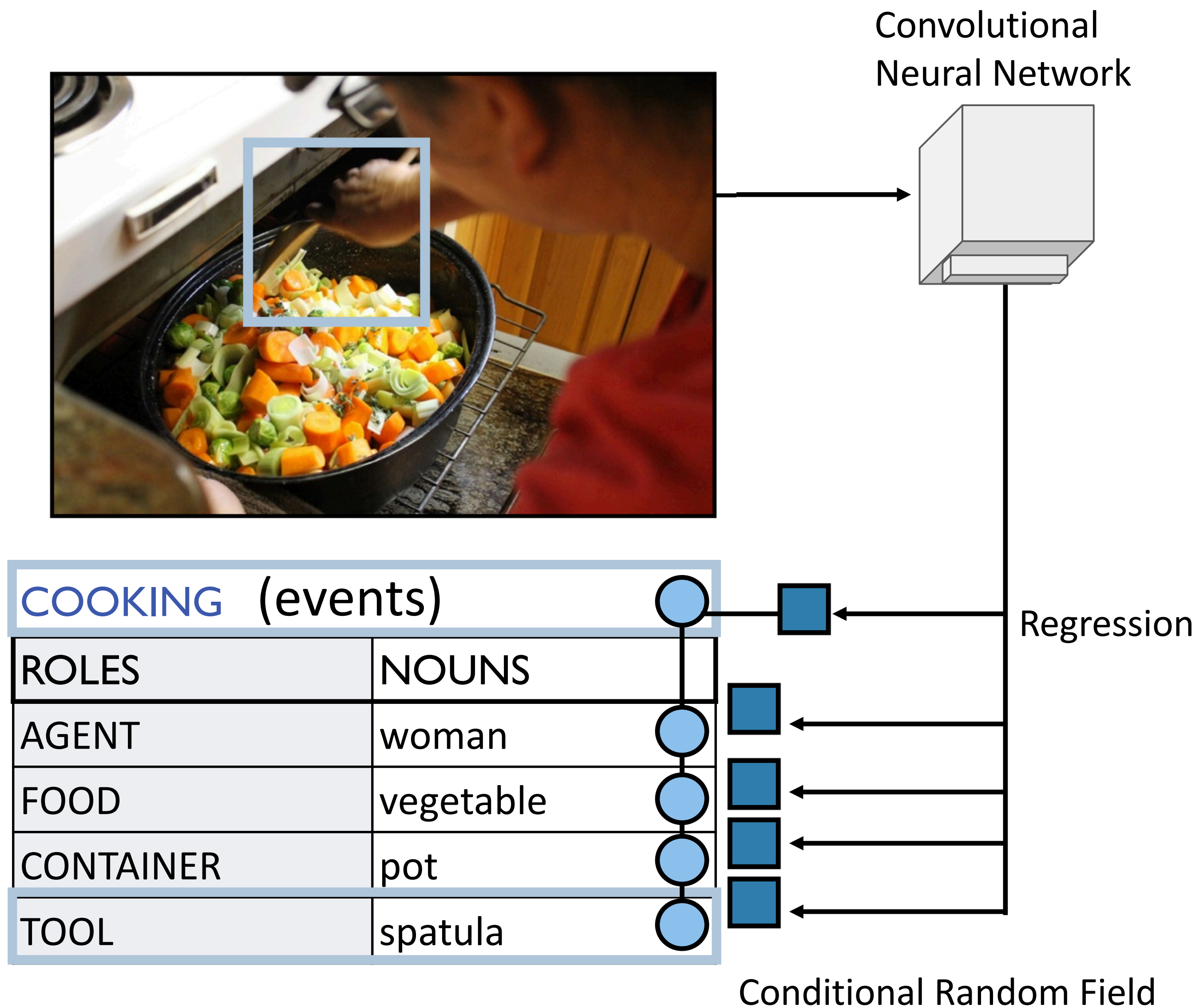


Algorithmic Bias in Grounded Setting



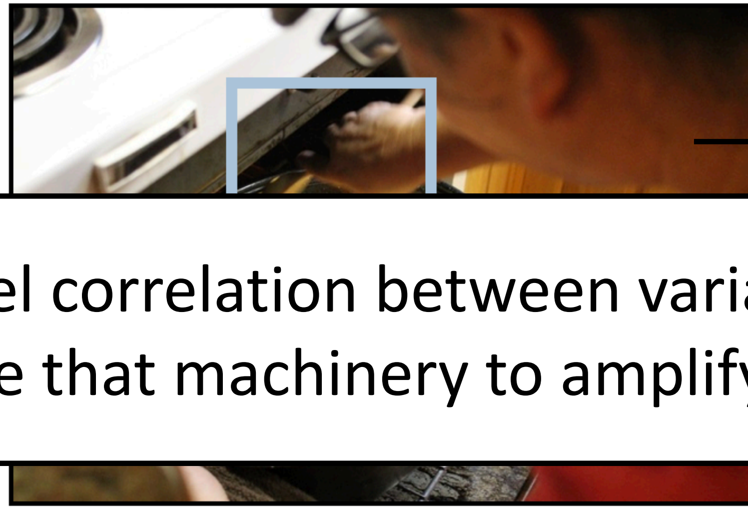


imSitu Visual Semantic Role Labeling (vSRL)



Yatskar et al. CVPR ’16, Yang et al. NAACL ’16, Gupta and Malik arXiv ’16

imSitu Visual Semantic Role Labeling (vSRL)



Convolutional Neural Network

Need to model correlation between variables
Model can use that machinery to amplify gender bias

COOKING (events)

ROLES	NOUNS
AGENT	woman
FOOD	vegetable
CONTAINER	pot
TOOL	spatula

Regression

Conditional Random Field

Yatskar et al. CVPR ’16, Yang et al. NAACL ’16, Gupta and Malik arXiv ’16

COCO Multi-Label Classification (MLC)

Internet

a <u>woman</u> is smiling in a kitchen near a pizza on a stove

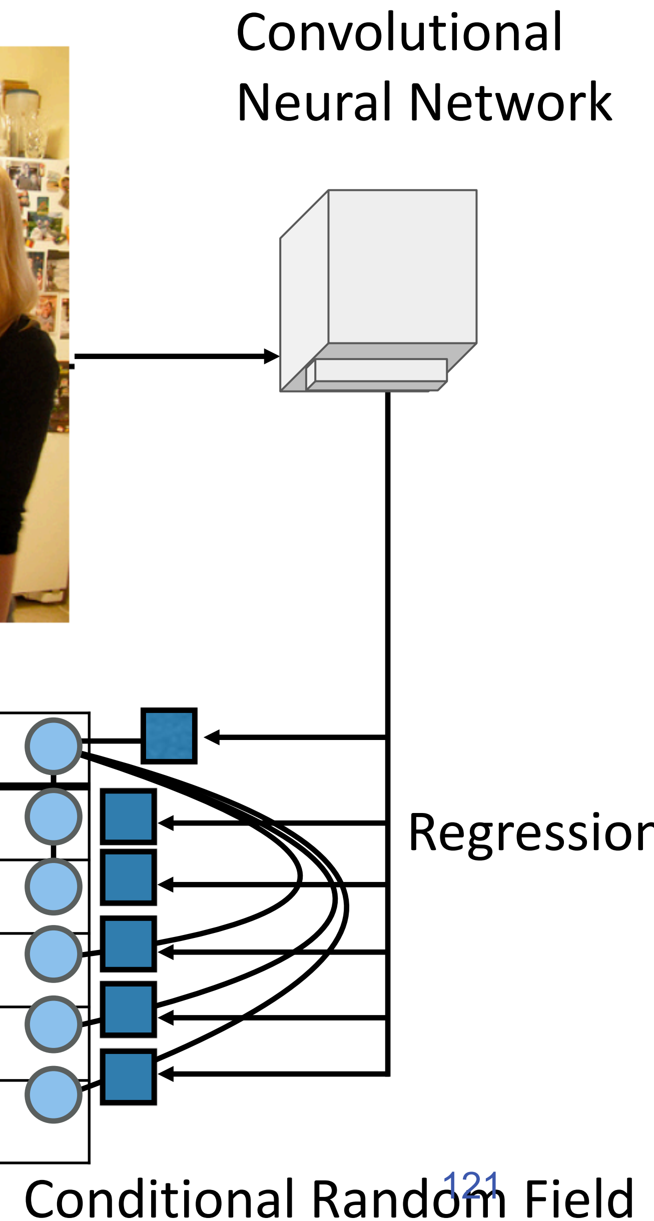
WOMAN	(objects)
PIZZA	yes
ZEBRA	no
FRIDGE	yes
CAR	no
...	...

