# Instance Weighting for Domain Adaptation in NLP

Jing Jiang & ChengXiang Zhai
University of Illinois at Urbana-Champaign

June 25, 2007

## Domain Adaptation

- Many NLP tasks are cast into classification problems
- Lack of training data in new domains
- Domain adaptation:
  - POS: WSJ → biomedical text
  - NER: news → blog, speech
  - Spam filtering: public email corpus → personal inboxes
- Domain overfitting

| NER Task | Train → Test | F1 |
|---|---|---|
| to find PER, LOC, ORG from news text | NYT → NYT | 0.855 |
| | Reuters → NYT | 0.641 |
| to find gene/protein from biomedical literature | mouse → mouse | 0.541 |
| | fly → mouse | 0.281 |

2

## Existing Work on Domain Adaptation

- Existing work
  - Prior on model parameters [Chelba & Acero 04]
  - Mixture of general and domain-specific distributions [Daumé III & Marcu 06]
  - Analysis of representation [Ben-David et al. 07]
- Our work
  - A fresh instance weighting perspective
  - A framework that incorporates both labeled and unlabeled instances

3

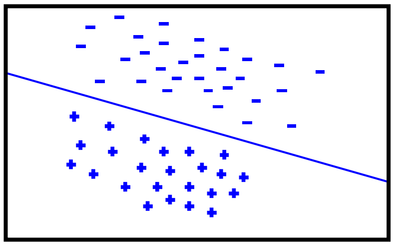
## Outline

- Analysis of domain adaptation
- Instance weighting framework
- Experiments
- Conclusions

