# Semi-supervised learning and domain adaptation

**Barbara Plank**
**CST, University of Copenhagen**

Guest lecture, University of Gothenburg
October 23, 2015

# Outline

- Semi-supervised learning
    - What is SSL?
- BREAK
- Domain adaptation
    - different ways to tackle DA

# What you have seen so far….

x → y

supervised learning

| | | |
|---|---|---|
| **sentiment analysis** | **tweet** | **label** |
| **tagging** | **sentence** | **sequence** |
| **parsing** | **sentence** |  |

labeled data

$$\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$$

- nearest neighbor
- perceptron
- Naive Bayes
- structured perceptron
- MST parsing
- neural networks

# What if there is no y?

x → ?

unsupervised learning

unlabeled data

semi-supervised learning

labeled data

unlabeled data

# What is semi-supervised learning (SSL?)

- labeled data (e.g. Named Entity Recognition, NER)

labeled
data

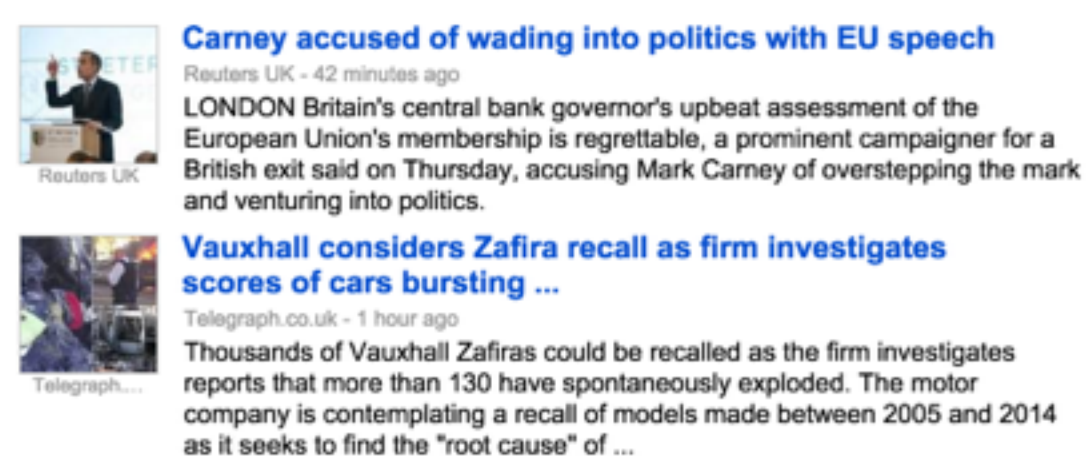
… says **John Brown/PER,** vice president of **ABC ltd/ORG**

**Sweeeats/ORG** corp in **Philadephia/LOC**

**Can we build a better model by exploiting unlabeled data?**

- Lots more unlabeled data

unlabeled
data


Carney accused of wading into politics with EU speech
Reuters UK - 42 minutes ago
LONDON Britain's central bank governor's upbeat assessment of the European Union's membership is regrettable, a prominent campaigner for a British exit said on Thursday, accusing Mark Carney of overstepping the mark and venturing into politics.
Reuters UK

Vauxhall considers Zafira recall as firm investigates scores of cars bursting ...
Telegraph.co.uk - 1 hour ago
Thousands of Vauxhall Zafiras could be recalled as the firm investigates reports that more than 130 have spontaneously exploded. The motor company is contemplating a recall of models made between 2005 and 2014 as it seeks to find the "root cause" of ...
Telegraph...

# Anti-SSL arguments

- "We'll find the time and money to annotate more labeled data"
- Hmm, but:
  - Annotating PT WSJ took a decade!
  - What about building a NER for, say, Irish? Who is going to annotate it for me?

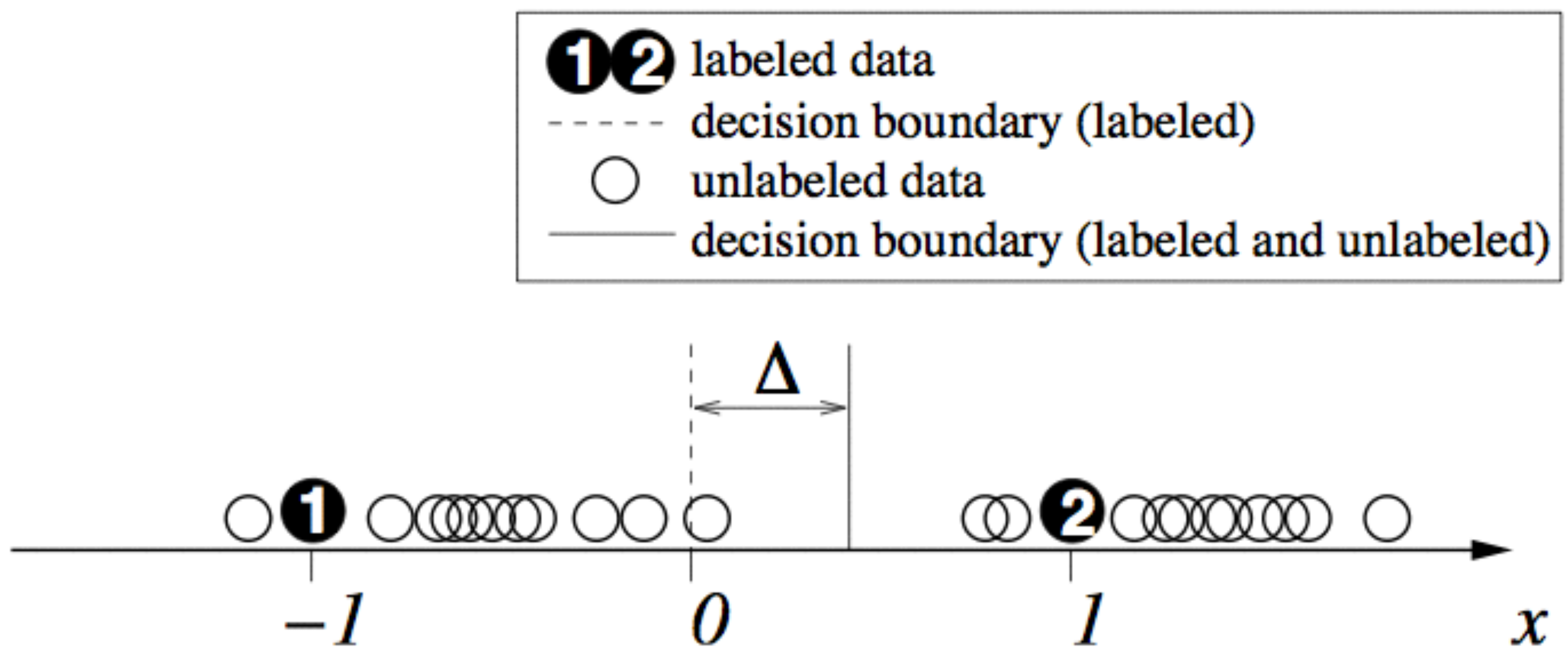
# Pro-SSL arguments

- I have a good idea, but I can’t afford to label data
- I have some annotated data, but I have even more unlabeled data
- I have labeled data from one domain, but I want to build a model for another domain: **domain adaptation**
- **Cognitive Science motivation:** Also humans do semi-supervised learning (children learning by parent pointing to animal and saying “dog”, but also by just observing environment)

# What is SSL? More formally

- Learning from both labeled and unlabeled data:
  - $l$ labeled instances $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and
  - $u$ unlabeled instances $\{\mathbf{x}_j\}_{j=l+1}^{l+u}$ , usually $u >> l$
- **Goal**: better classifier than from labeled data alone

# How can unlabeled data ever help?



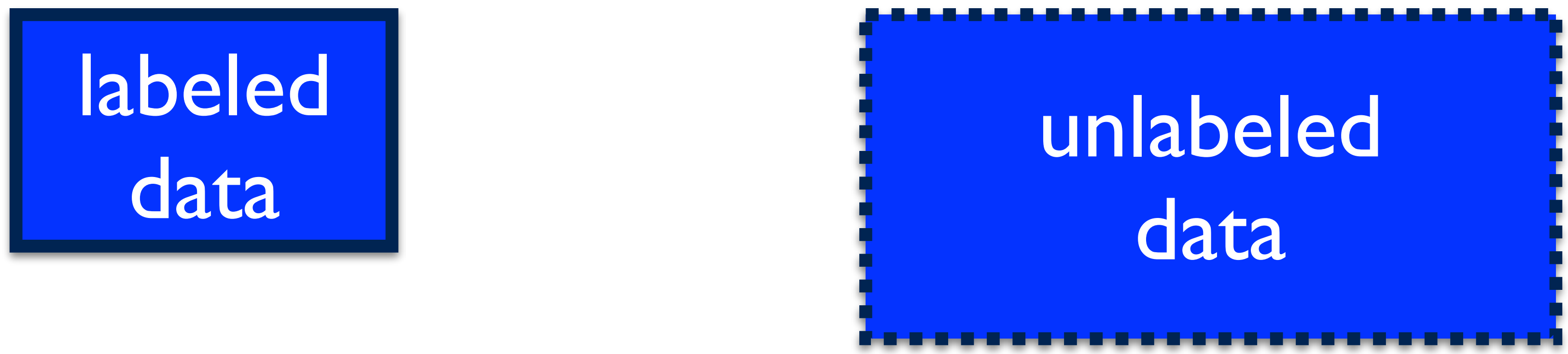

Zhu et al., (2007)

# Bootstrapping methods

A widely used SSL bootstrapping algorithm:
**self-training**

# Self-training

- Input: labeled data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$, unlabeled data $\{\mathbf{x}_j\}_{j=l+1}^{l+u}$.