Caption Inferred Label

COCO Objects

COCO Multi-Label Classification (MLC)

WOMAN	
PIZZA	yes
ZEBRA	no
FRIDGE	yes
CAR	no
…	…



Defining Dataset Bias (events)

Training Gender Ratio (◆ verb)

Training Set

◆ cooking

● woman

● man

COOKING	
ROLES	NOUNS
AGENT	woman
FOOD	stir-fry

COOKING	
ROLES	NOUNS
AGENT	man
FOOD	noodle

$$\frac{\#(\text{◆ cooking},\ \text{● man})}{\#(\text{◆ cooking},\ \text{● man}) + \#(\text{◆ cooking},\ \text{● woman})} = 1/3$$

Defining Dataset Bias (objects)

Training Gender Ratio (▲ noun)

Training Set

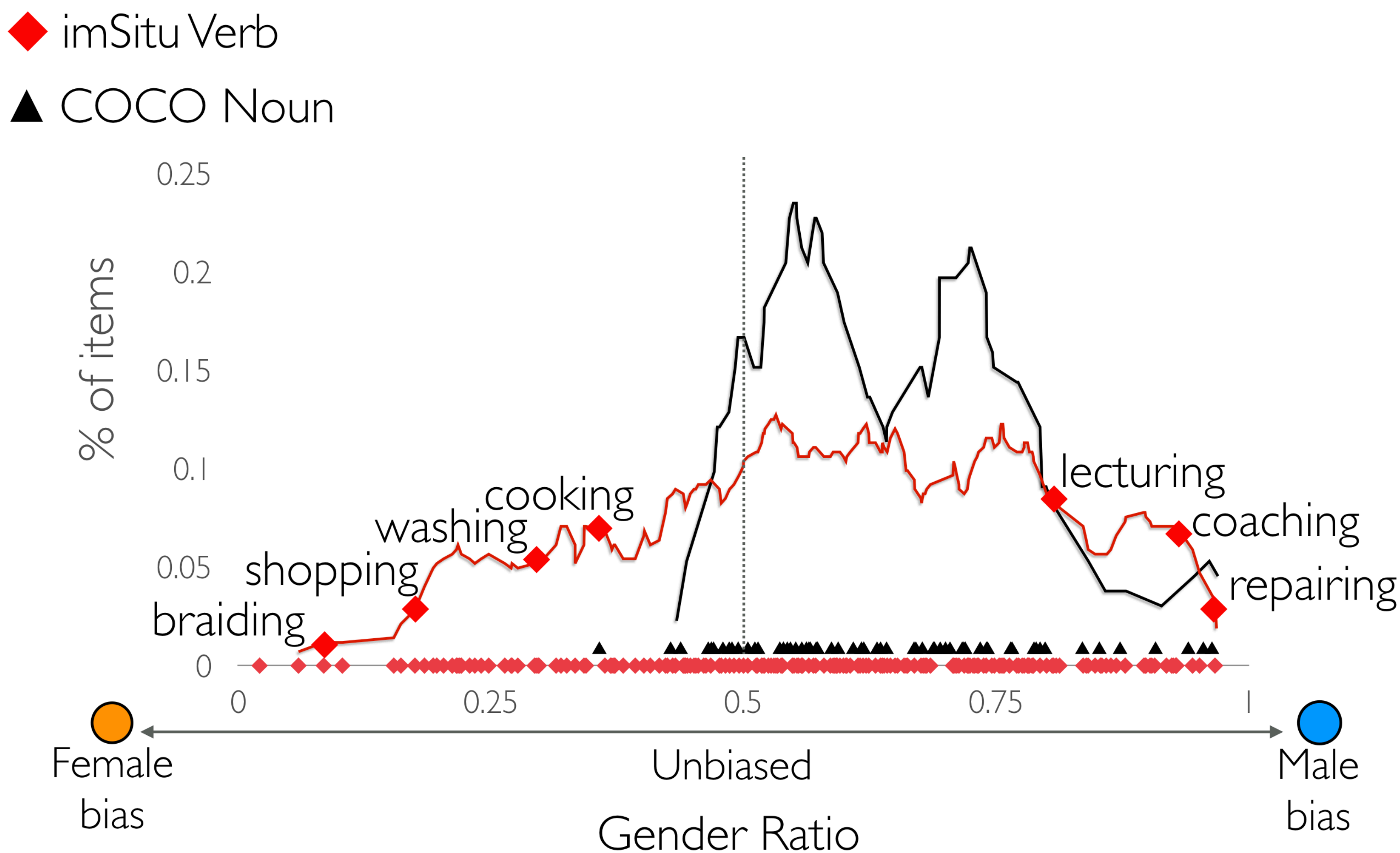
- ▲ snowboard
- woman
- man

MAN	
▲ snowboard	yes
refrigerator	no
bowl	no

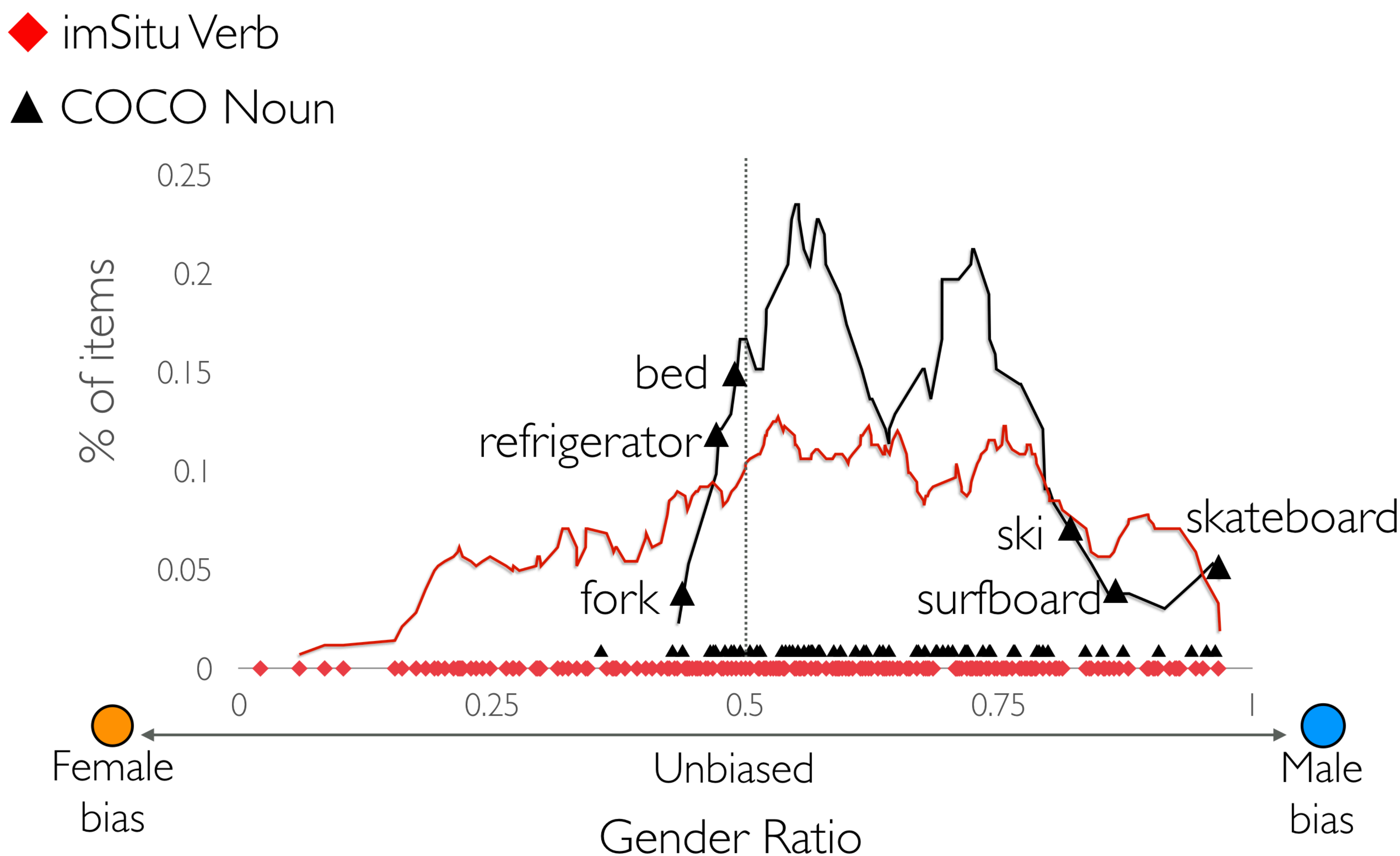
WOMAN	
▲ snowboard	yes
refrigerator	no
bowl	no