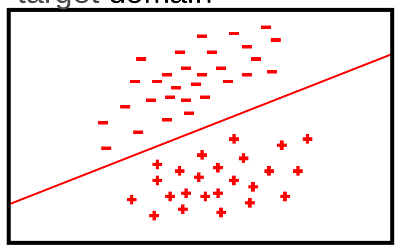
4

## The Need for Domain Adaptation

source domain



target domain

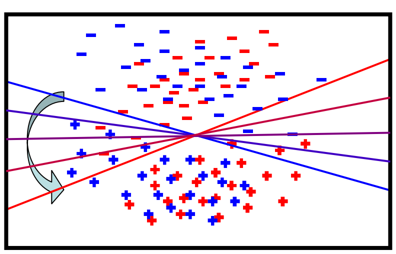


5

## The Need for Domain Adaptation

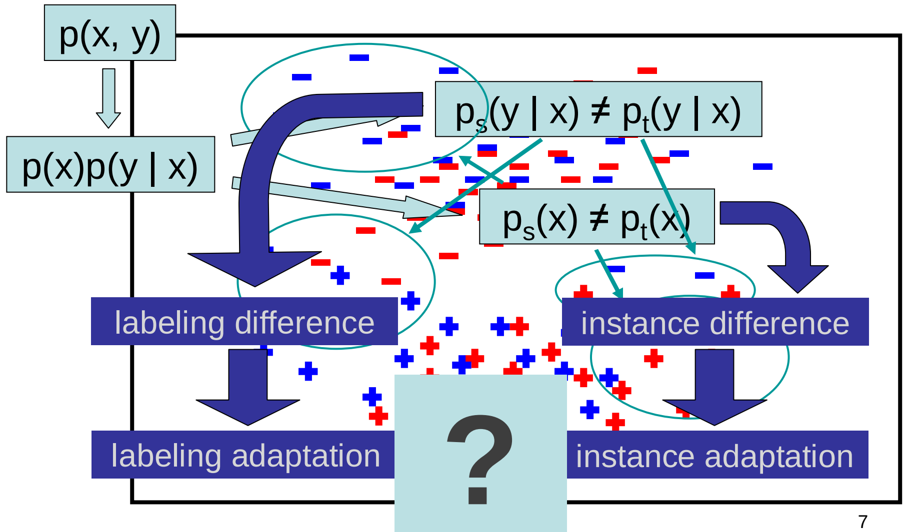
source domain

target domain



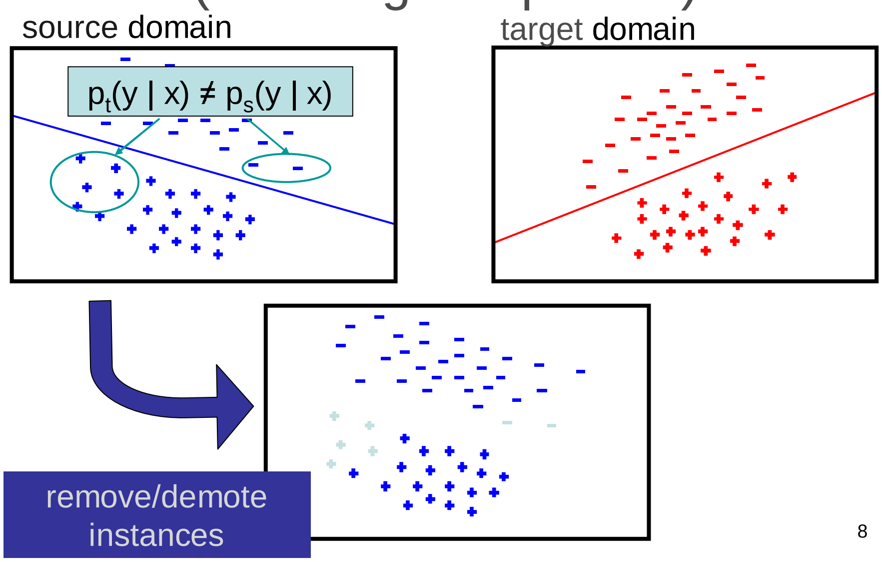
6

# Where Does the Difference Come from?



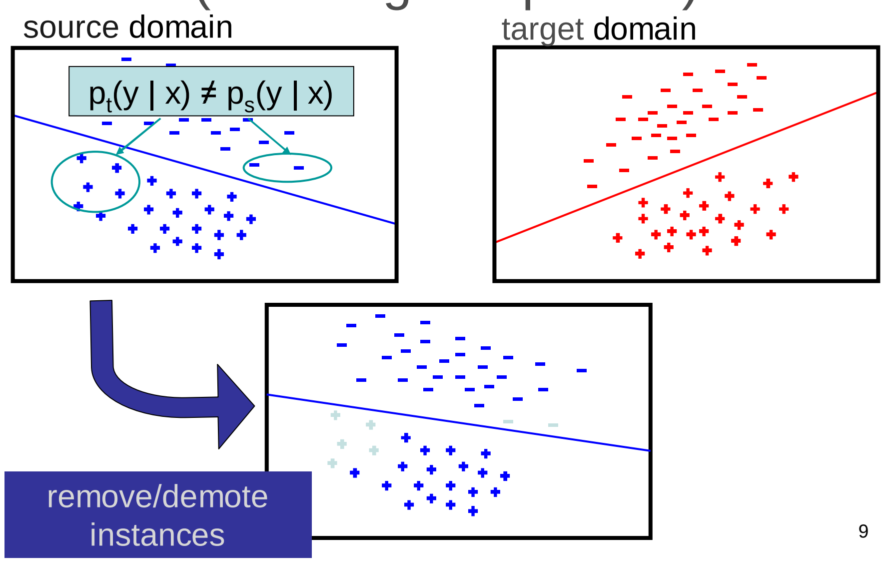

7

# An Instance Weighting Solution (Labeling Adaptation)



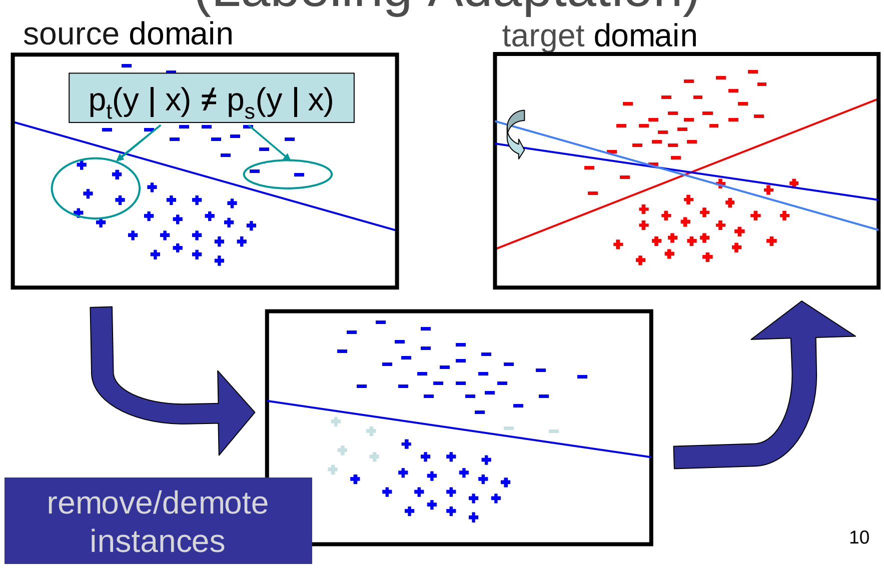

8

# An Instance Weighting Solution (Labeling Adaptation)



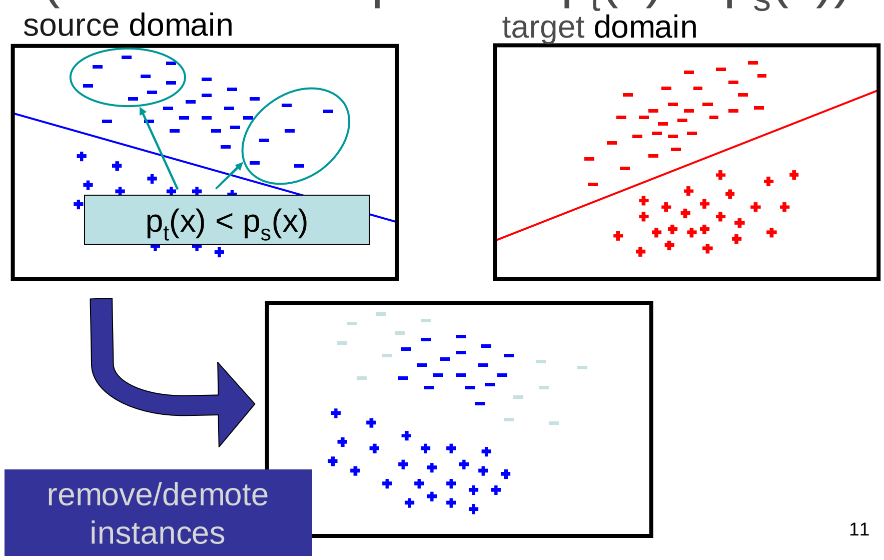

9

# An Instance Weighting Solution (Labeling Adaptation)



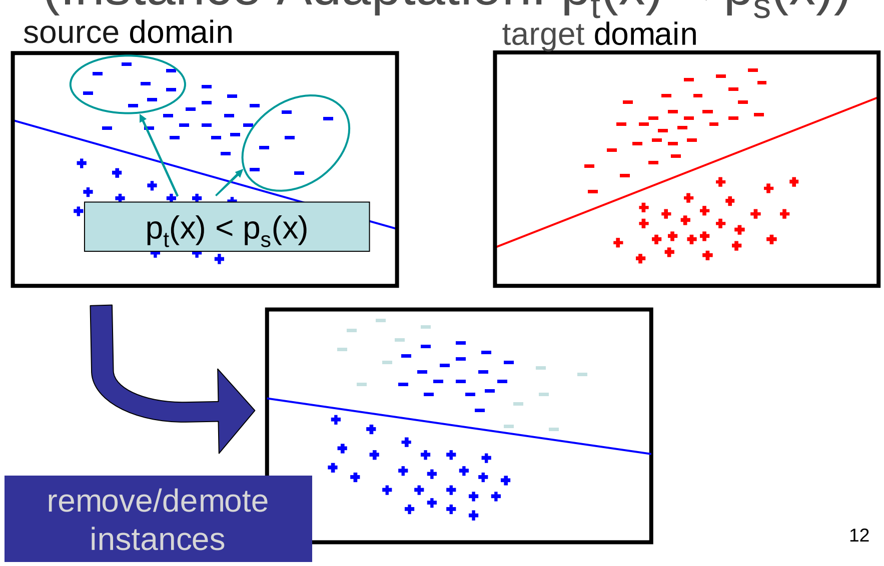

10