- Procedure:

1. Initially, let $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and $U = \{\mathbf{x}_j\}_{j=l+1}^{l+u}$.
2. Repeat:
3. Train $f$ from $L$ using supervised learning.
4. Apply $f$ to the unlabeled instances in $U$.
5. Remove a subset $S$ from $U$; add $\{(\mathbf{x}, f(\mathbf{x})) | \mathbf{x} \in S\}$ to $L$.

# Self-training

- Procedure:

1. Initially, let $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and $U = \{\mathbf{x}_j\}_{j=l+1}^{l+u}$.
2. Repeat:
3. Train $f$ from $L$ using supervised learning.
4. Apply $f$ to the unlabeled instances in $U$.
5. Remove a subset $S$ from $U$; add $\{(\mathbf{x}, f(\mathbf{x})) | \mathbf{x} \in S\}$ to $L$.

- Parameters, e.g., iterations, pool/growth size, select

- Questions:

Q1: This is called a wrapper method. Why?
Q2: Why might this help to build a better system?
Q3: What might go wrong?

# Self-training: Summary

| | |
|---|---|
| Q1: Wrapper? | choice of f left open |
| Q2: Works when? | broad margin, expected low error |
| Q3: Limitations? | errors get reinforced |
| Variants? | Yes, many, e.g., delible self-training, weigh instances,… |

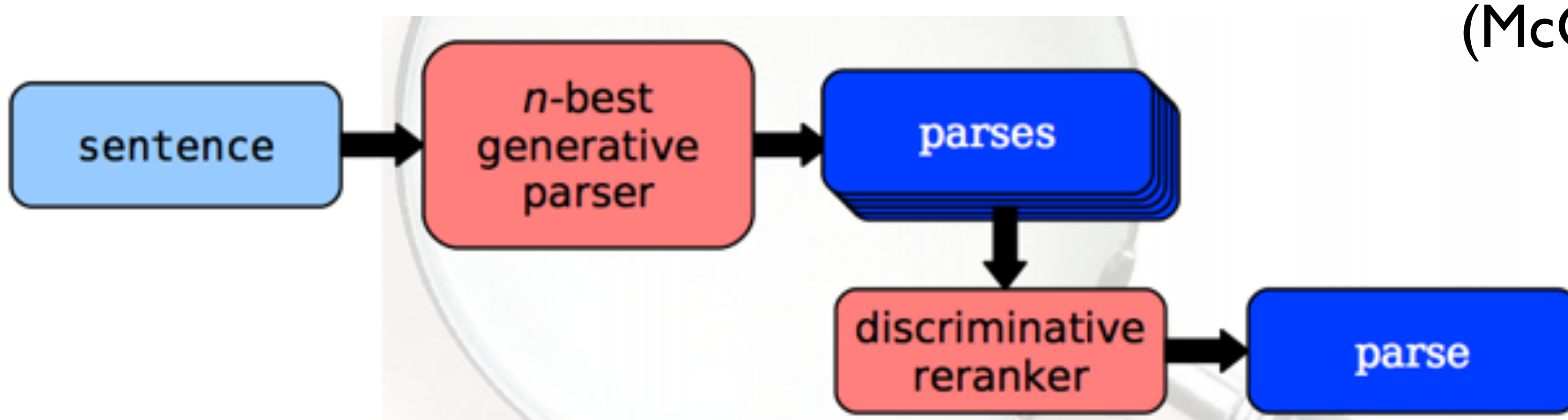
# Self-training for Parsing

| | Parser type | Seed size | Iterations | Improved? |
|---|---|---|---|---|
| **Charniak (1997)** | Generative | Large | Single | No |
| **McClosky et al. (2006)** | Gen.+Disc. | Large | Single | **Yes** |
| **Steedman et al. (2003)** | Generative | Small | Multiple | No |
| **Reichart and Rappoport (2007)** | Generative | Small | Single | **Yes** |

(large = ~40k sentences, small = <1k sentences)

**Summary of self-training for parsing experiments**

(McClosky et al., 2008)



labeled data

unlabeled data

What if gap between data is large? different ***domains***?

SOURCE

TARGET

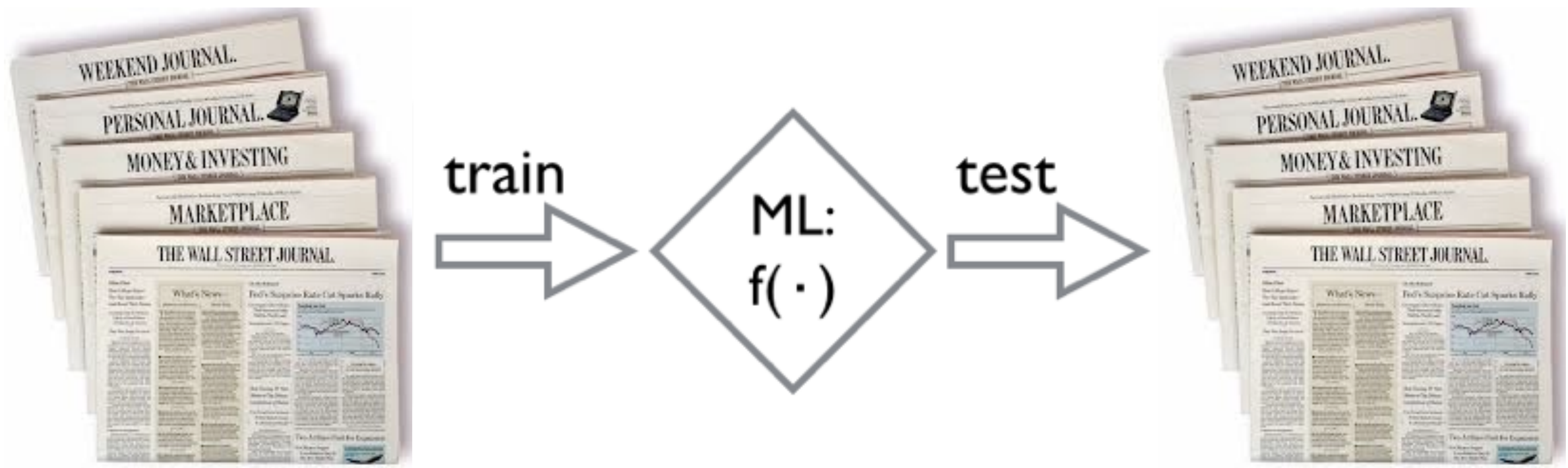
# Off-the-shelf POS tagger

The/DT share/NN rose/VBD to/TO 10/CD $/$ a/DT unit/NN ./.

May/**NNP** I/**PRP** brrow/VBP 10bucks/**UH**

# Why does it fail?

# Machine Learning (ML)



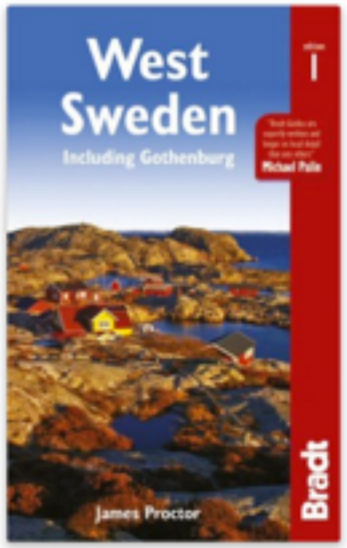

- general assumption: i.i.d. samples

BREAK

# Outline

- Semi-supervised learning
- BREAK
- Domain adaptation
    - different ways to tackle sample selection bias

# Amazon reviews




**very good indeed for its size**

By pcp0827 on July 3, 2014

Format: Paperback

A thin, but excellent, guide to the beautiful region of Gothenburg and the western archipelago. A small number of excellent recommendations of hotels and restaurants.