$$\frac{\#(\blacktriangle\ \text{snowboard},\ \text{man})}{\#(\blacktriangle\ \text{snowboard},\ \text{man}) + \#(\blacktriangle\ \text{snowboard},\ \text{woman})} = 2/3$$

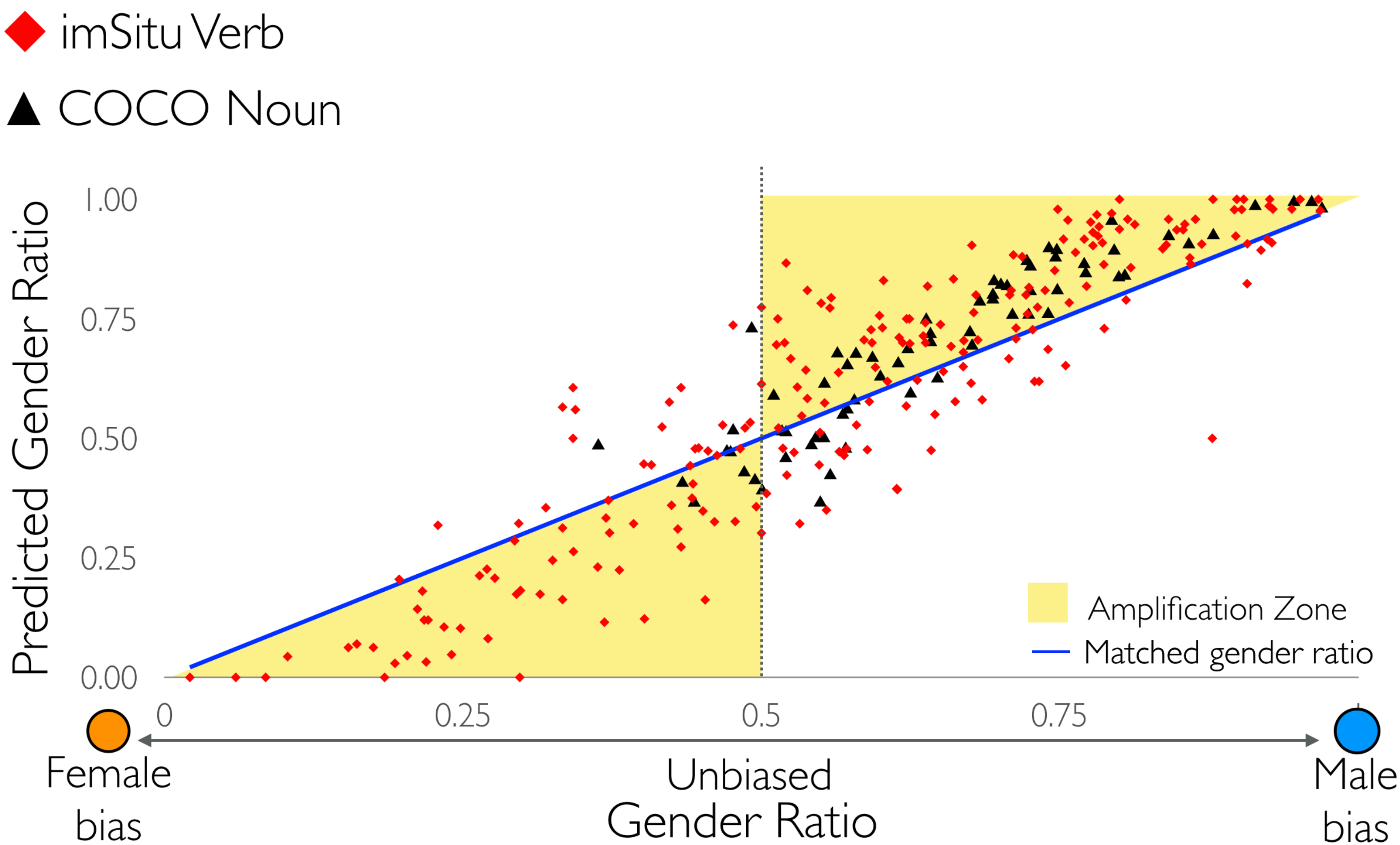
Gender Dataset Bias



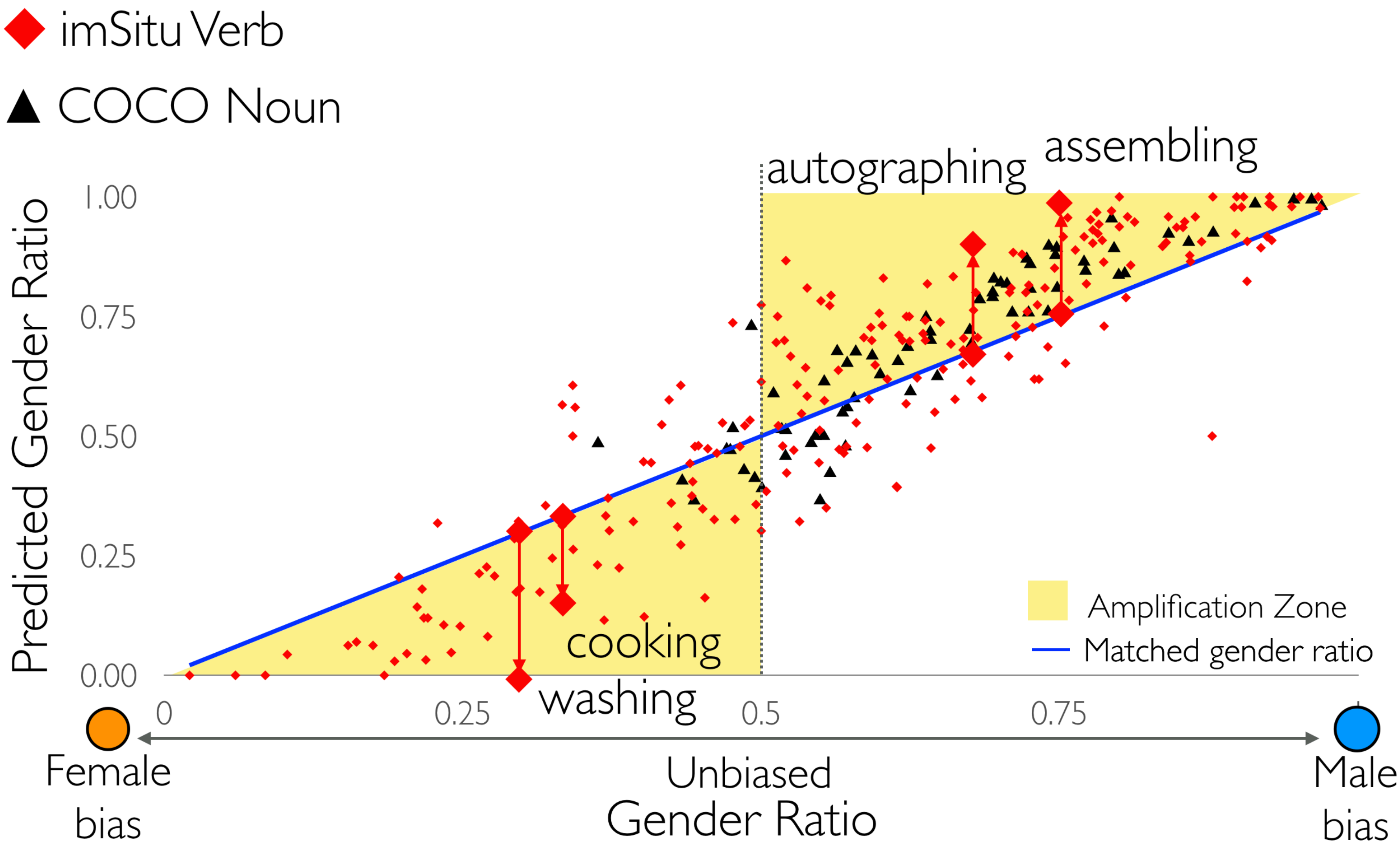
Gender Dataset Bias



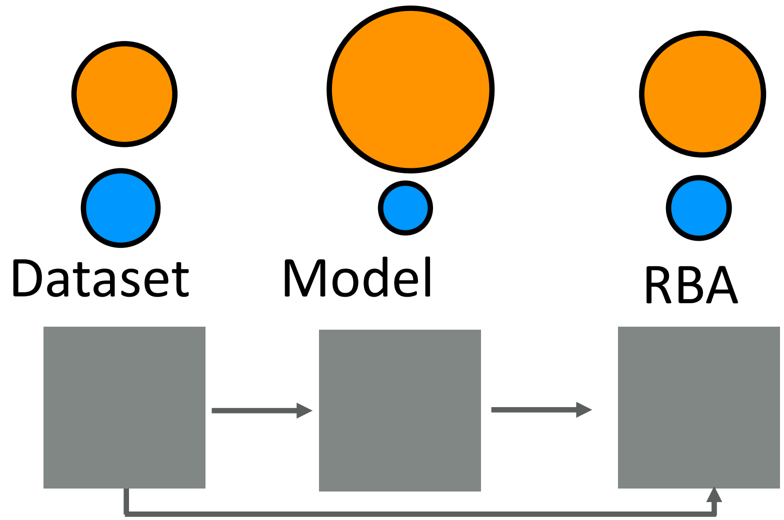
Model Bias Amplification



Model Bias Amplification



Reducing Bias Amplification (RBA)



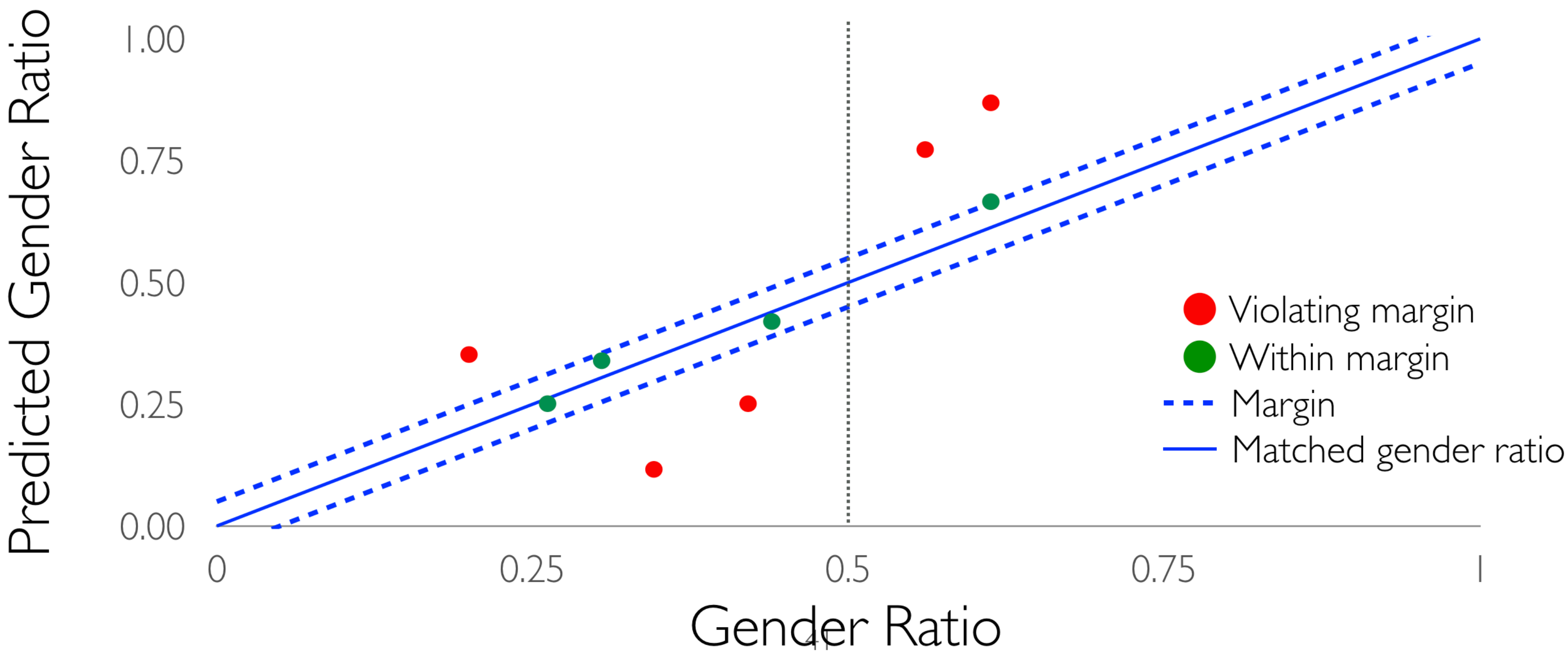
- Corpus level constraints on model output (ILP)
  - Doesn't require model retraining
- Reuse model inference through Lagrangian relaxation
  - Can be applied to any structured model

Reducing Bias Amplification (RBA)

Integer Linear Program

$$\sum_{i} \max_{y_i} \; s(y_i, \text{image})$$

$$\forall \text{ points} \quad \left| \text{Training Ratio} - \underbrace{\text{Predicted Ratio}}_{f(y_1 \ldots y_n)} \right| \;<=\; \text{margin}$$