# An Instance Weighting Solution (Instance Adaptation: $p_t(x) < p_s(x)$)



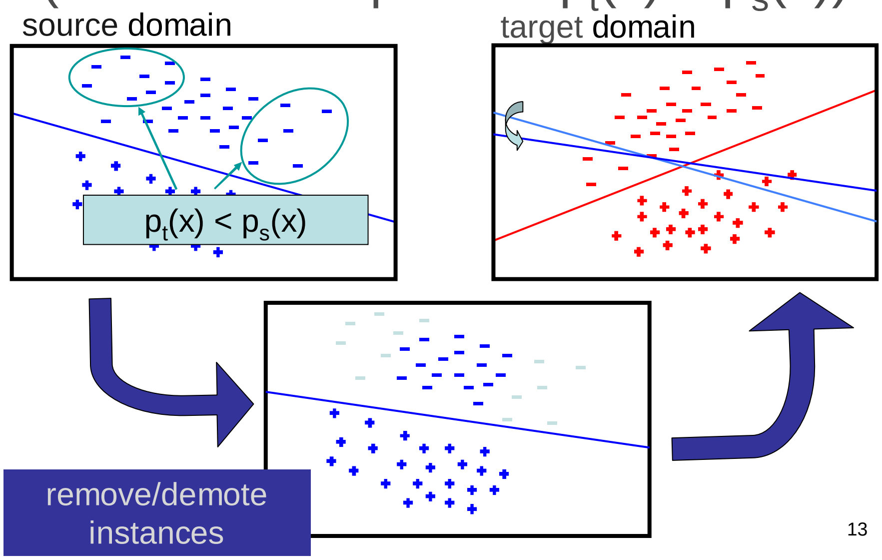

11

# An Instance Weighting Solution (Instance Adaptation: $p_t(x) < p_s(x)$)



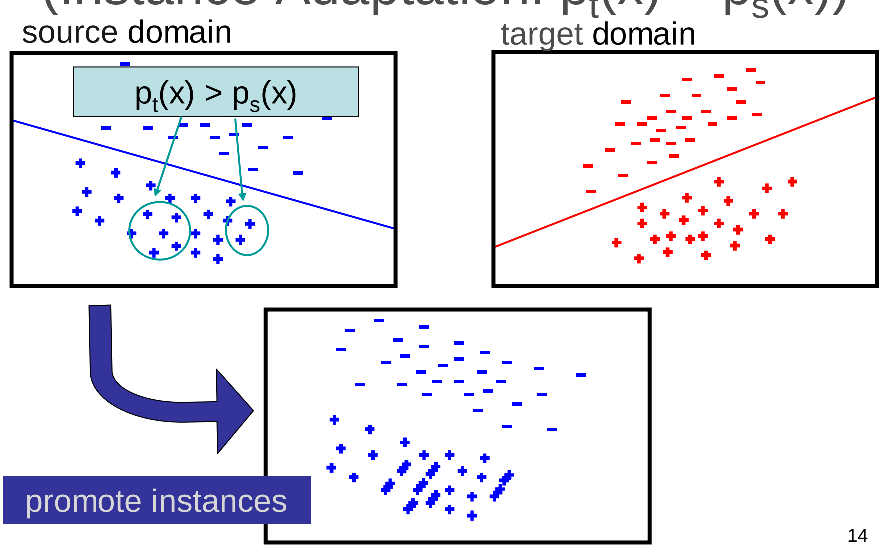

12

## An Instance Weighting Solution (Instance Adaptation: $p_t(x) < p_s(x)$)



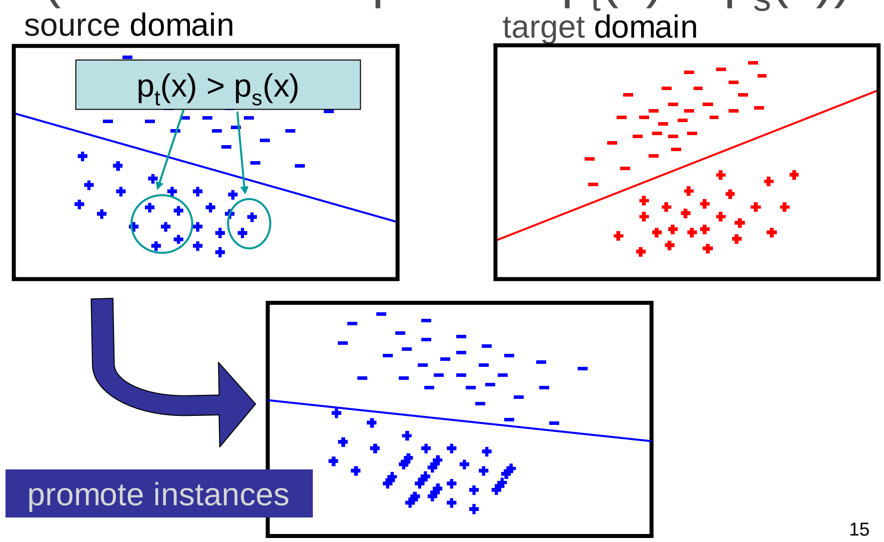

13

## An Instance Weighting Solution (Instance Adaptation: $p_t(x) > p_s(x)$)



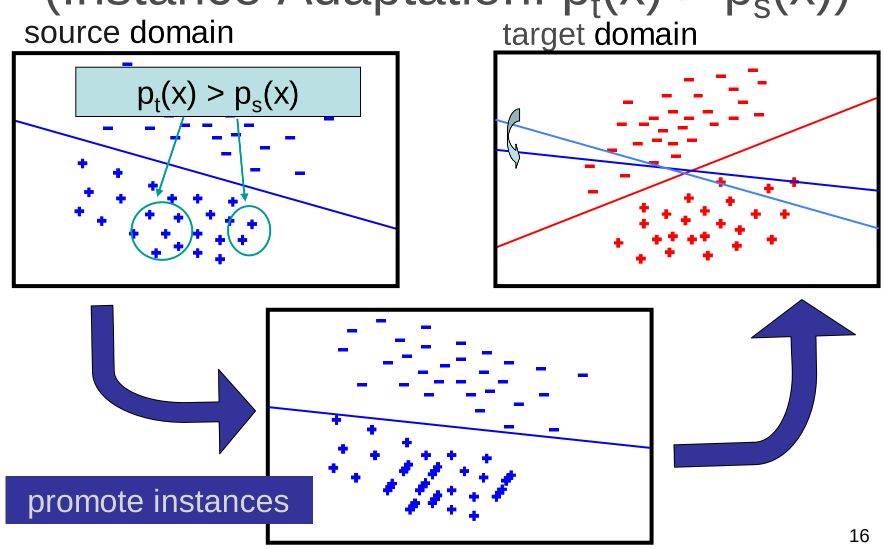

14

# An Instance Weighting Solution
# (Instance Adaptation: $p_t(x) > p_s(x)$)



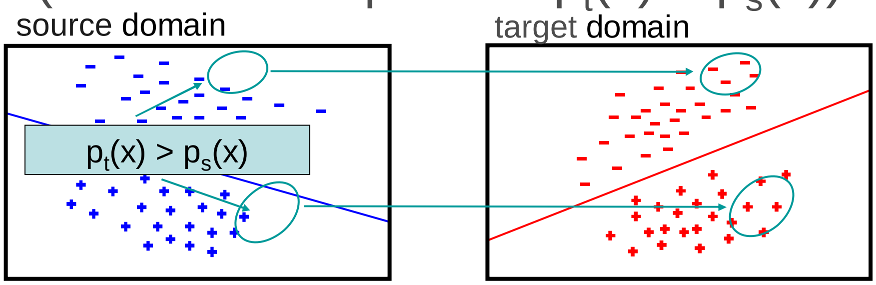

15

# An Instance Weighting Solution
# (Instance Adaptation: $p_t(x) > p_s(x)$)




16

## An Instance Weighting Solution (Instance Adaptation: $p_t(x) > p_s(x)$)




- Labeled target domain instances are useful