**Great essential for the baker in your life....you!**

By Aussiebabe on May 20, 2005

Color: Onyx Black

I love the mixer's power, capacity and ease of use. It makes even garlic mashed potatoes heavenly. I selected a black model to match my Viking kitchens without realizing the color would need more frequent cleaning. If you use this mixer every day buy a cover or another color to save time.

# Sample Selection Bias


**domain adaptation**

domain, genre, time,...

SOURCE

The CROSS-DOMAIN GULF

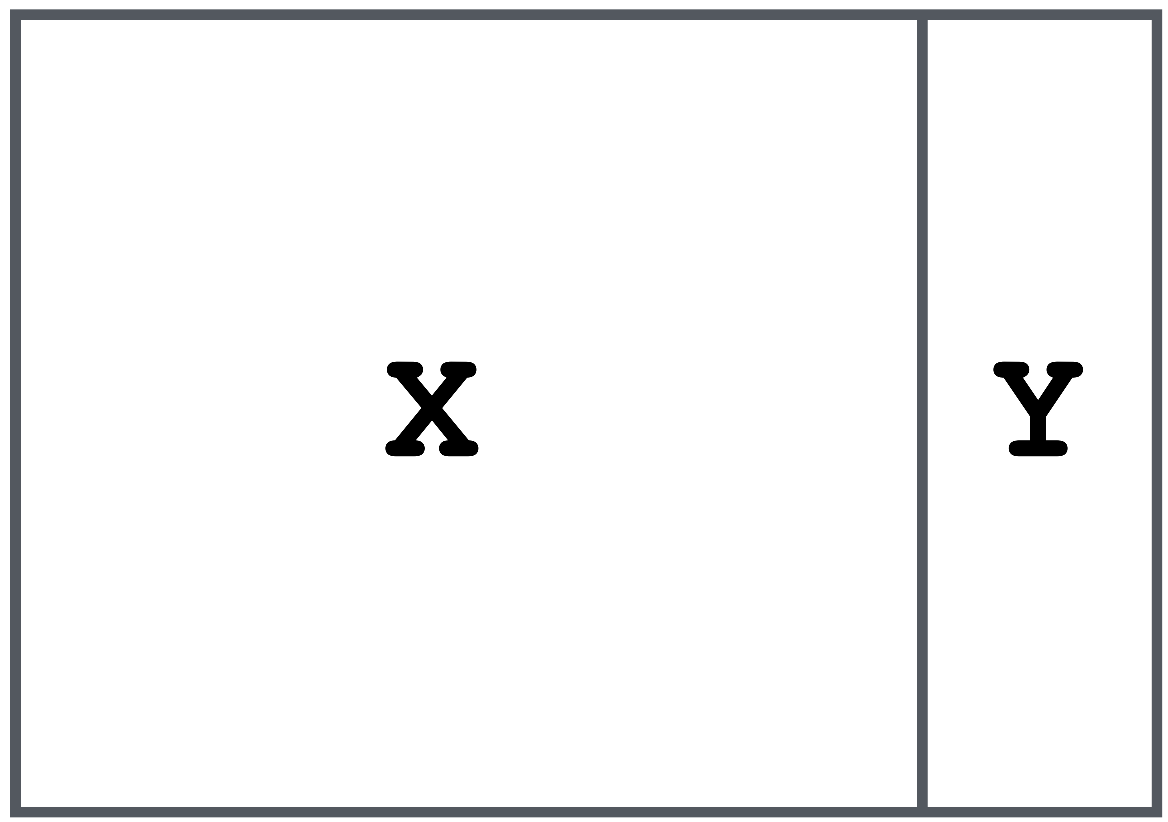
TARGET

# Generalization!

The ability of a learning machine to perform accurately on new, unseen examples

# Possible approaches

**training data:**

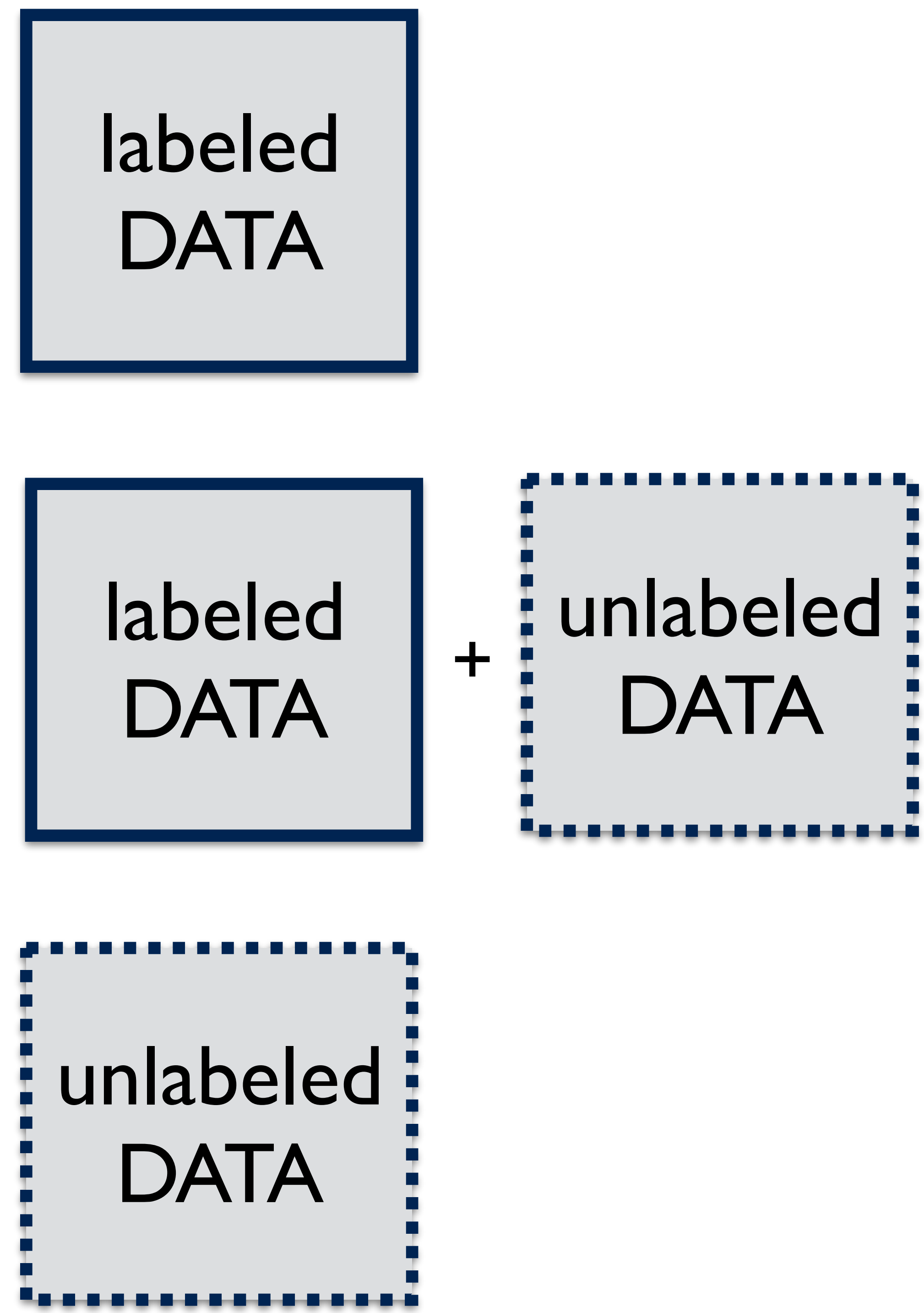



1. add more **X**
2. modify **X** (make more similar to target)
3. modify **Y** (we'll not touch upon that here)

# First, a few words on terminology…

# General ML trichotomy

1. supervised ML
2. semi-supervised ML
3. unsupervised ML



# Domain Adaptation: 4 Setups

1. supervised DA
(e.g. Daumè, 2007)

labeled SOURCE | labeled TARGET

2. semi-supervised DA
(e.g. Daumè, 2010;
Chang, Conner & Roth, 2010)

labeled SOURCE | labeled TARGET | unlabeled TARGET

before 2010 (Plank, 2011)