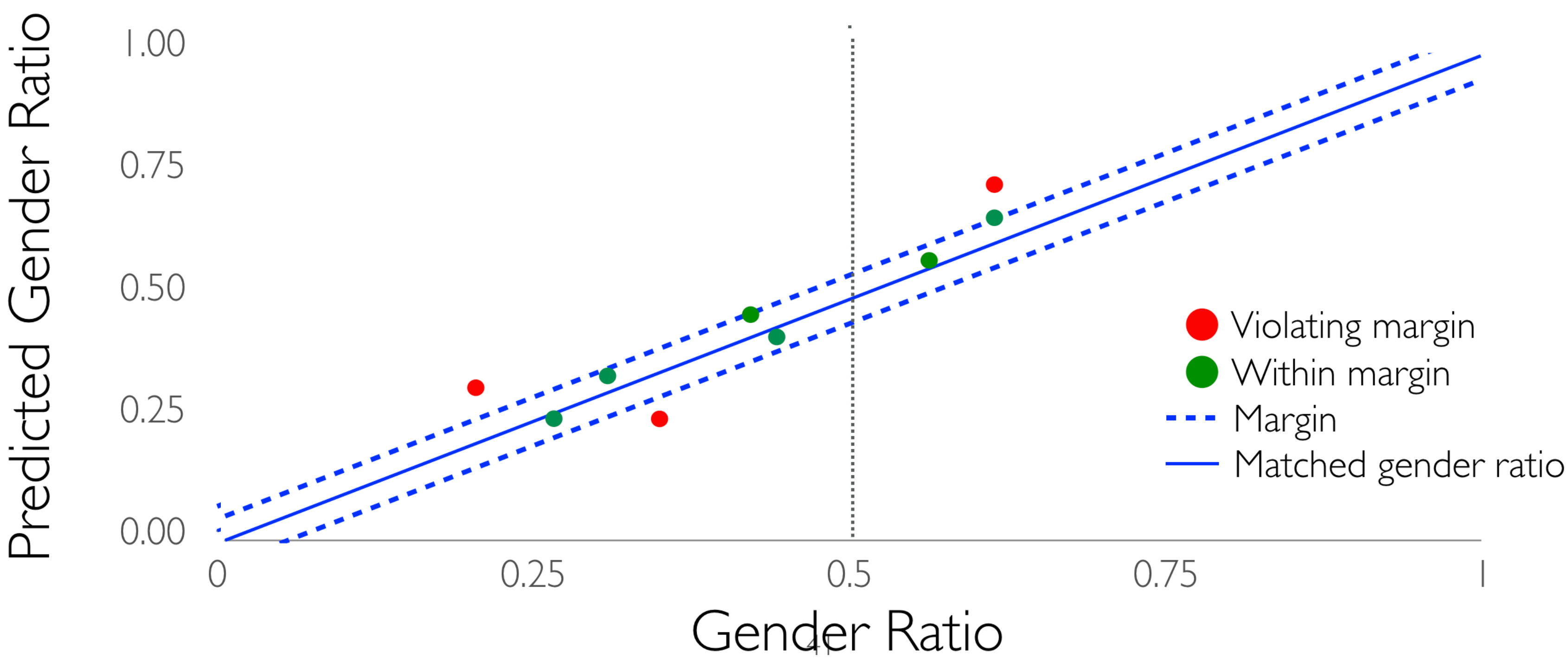


Reducing Bias Amplification (RBA)

Integer Linear Program

$$\sum_{i} \max_{y_i} \; s(y_i, \text{image})$$

$$\forall \text{ points} \quad \left| \text{Training Ratio} - \underset{f(y_1 \ldots y_n)}{\text{Predicted Ratio}} \right| <= \text{margin}$$

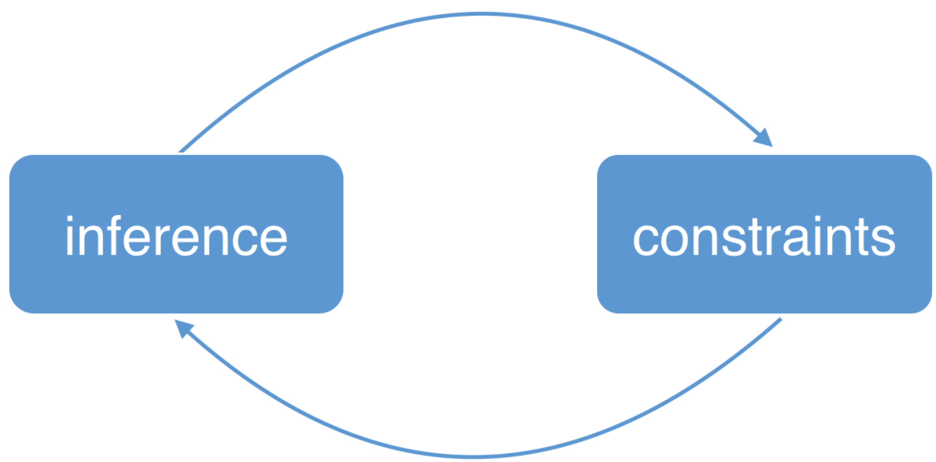


Reducing Bias Amplification (RBA)

$$\sum_{i} \max_{y_i} \; s(y_i, \text{image})$$

$$\forall \text{ points} \quad \left| \text{Training Ratio} - \underset{f(y_1 \ldots y_n)}{\text{Predicted Ratio}} \right| <= \text{margin}$$

$$\boxed{\text{Lagrangian Relaxation}}$$



Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

Reducing Bias Amplification (RBA)

$$\sum_{i} \max_{y_i} s(y_i, \text{image})$$

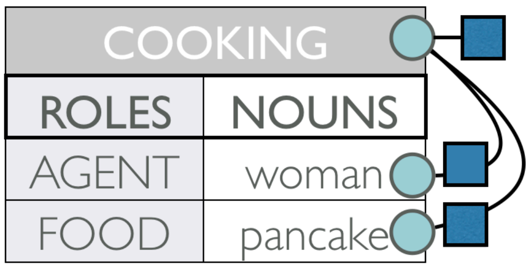
$\forall$ points $|$ Training Ratio - Predicted Ratio $|$ <= margin

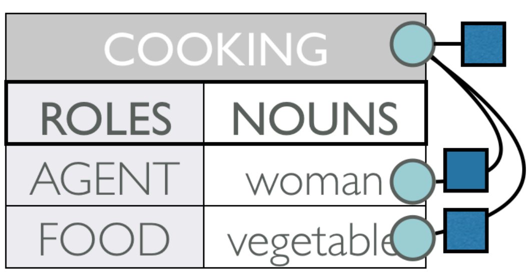
$$\max_{\{y^i\}\in\{Y^i\}} \sum_i f_\theta(y^i, i), \quad \text{s.t.} \quad A\sum_i y^i - b \le 0$$

Lagrangian : $\sum_i f_\theta(y^i) - \sum_{j=1}^{l} \lambda_j (A_j \sum_i y^i - b_j) \qquad \lambda_j \geq 0$

Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

Lagrangian Relaxation





$$\sum_i \max_{y_i} \; s(y_i, \text{image})$$

$$\Big| \underset{(1/2)}{\text{Training Ratio}} - \text{Predicted Ratio} \Big| <= \text{margin}$$

- Lagrange Multiplier (λ) Per Constraint

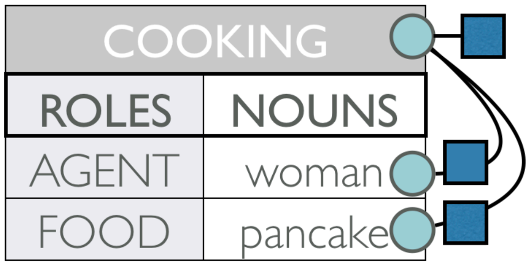
inference

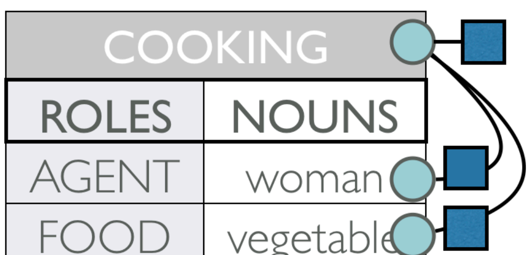
update λ

update potentials

Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

Lagrangian Relaxation





$$\boxed{\sum_i \max_{y_i} s(y_i, \text{image})}$$

$$\left| \text{Training Ratio} - \text{Predicted Ratio} \right| <= \text{margin}$$

(1/2)

- Lagrange Multiplier (λ) Per Constraint

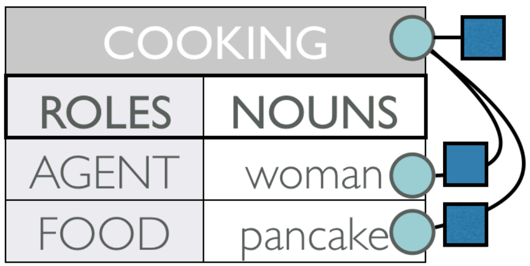
inference

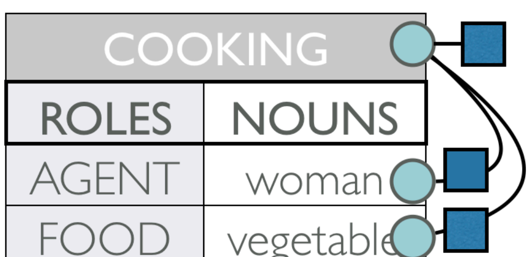
update λ

update potentials

Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

Lagrangian Relaxation





$$\sum_{i} \max_{y_i} \; s(y_i, \text{image})$$

$$\left| \text{Training Ratio} - \text{Predicted Ratio} \right| <= \text{margin}$$

(1/2)