- Unlabeled target domain instances may also be useful

17

## The Exact Objective Function

true marginal and conditional probabilities in the target domain

log likelihood (log loss function)

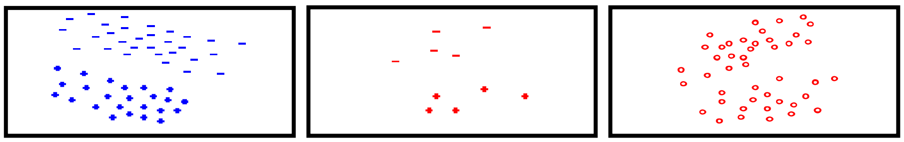
$$\theta_t^* = \arg\max_{\theta} \int_X p_t(x) \sum_{y \in Y} p_t(y \mid x) \log p(y \mid x; \theta) dx$$

unknown

18

# Three Sets of Instances

$D_s$ $D_{t,l}$ $D_{t,u}$

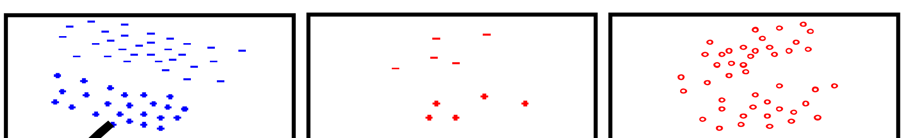


$$\theta_t^* = \arg\max_\theta \int_X p_t(x) \sum_{y \in Y} p_t(y \mid x) \log p(y \mid x; \theta) dx$$

19

# Three Sets of Instances: Using $D_s$

$D_s$ $D_{t,l}$ $D_{t,u}$



$$\theta_t^* = \arg\max_\theta \int_X p_t(x) \sum_{y \in Y} p_t(y \mid x) \log p(y \mid x; \theta) dx$$