3. unsupervised DA
(e.g. Blitzer et al., 2007;
McClosky et al., 2008)

labeled SOURCE | unlabeled TARGET

2012 onwards

4. blind/unknown DA
(e.g. Søgaard & Johannsen, 2012; Plank & Moschitti, 2013; Elming et al., 2014)

labeled SOURCE | ?? UNKNOWN ?? at test time

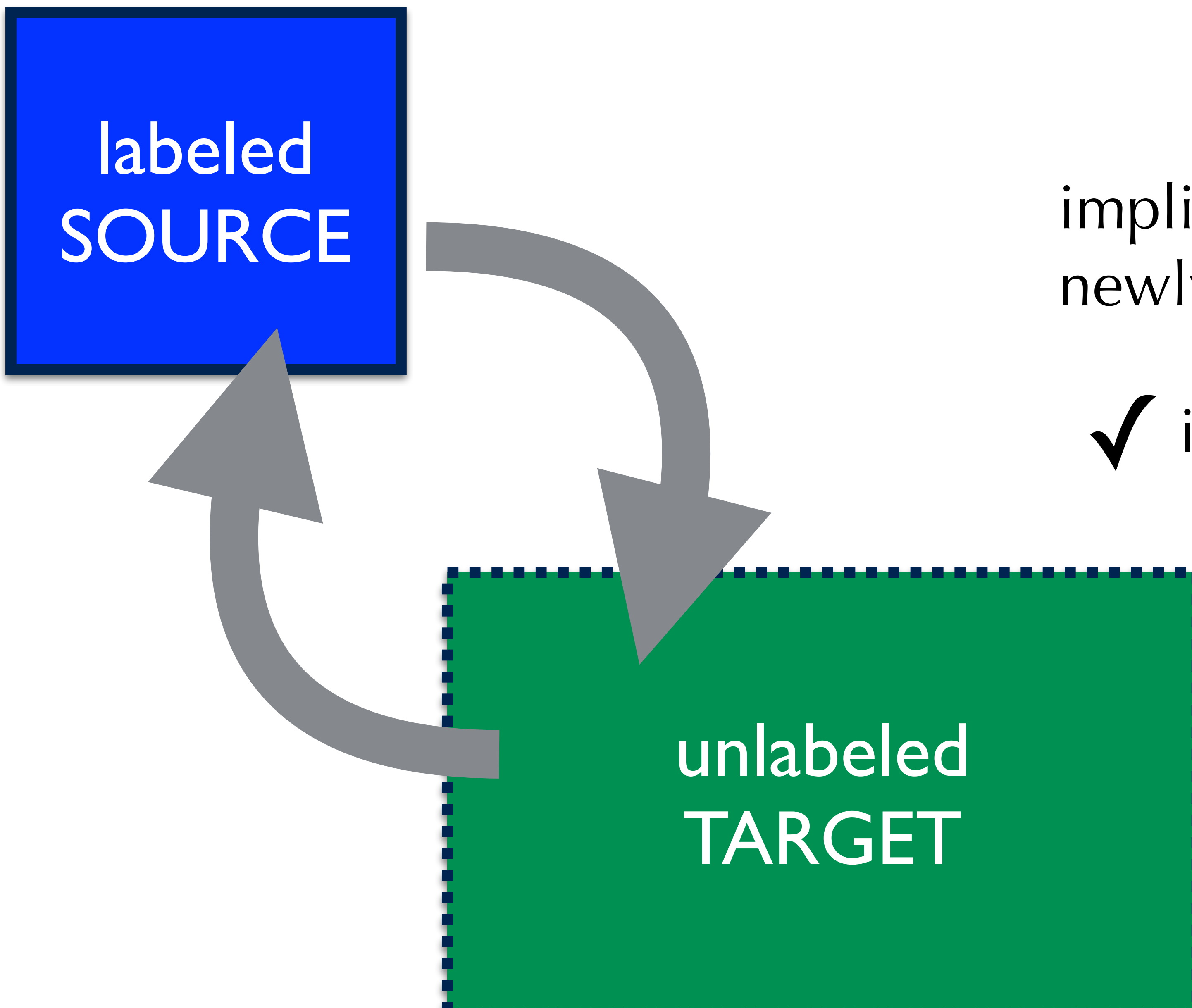
We’ll focus on unsupervised DA

# semi-supervised machine learning

to address the biased selection of sentences (**x**)

# Semi-supervised learning (SSL)

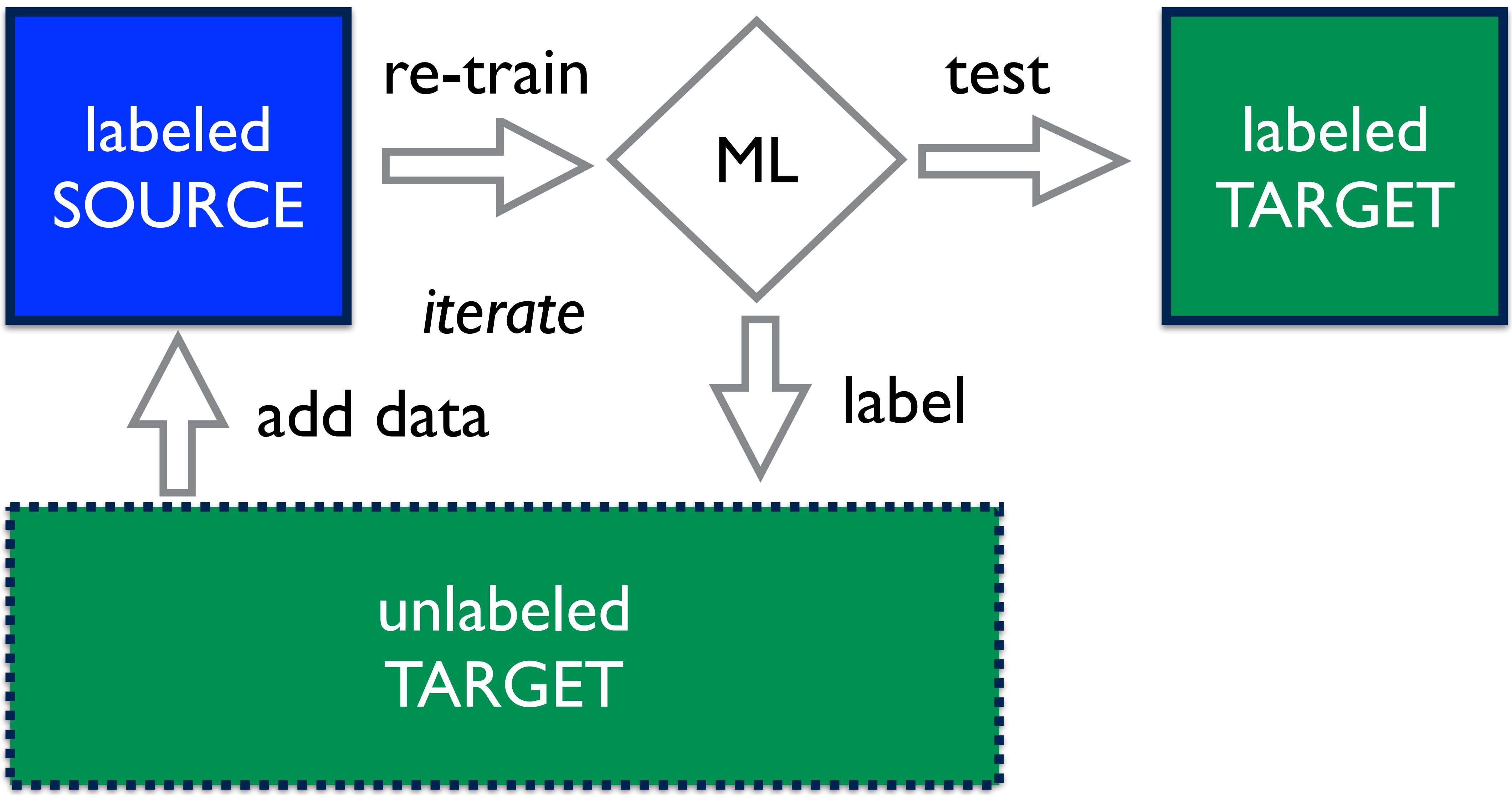
How can it help us to bridge the cross-domain gulf?




implicitly adapting by adding newly labeled data from TARGET

✓ if gulf is not too wide

# Self-training



# Self-training

**Pros**

✓ Simple wrapper method

✓ Can correct bias to some extent (if expected error on target is low/gulf not too wide)

**Cons**

- many parameters
- might introduce more bias (both selection and label bias)

Self-training alone often does not work, needs some additional 'signal'

e.g.