- Lagrange Multiplier (λ) Per Constraint

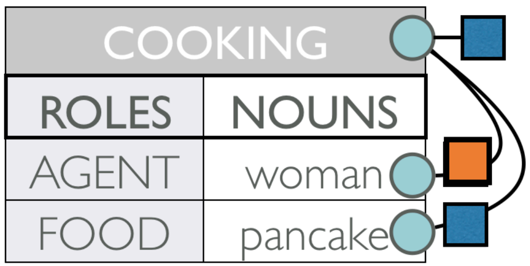
inference

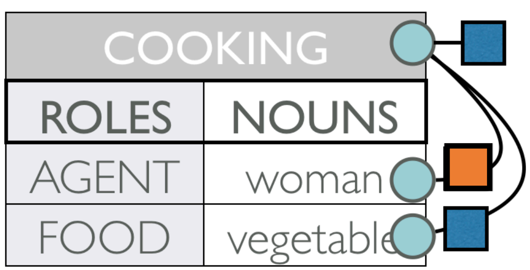
update λ

$$\lambda^{(t)} = \max\left(0, \lambda^{(t-1)} + \sum_i \eta(Ay^{i,(t)} - b)\right)$$

Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

Lagrangian Relaxation





$$\sum_i \max_{y_i} \; s(y_i\,, \text{image})$$

$$\left| \underset{(1/2)}{\text{Training Ratio}} - \text{Predicted Ratio} \right| <= \text{margin}$$

- Lagrange Multiplier (λ) Per Constraint

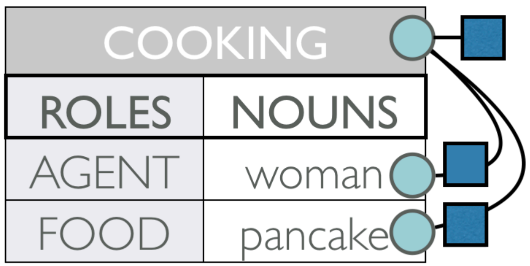
inference

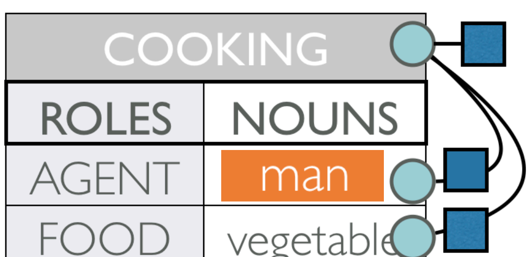
update λ

update potentials

Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

Lagrangian Relaxation





$$\sum_i \max_{y_i} s(y_i, \text{image})$$

$$\left| \underset{(1/2)}{\text{Training Ratio}} - \text{Predicted Ratio} \right| <= \text{margin}$$

- Lagrange Multiplier (λ) Per Constraint

$\boxed{\text{inference}}$

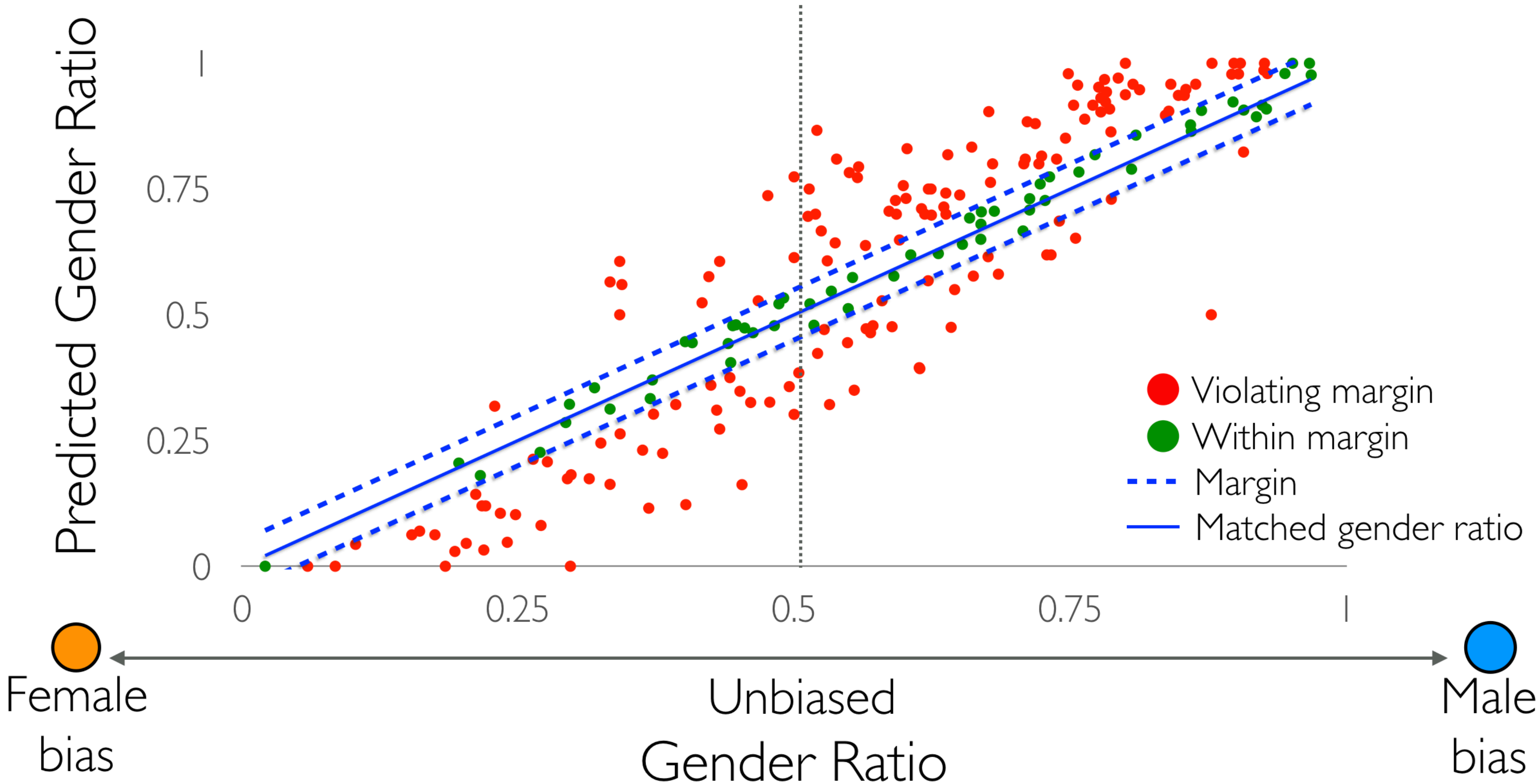
update λ

update potentials

Sontag et al., 2011; Rush and Collins, 2012; Chang and Collins, 2011; Peng et al., 2015, Chang et al., 2013; Dalvi, 2015