$X \approx D_s$

$$\approx \arg\max_\theta \frac{1}{\sum_{i=1}^{N_s} \alpha_i \beta_i} \sum_{i=1}^{N_s} \alpha_i \beta_i \log p(y_i^s \mid x_i^s; \theta)$$

$$\beta_i = \frac{p_t(x_i^s)}{p_s(x_i^s)}$$

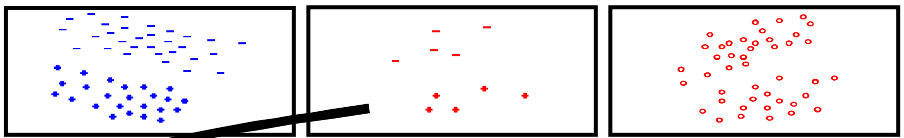
$$\alpha_i = \frac{p_t(y_i^s \mid x_i^s)}{p_s(y_i^s \mid x_i^s)}$$

**in principle, non-parametric density estimation; in practice, high dimensional data (future work)**

need labeled target data

## Three Sets of Instances: Using $D_{t,l}$

$D_s$ $\quad D_{t,l}$ $\quad D_{t,u}$



$$\theta_t^* = \arg\max_\theta \int_X p_t(x) \sum_{y \in Y} p_t(y \mid x) \log p(y \mid x; \theta) dx$$

$$\approx \arg\max_\theta \frac{1}{N_{t,l}} \sum_{j=1}^{N_{t,l}} \log p(y_j^t \mid x_j^t; \theta)$$