Parsing: use of reranker on top

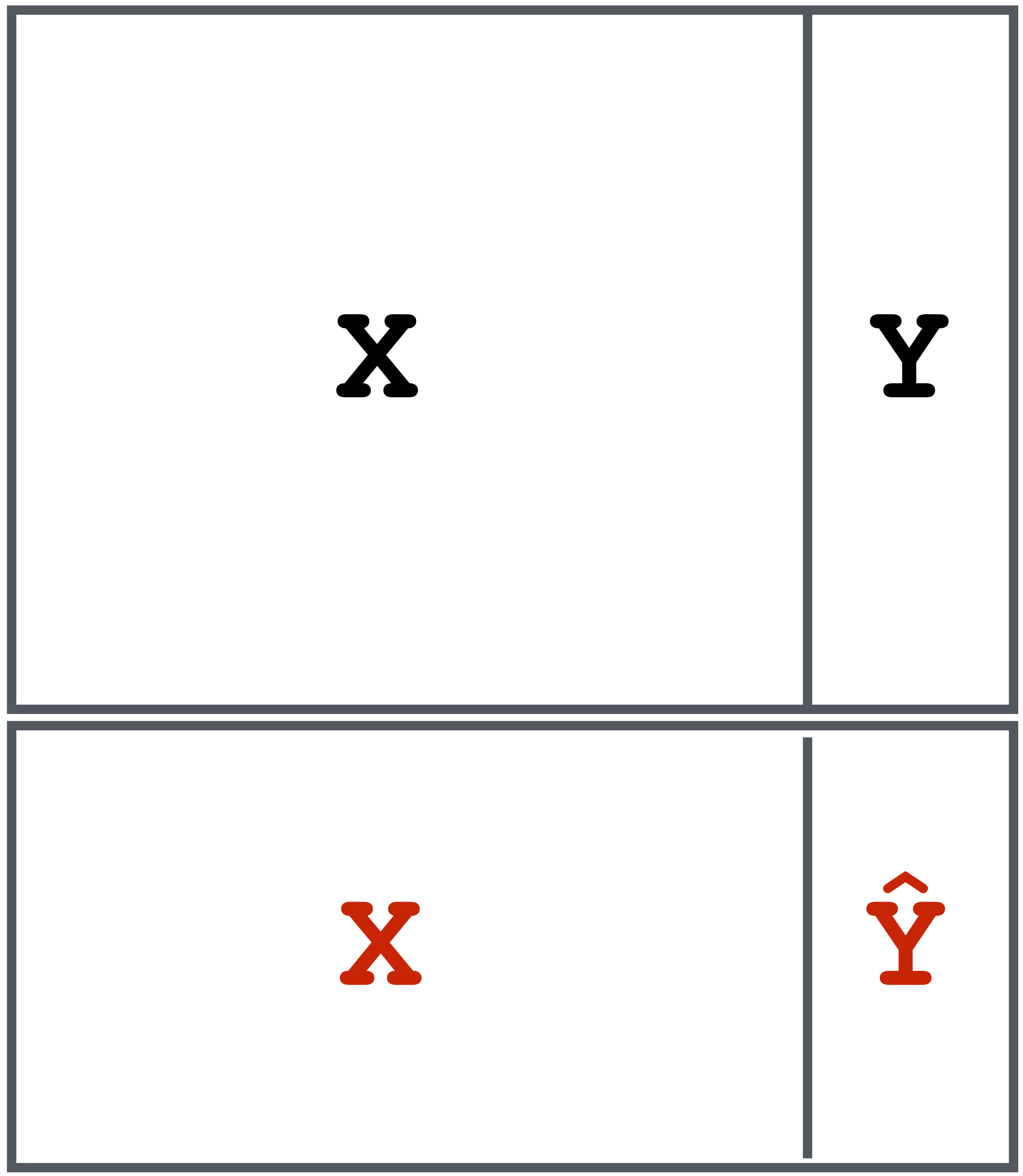
Tagging: use of dictionaries, hyperlinks…

# Other SSL approaches

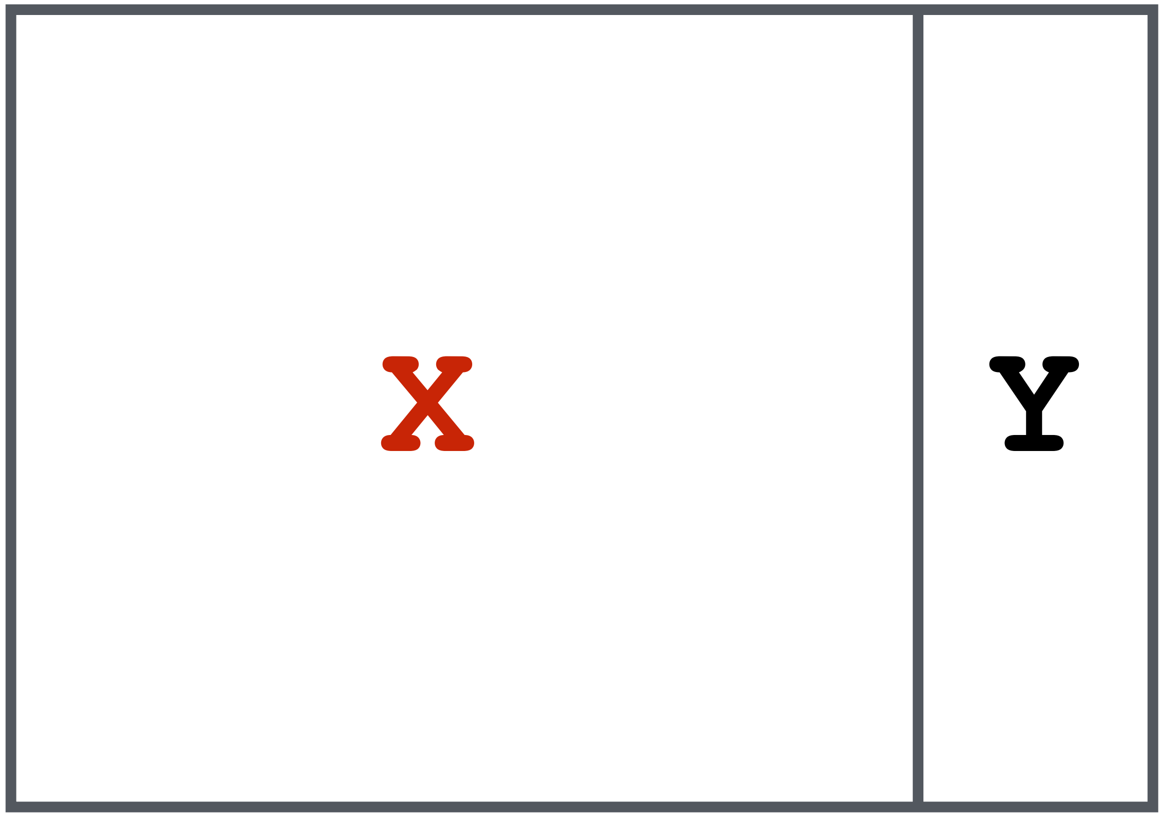
- **Co-training**
  - similar to self-training but with *two views (*two classifiers labeling data for each other
    - ✓ often less sensitive to mistakes
    - – computationally more expensive (ensemble)
- **Tri-training**
  - add data if two classifiers agree on label

What has this to do with generalization?