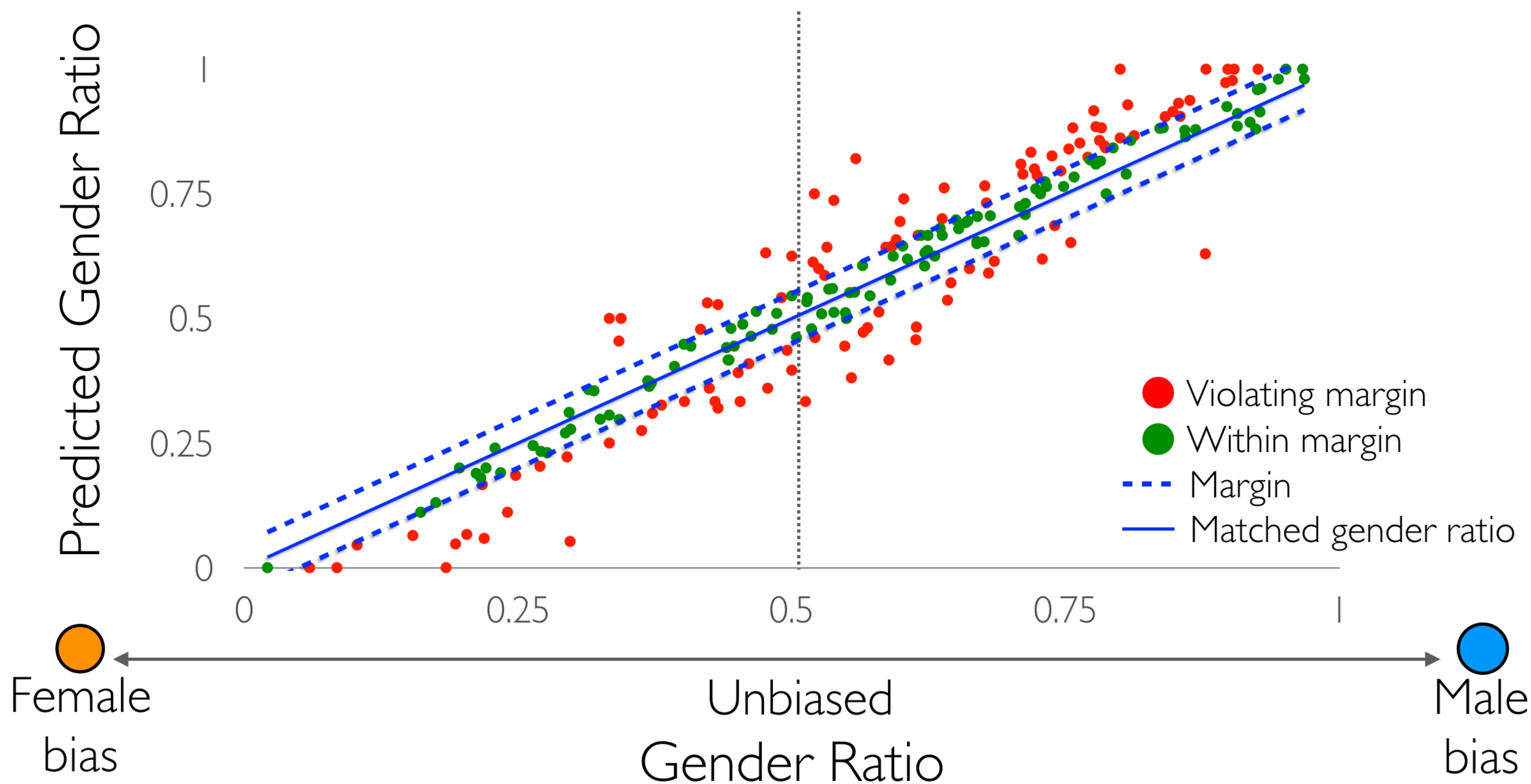
Gender Bias De-amplification in imSitu

imSitu Verb   Violation: 72.6%   .050 |bias↑| 24.07 acc.



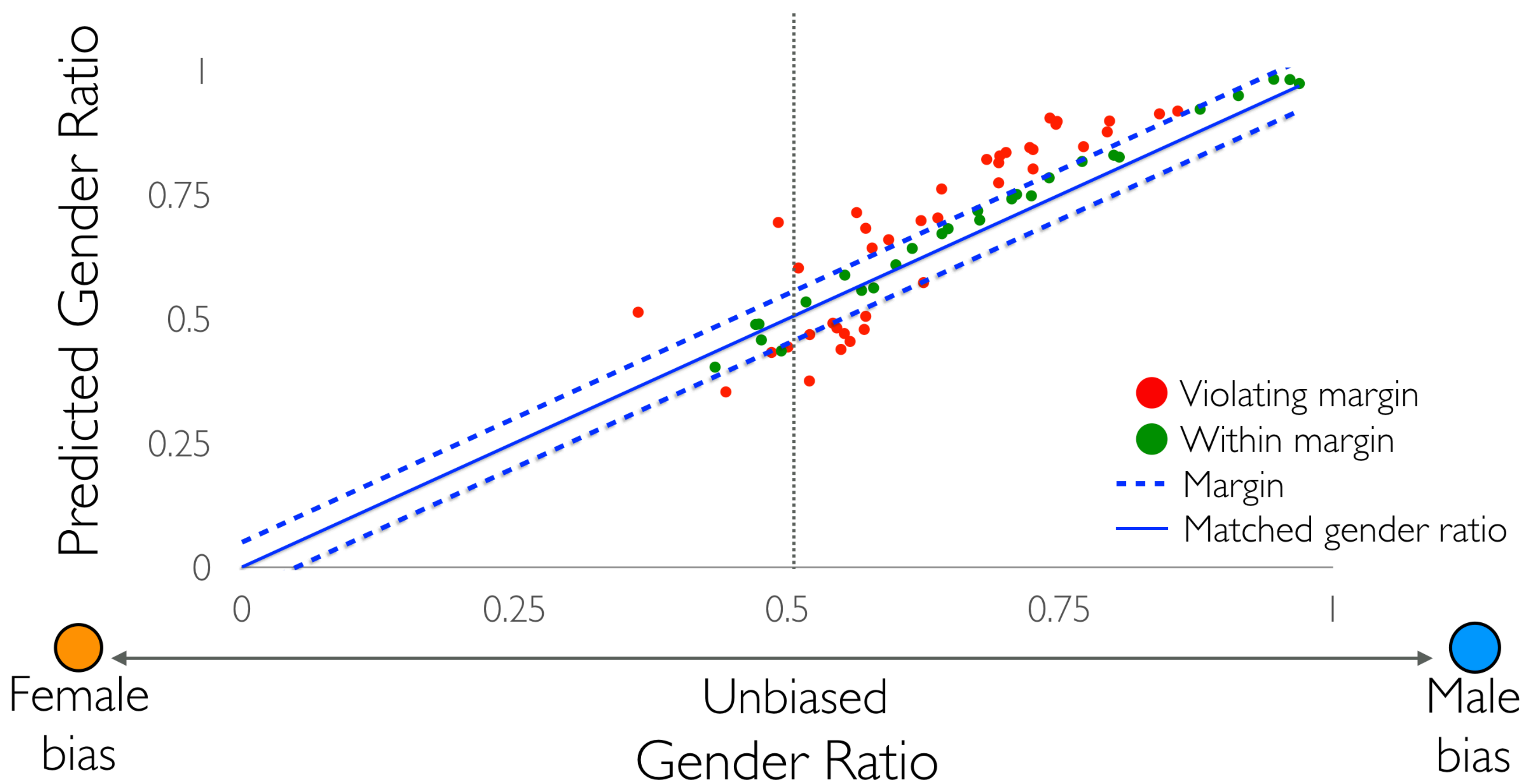
Gender Bias De-amplification in imSitu

imSitu Verb	Violation: 72.6%	.050 |bias↑|	24.07 acc.
w/ RBA	Violation: 50.5%	.024 |bias↑|	23.97 acc.