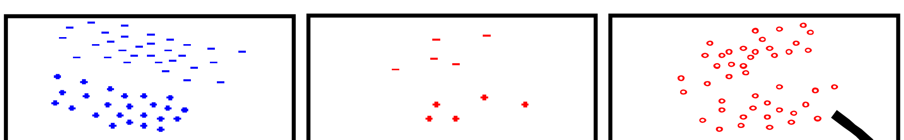
$X \approx D_{t,l}$

small sample size, estimation not accurate

21

## Three Sets of Instances: Using $D_{t,u}$

$D_s$ $\quad D_{t,l}$ $\quad D_{t,u}$



$$\theta_t^* = \arg\max_\theta \int_X p_t(x) \sum_{y \in Y} p_t(y \mid x) \log p(y \mid x; \theta) dx$$

$$\approx \arg\max_\theta \frac{1}{C_{t,u}} \sum_{k=1}^{N_{t,u}} \sum_{y \in Y} \gamma_k(y) \log p(y \mid x_k^t; \theta)$$

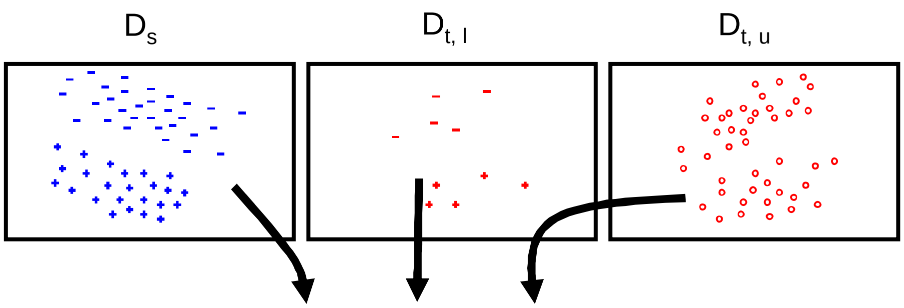
$X \approx D_{t,u}$

$$\gamma_k(y) = p_t(y \mid x_k^{t,u})$$

pseudo labels (e.g. bootstrapping, EM)

22

# Using All Three Sets of Instances




$$X \approx D_s + D_{t,l} + D_{t,u}?$$

$$\theta_t^* = \arg\max_\theta \int_X p_t(x) \sum_{y \in Y} p_t(y \mid x) \log p(y \mid x; \theta) dx$$

$$\approx ?$$

23

# A Combined Framework

$$\hat{\theta} = \arg\max_\theta [\lambda_s \frac{1}{C_s} \sum_{i=1}^{N_s} \alpha_i \beta_i \log p(y_i^s \mid x_i^s; \theta)$$

$$+ \lambda_{t,l} \frac{1}{C_{t,l}} \sum_{j=1}^{N_{t,l}} \log p(y_i^t \mid x_i^t; \theta)$$

$$+ \lambda_{t,u} \frac{1}{C_{t,u}} \sum_{k=1}^{N_{t,u}} \sum_{y \in Y} \gamma_k(y) \log p(y \mid x_k^t; \theta)$$

$$+ \log p(\theta)]$$

$$\lambda_s + \lambda_{t,l} + \lambda_{t,u} = 1$$

a flexible setup covering both standard methods and new domain adaptive methods

24

## Standard Supervised Learning using only $D_s$

$$\hat{\theta} = \arg\max_{\theta}[\lambda_s \frac{1}{C_s}\sum_{i=1}^{N_s}\alpha_i\beta_i \log p(y_i^s \mid x_i^s;\theta)$$

$$+\lambda_{t,l}\frac{1}{C_{t,l}}\sum_{j=1}^{N_{t,l}}\log p(y_i^t \mid x_i^t;\theta)$$

$$+\lambda_{t,u}\frac{1}{C_{t,u}}\sum_{k=1}^{N_{t,u}}\sum_{y\in Y}\gamma_k(y)\log p(y \mid x_k^t;\theta)$$

$$+\log p(\theta)]$$

$\alpha_i = \beta_i = 1$, $\lambda_s = 1$, $\lambda_{t,l} = \lambda_{t,u} = 0$

25

## Standard Supervised Learning using only $D_{t,l}$

$$\hat{\theta} = \arg\max_{\theta}[\lambda_s \frac{1}{C_s}\sum_{i=1}^{N_s}\alpha_i\beta_i \log p(y_i^s \mid x_i^s;\theta)$$

$$+\lambda_{t,l}\frac{1}{C_{t,l}}\sum_{j=1}^{N_{t,l}}\log p(y_i^t \mid x_i^t;\theta)$$

$$+\lambda_{t,u}\frac{1}{C_{t,u}}\sum_{k=1}^{N_{t,u}}\sum_{y\in Y}\gamma_k(y)\log p(y \mid x_k^t;\theta)$$