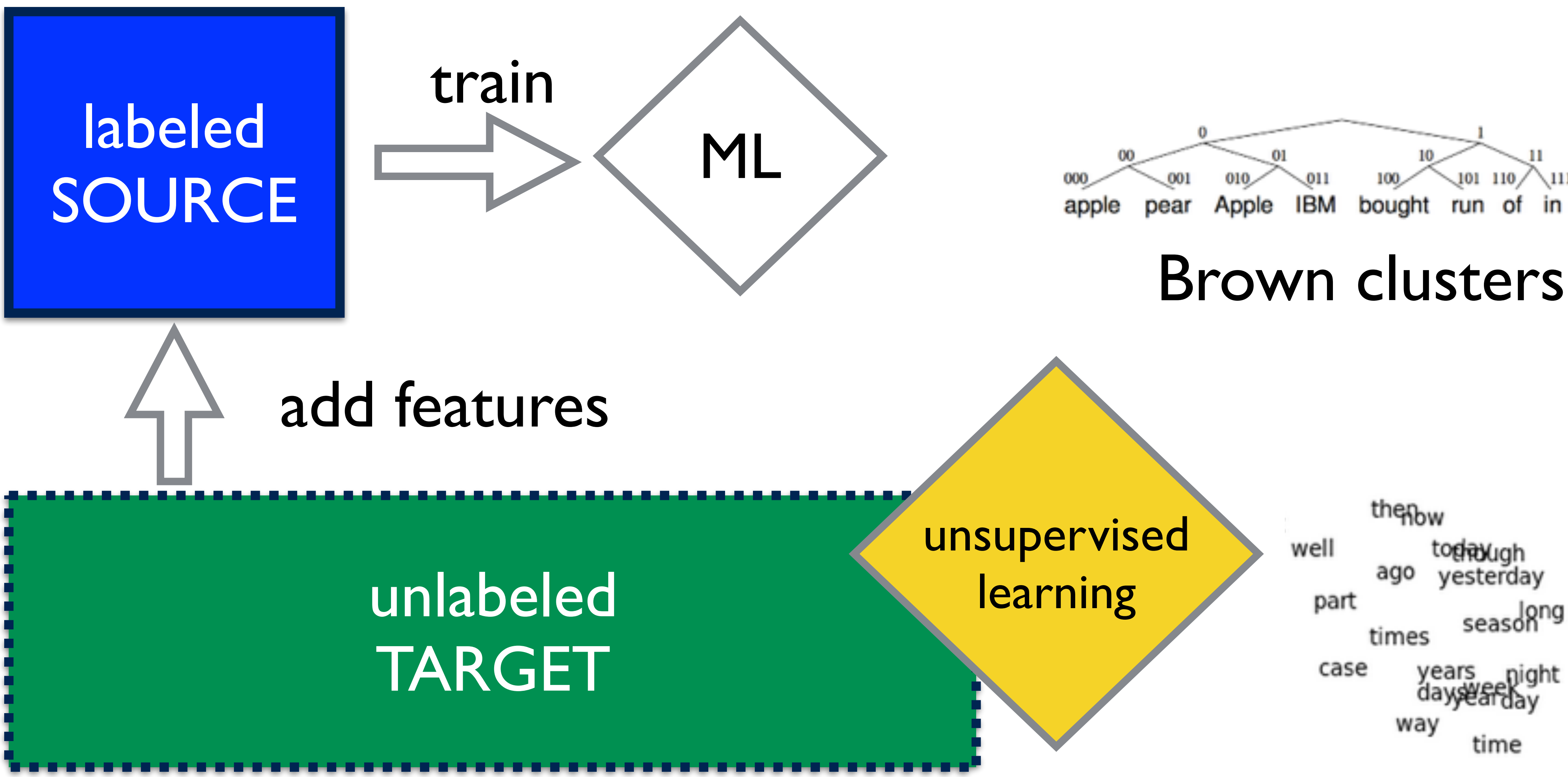
# so far: more X



# What about modifying X?



# Implicit use of unlabeled data




(e.g., Turian et al., 2010; Mikolov et al., 2013; Baroni et al., 2014)

# Example: Brown clusters

| | | |
|---|---|---|
| 11011000011 | consumers | 76394 |
| 11011000011 | employers | 119946 |
| 11011000011 | residents | 126880 |
| 1101011011001 | citizen | 21543 |
| 1101011011001 | photographer | 22341 |
| 1101011011001 | legend | 24429 |
| 1101011011001 | priest | 24698 |
| 1101011011001 | farmer | 25568 |
| 1101011011001 | lawyer | 25588 |
| 1101011011001 | journalist | 26313 |
| 1101011011001 | filmmaker | 3919 |

# Add features to X

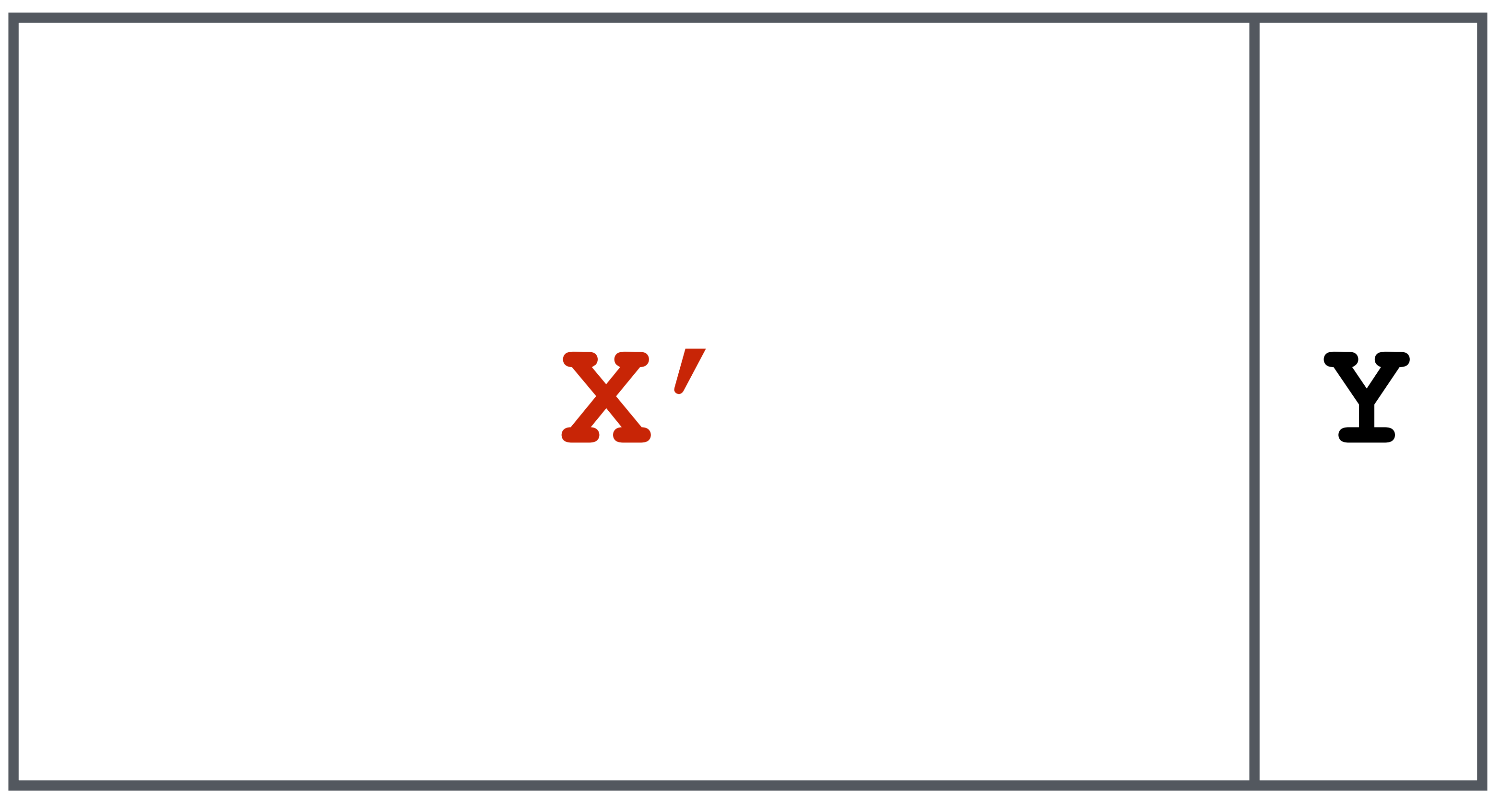
- add features learned from unlabeled data
- Hypothesis: additional features will help to bridge the gap between source and target
- shared feature representation is the idea behind structural correspondence learning (SCL)

**powerful** **boring**
**defective**
**fascinating**

**excellent**
**good**

blue: source
green: target
black: in both domains

# Add features to X

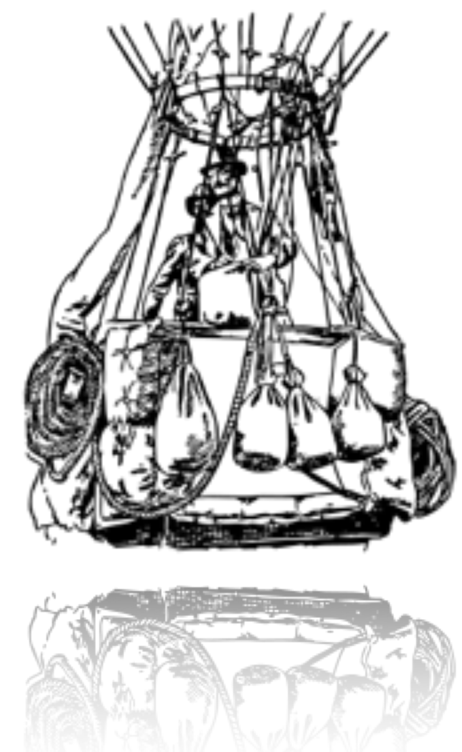


# Possible approaches

1. More **X**:
   a. Use semi-supervised learning (self-training, co-training, tri-training)
2. Modify **X**:
   a. Add features: embeddings, clusters

WHAT ELSE CAN WE DO TO CORRECT SAMPLE BIAS?