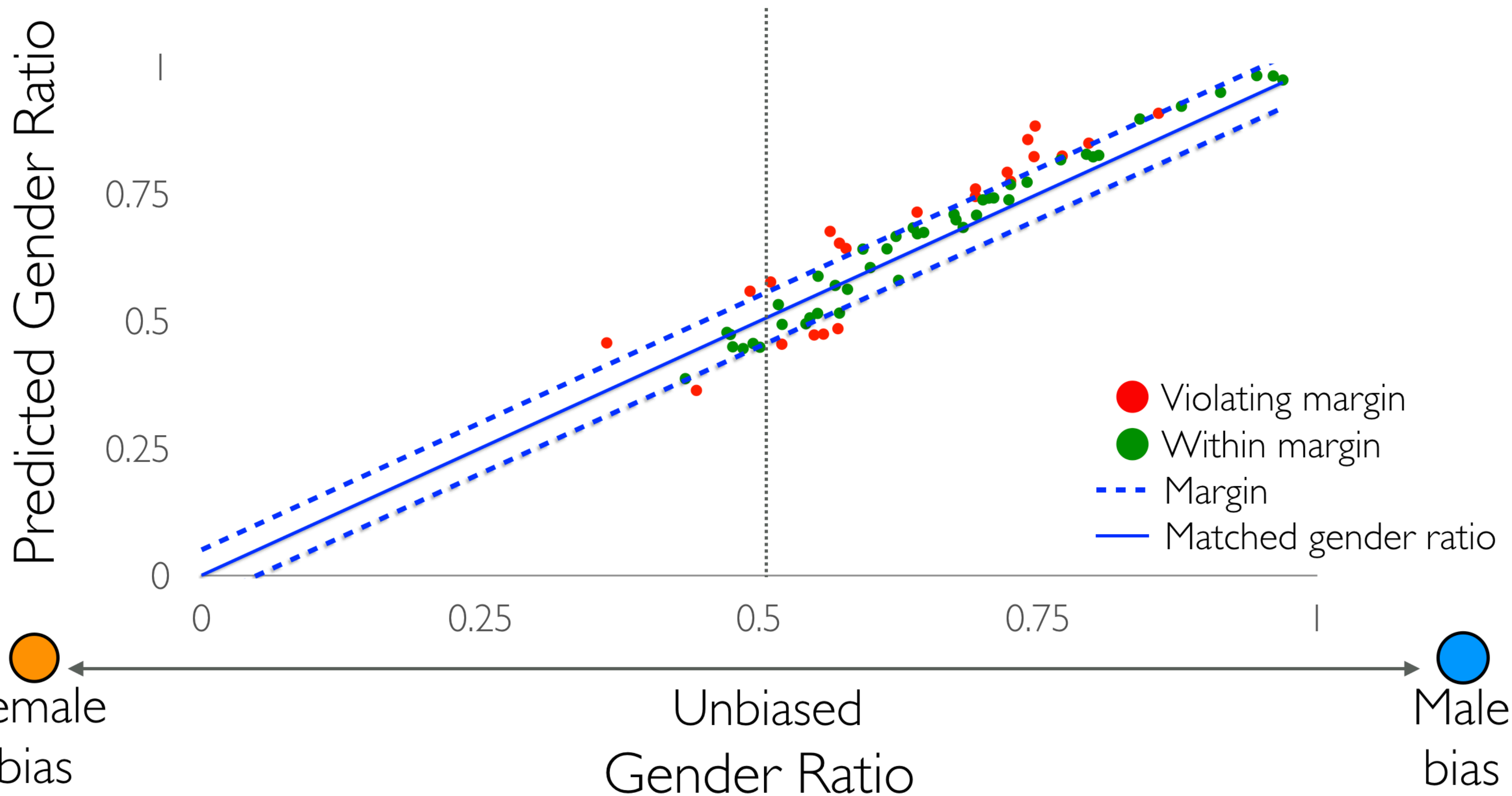
Gender Bias De-amplification in COCO

COCO Noun   Violation: 60.6%   .032 |bias↑| 45.27 mAP

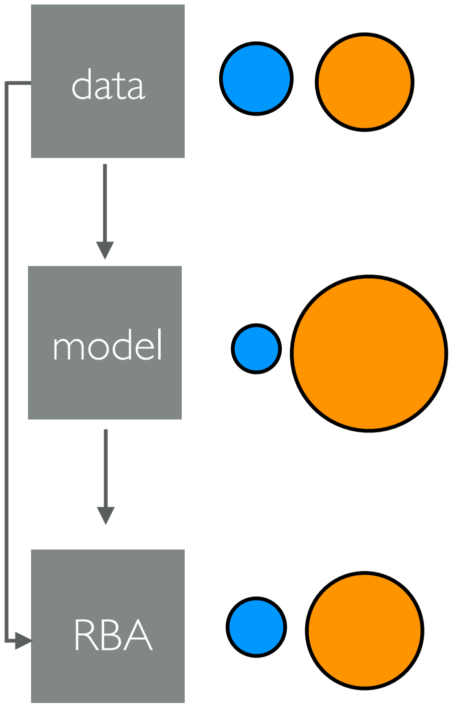


Gender Bias De-amplification in COCO

COCO Noun	Violation:	60.6%	.032	|bias↑|	45.27	mAP
w/ RBA	Violation:	36.4%	.022	|bias↑|	45.19	mAP



Future Work

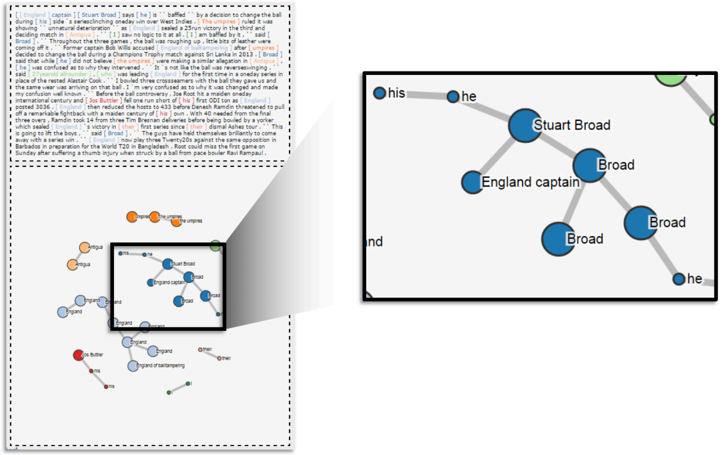


Can existing data be made more balanced?

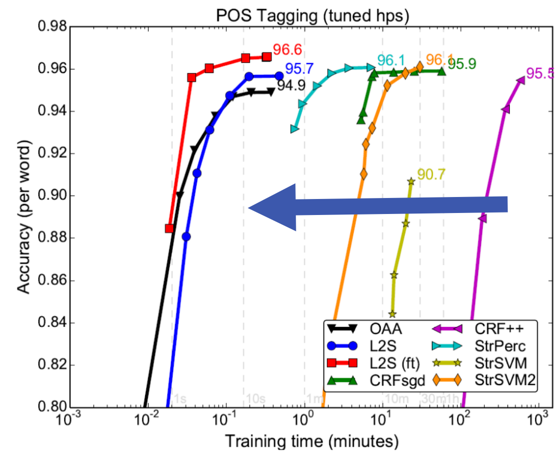
Do all models amplify equally?
i.e. different objectives

Other direct applications?
i.e. co-ref, racial bias

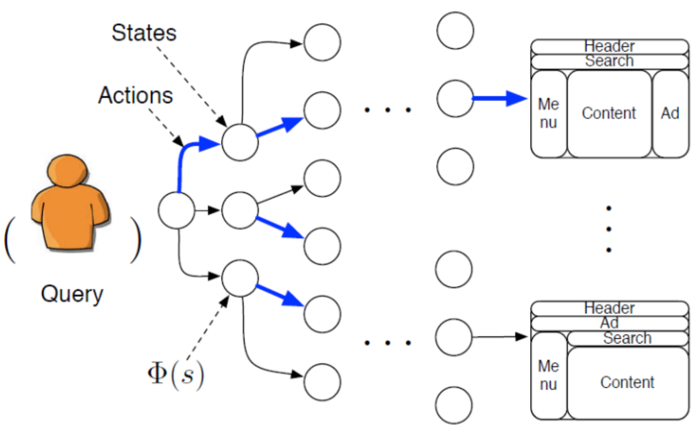
What We Care about



NLP Applications



Training/test/dev speed



Learning signals

activity	cooking
agent	woman
food	vegetable

Fairness (data biases)

Collaborators

- Academic
  - Hal Daume (UMD)
  - Chih-Jen Lin (NTU)
  - Dan Roth (Upenn)
  - Venkatesh Saligrama (BU)
  - Alexander Rush (Harvard)
  - Cho-Jui Hsieh (UCDavis)
  - Vivek Srikumar (Utah)
  - Hongning Wang (UVa)
  - Vicente Ordonez (UVa)
- Industrial
  - John Langford (MSR)
  - Adam Kalai (MSR)
  - Ming-Wei Chang (Google)
  - Scott Yih (AI2)
  - Hoifung Poon (MSR)
  - Mark Yatskar (AI2)
- Students
  - Wasi Ahmad (UCLA)
  - Jieyu Zhao (UCLA)
  - Rizwan Parvez (UCLA)
  - Tianlu Wang (UVa)
  - Ken Arnold (Havard)
  - Shyam Upaphaya (UIUC)

Conclusions

Goal: Practical Structured Prediction Approaches

Tutorials/Workshops:

1. AAAI-16: Learning and Inference in SP Models
2. NAACL15: Hands-on Learning to Search for SP
3. EMNLP 16, 17: workshop SP for NLP

References/Code/Demos:

http://kwchang.net

Illinois-SL: a structured learning package

Vowpal Wabbit: an online learning library