$$+\log p(\theta)]$$

$\lambda_{t,l} = 1$, $\lambda_s = \lambda_{t,u} = 0$

26

## Standard Supervised Learning using both $D_s$ and $D_{t,l}$

$$\hat{\theta} = \arg\max_{\theta} [\lambda_s \frac{1}{C_s} \sum_{i=1}^{N_s} \alpha_i \beta_i \log p(y_i^s \mid x_i^s; \theta)$$
$$+ \lambda_{t,l} \frac{1}{C_{t,l}} \sum_{j=1}^{N_{t,l}} \log p(y_i^t \mid x_i^t; \theta)$$
$$+ \lambda_{t,u} \frac{1}{C_{t,u}} \sum_{k=1}^{N_{t,u}} \sum_{y \in Y} \gamma_k(y) \log p(y \mid x_k^t; \theta)$$
$$+ \log p(\theta)]$$

$\alpha_i = \beta_i = 1$, $\lambda_s = N_s/(N_s+N_{t,l})$, $\lambda_{t,l} = N_{t,l}/(N_s+N_{t,l})$, $\lambda_{t,u} = 0$



## Domain Adaptive Heuristic: **1. Instance Pruning**

$$\hat{\theta} = \arg\max_{\theta} [\lambda_s \frac{1}{C_s} \sum_{i=1}^{N_s} \alpha_i \beta_i \log p(y_i^s \mid x_i^s; \theta)$$
$$+ \lambda_{t,l} \frac{1}{C_{t,l}} \sum_{j=1}^{N_{t,l}} \log p(y_i^t \mid x_i^t; \theta)$$
$$+ \lambda_{t,u} \frac{1}{C_{t,u}} \sum_{k=1}^{N_{t,u}} \sum_{y \in Y} \gamma_k(y) \log p(y \mid x_k^t; \theta)$$
$$+ \log p(\theta)]$$

$\alpha_i = 0$ if $(x_i, y_i)$ are predicted incorrectly by a model trained from $D_{t,l}$; 1 otherwise

## Domain Adaptive Heuristic:
## **2. $D_{t,l}$ with higher weights**

$$\hat{\theta} = \arg\max_{\theta}[\lambda_s \frac{1}{C_s}\sum_{i=1}^{N_s}\alpha_i\beta_i \log p(y_i^s \mid x_i^s;\theta)$$
$$+\lambda_{t,l}\frac{1}{C_{t,l}}\sum_{j=1}^{N_{t,l}}\log p(y_i^t \mid x_i^t;\theta)$$
$$+\lambda_{t,u}\frac{1}{C_{t,u}}\sum_{k=1}^{N_{t,u}}\sum_{y\in Y}\gamma_k(y)\log p(y \mid x_k^t;\theta)$$
$$+\log p(\theta)]$$

$\lambda_s < N_s/(N_s+N_{t,l})$, $\lambda_{t,l} > N_{t,l}/(N_s+N_{t,l})$

29

## Standard Bootstrapping

$$\hat{\theta} = \arg\max_{\theta}[\lambda_s \frac{1}{C_s}\sum_{i=1}^{N_s}\alpha_i\beta_i \log p(y_i^s \mid x_i^s;\theta)$$
$$+\lambda_{t,l}\frac{1}{C_{t,l}}\sum_{j=1}^{N_{t,l}}\log p(y_i^t \mid x_i^t;\theta)$$
$$+\lambda_{t,u}\frac{1}{C_{t,u}}\sum_{k=1}^{N_{t,u}}\sum_{y\in Y}\gamma_k(y)\log p(y \mid x_k^t;\theta)$$
$$+\log p(\theta)]$$

$\gamma_k(y) = 1$ if $p(y \mid x_k)$ is large; 0 otherwise

30

## Domain Adaptive Heuristic: **3. Balanced Bootstrapping**

$$\hat{\theta} = \arg\max_{\theta} [\lambda_s \frac{1}{C_s} \sum_{i=1}^{N_s} \alpha_i \beta_i \log p(y_i^s \mid x_i^s; \theta)$$

$$+ \lambda_{t,l} \frac{1}{C_{t,l}} \sum_{j=1}^{N_{t,l}} \log p(y_i^t \mid x_i^t; \theta)$$

$$+ \lambda_{t,u} \frac{1}{C_{t,u}} \sum_{k=1}^{N_{t,u}} \sum_{y \in Y} \gamma_k(y) \log p(y \mid x_k^t; \theta)$$

$$+ \log p(\theta)]$$

$\gamma_k(y) = 1$ if $p(y \mid x_k)$ is large; 0 otherwise

$\lambda_s = \lambda_{t,u} = 0.5$

31

## Experiments

- Three NLP tasks:
  - POS tagging: WSJ (Penn TreeBank) → Oncology (biomedical) text (Penn BioIE)
  - NE type classification: newswire → conversational telephone speech (CTS) and web-log (WL) (ACE 2005)
  - Spam filtering: public email collection → personal inboxes (u01, u02, u03) (ECML/PKDD 2006)

32

# Experiments

- Three heuristics:
  1. Instance pruning
  2. $D_{t,l}$ with higher weights
  3. Balanced bootstrapping
- Performance measure: accuracy

33

# Instance Pruning

## Removing "Misleading" Instances from $D_s$

POS

| *k* | **Oncology** |
|---|---|
| 0 | 0.8630 |
| 8000 | 0.8709 |
| 16000 | 0.8714 |
| all | 0.8720 |

NE Type

| *k* | **CTS** | *k* | **WL** |
|---|---|---|---|
| 0 | 0.7815 | 0 | 0.7045 |
| 1600 | 0.8640 | 1200 | 0.6975 |
| 3200 | 0.8825 | 2400 | 0.6795 |
| all | 0.8830 | all | 0.6600 |

Spam

| *k* | **User 1** | **User 2** | **User 3** |
|---|---|---|---|
| 0 | 0.6306 | 0.6950 | 0.7644 |
| 300 | 0.6611 | 0.7228 | 0.8222 |
| 600 | 0.7911 | 0.8322 | 0.8328 |
| all | 0.8106 | 0.8517 | 0.8067 |

useful in most cases; failed in some case

When is it guaranteed to work? (future work)

34

# $D_{t,l}$ with Higher Weights
## until $D_s$ and $D_{t,l}$ Are Balanced

POS

| method | Oncology |
|---|---|
| $D_s$ | 0.8630 |
| $D_s + D_{t,l}$ | 0.9349 |
| $D_s + 10D_{t,l}$ | 0.9429 |
| $D_s$ + 20 | [illegible] |

NE Type

| method | CTS | WL |
|---|---|---|
| $D_s$ | 0.7815 | 0.7045 |
| $D_s + D_{t,l}$ | 0.9340 | 0.7735 |
| $D_s + 5D_{t,l}$ | 0.9360 | 0.7820 |
| [illegible] | [illegible] | 7840 |

$D_{t,l}$ is very useful

promoting $D_{t,l}$ is more useful

| method | User 1 | User 2 | User 3 |
|---|---|---|---|
| $D_s$ | 0.6306 | 0.6950 | 0.7644 |
| $D_s + D_{t,l}$ | 0.9572 | 0.9572 | 0.9461 |
| $D_s + 5D_{t,l}$ | 0.9628 | 0.9611 | 0.9601 |
| $D_s + 10D_{t,l}$ | 0.9639 | 0.9628 | 0.9633 |

35

# Instance Pruning
## + $D_{t,l}$ with Higher Weights

POS

| method | Oncology |
|---|---|
| $D_s + 20D_{t,l}$ | 0.9443 |
| $D_s' + 20D_{t,l}$ | 0.9422 |

NE Type

| Method | CTS | WL |
|---|---|---|
| $D_s + 10D_{t,l}$ | 0.9355 | 0.7840 |
| $D_s' + 10D_{t,l}$ | 0.8950 | 0.6670 |

The two heuristics do not work well together

How to combine heuristics? (future work)

| method | User 1 | User 2 | User 3 |
|---|---|---|---|
| $D_s + 10D_{t,l}$ | 0.9639 | 0.9628 | 0.9633 |
| $D_s' + 10D_{t,l}$ | 0.9717 | 0.9478 | 0.9494 |

36

# Balanced Bootstrapping

POS

| method | Oncology |
|---|---|
| supervised | 0.8630 |
| standard bootstrap | 0.8728 |
| balanced | **0.8750** |

NE Type

| method | CTS | WL |
|---|---|---|
| supervised | 0.7781 | 0.7351 |
| standard bootstrap | 0.8917 | 0.7498 |
| balanced | **0.8923** | **0.7523** |

Promoting target instances is useful, even with pseudo labels

| method | User 1 | User 2 | User 3 |
|---|---|---|---|
| supervised | 0.6476 | 0.6976 | 0.8068 |
| standard bootstrap | 0.8720 | 0.9212 | 0.9760 |
| balanced bootstrap | **0.8816** | **0.9256** | **0.9772** |

37

# Conclusions

- Formally analyzed the domain adaptation from an instance weighting perspective
- Proposed an instance weighting framework for domain adaptation
  - Both labeled and unlabeled instances
  - Various weight parameters
- Proposed a number of heuristics to set the weight parameters
- Experiments showed the effectiveness of the heuristics

38

## Future Work

- Combining different heuristics
- Principled ways to set the weight parameters
  - Density estimation for setting β

39

## Thank You!