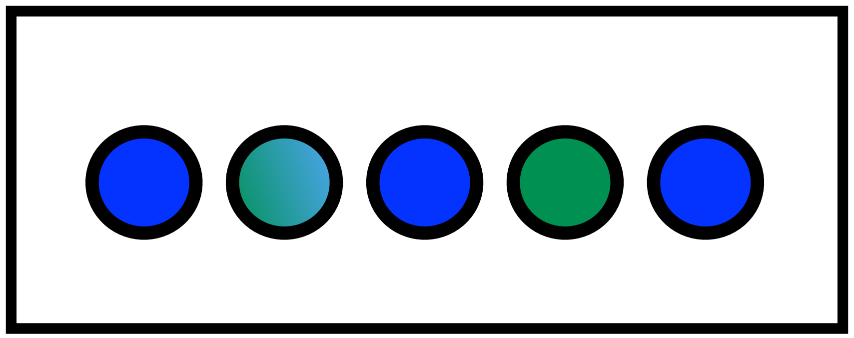
**Drop!**
- data points
- features



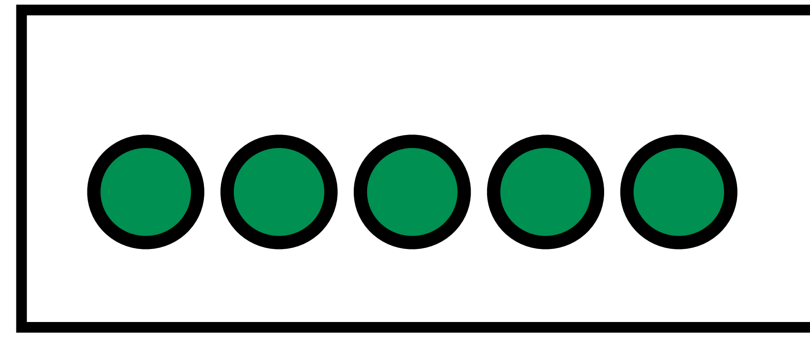
# Importance weighting

# Importance weighting (IW)

SOURCE train



TARGET test



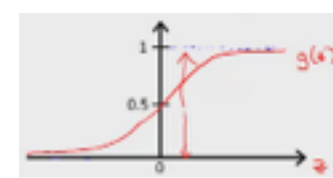
assign instance-dependent weights (Shimodaira, 2001):

?

$$\frac{P_T(\mathbf{x})}{P_S(\mathbf{x})}$$

unlabeled TARGET

*approximation, e.g.:*



domain classifier to discriminate between SOURCE & TARGET

(Zadrozny et al., 2004; Bickel and Scheffer, 2007; Søgaard and Haulrich, 2011)

# Importance weighting (IW)

**Pros**

✓ simple idea

✓ works well if we know how our sample differs

✓ also useful to combat label bias (more on this later)

**Cons**

- challenge is to find a good weight function
- finite sample: can overcome bias only to certain extent

# Importance weighting in NLP

Only 4 NLP studies[1], of which 2 on unsupervised DA with mixed results

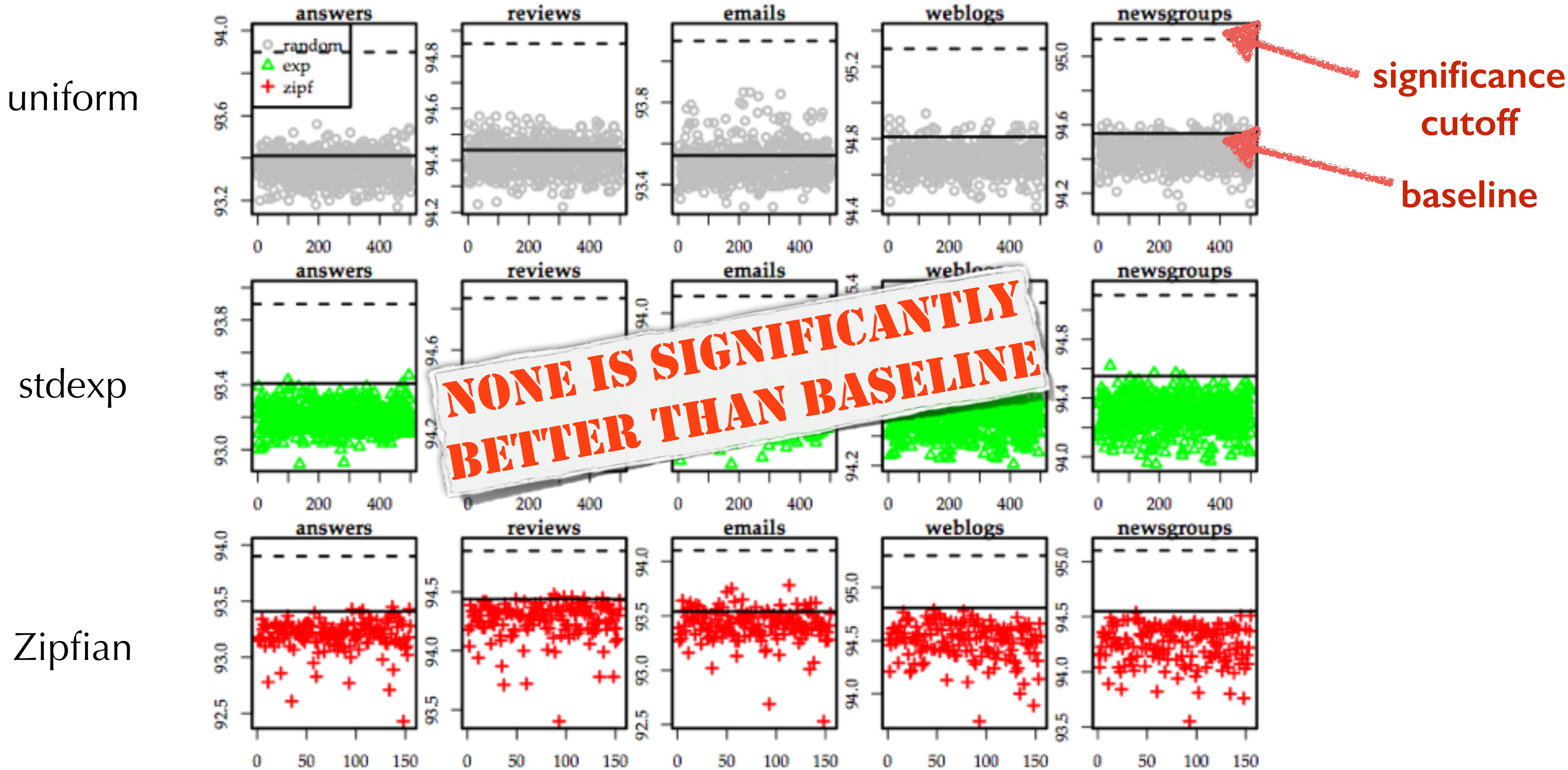
*Does importance weighting work for unsupervised DA of POS taggers?*

[1] (Jiang & Zhai, 2007; Foster et al., 2010; Søgaard & Haulrich, 2011; Plank & Moschitti, 2013)

We tried many ways (different ways to get weights), but..

(Plank, Johannsen, Søgaard, 2014) EMNLP

# Importance weighting for POS

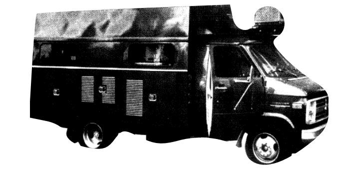



(500 runs in each plot)

# Possible approaches

1. More **X**:
   a. Use semi-supervised learning (self-training, co-training, tri-training)
2. Modify **X**:
   a. Add features: embeddings, clusters
   b. Use only some
      a. instances: importance weighting
      b. features: dropout

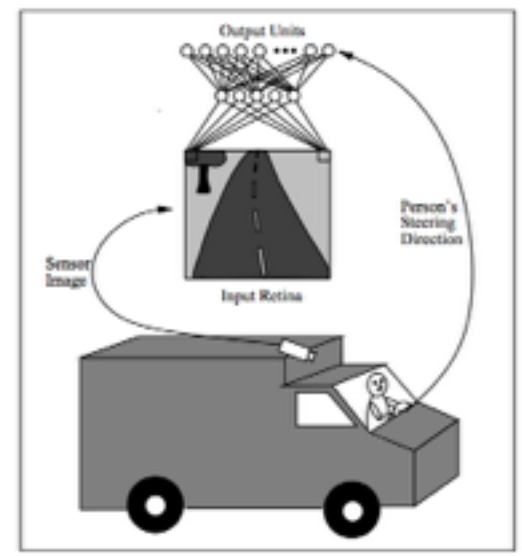
# Dropout



# Feature swamping

Motivation:

Figure 1: *The CMU Navlab Autonomous Navigation Testbed*

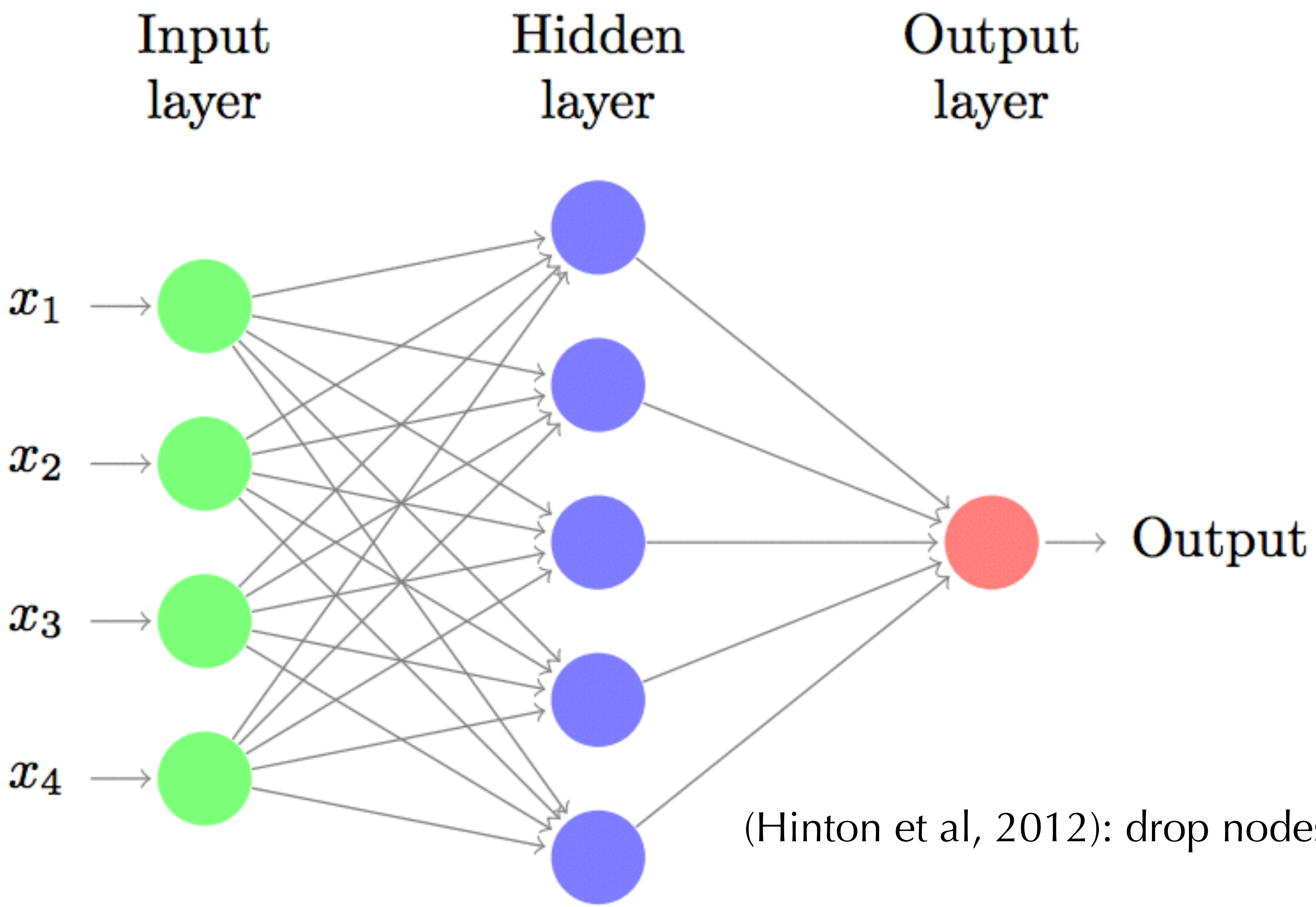



(ALVINN)

**Problem:** feature swamping (Sutton et al. 2006)

**Idea:** corrupt features

# Neural Network




(Hinton et al, 2012): drop nodes in NN!

# Data Corruption

Original

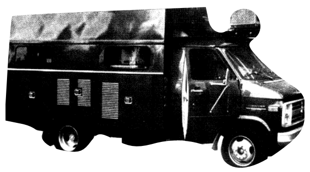
| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | 1 | 1 | | | | | 1 | | | | | 1 |

Corrupted data

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | 1 | 1 | | | | | 1 | | | | | 1 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | 1 | 1 | | | | | 1 | | | | | 1 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | 1 | 1 | | | | | 1 | | | | | 1 |



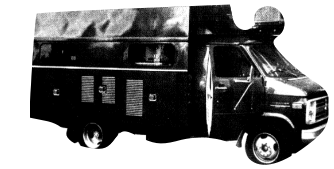
# Dropout

**Algorithm 1** Averaged perceptron with drop-out

**input:** dataset $\mathcal{D}$ of size $M \times N$, number of rounds $R$, distribution $P$, drop-out rate $\delta = 0.1$
initialize $t = 0$, $\boldsymbol{w}^t = \mathbf{0}$
**for** $r = 1$ **to** $R$ **do**
  **for** $(x^i, y^i)$ **in** $\mathcal{D}$ **do**
    draw active: $\xi = P(1 - \delta, M)$
    predict: $\hat{y} \leftarrow \arg\max_{y' \in \mathcal{Y}} \boldsymbol{w}^t \cdot \xi[\boldsymbol{f}(x^i, y')]$
    update the model: $\boldsymbol{w}^{t+1} \leftarrow \boldsymbol{w}^t + \xi[\boldsymbol{f}(x^i, y^i) - \boldsymbol{f}(x^i, \hat{y})]$
    $t = t + 1$
  **end for**
**end for**
**output:** the averaged model $\hat{\boldsymbol{w}} \leftarrow \frac{1}{t} \sum_{i=1}^{t} \boldsymbol{w}^i$

$\xi$ vector indicating how "active" feature is

- **binomial dropout** (Søgaard & Johannsen, 2012): sample P from random binomial ("hard dropout", 0/1)
- **Antagonistic adversaries** (Søgaard, 2013a): drop features "where it hurts most" (those that get weight more than standard deviation away from mean)



# Another view on dropout

*Ensemble methods* (e.g., NetFlix challenge)

dropout

~ model averaging

~ regularization

(Hinton et al., 2012; Wager, Wang & Liang, 2013)

# What has dropout to do with generalization?

# Possible approaches

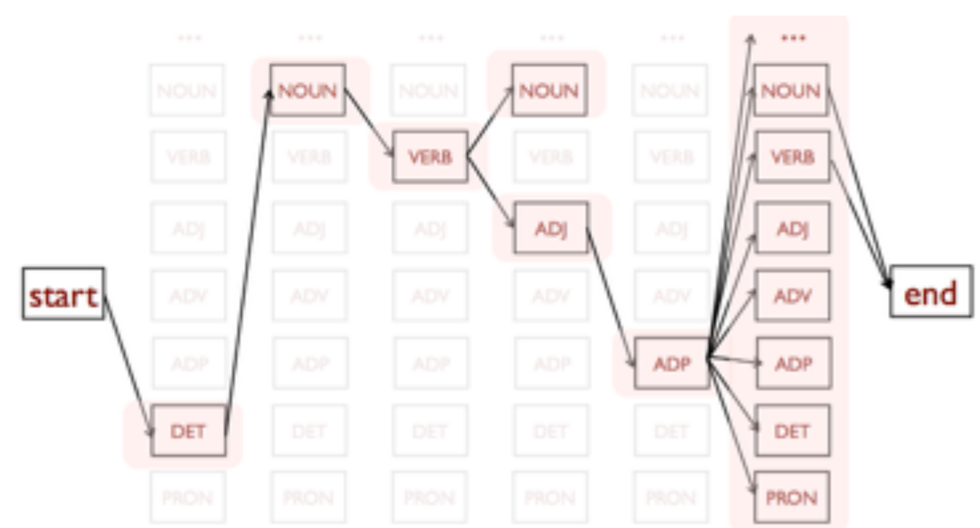
1. More **X**:
   a. Use semi-supervised learning (self-training, co-training, tri-training)
2. Modify **X**:
   a. Add features: embeddings, clusters
   b. Drop (weight) instances: importance weighting
   c. Drop features: dropout
3. (Use additional knowledge to guide learner): distant supervision

distant supervision

# Distant supervision

- **Distantly supervised:** use a large knowledge base (KB) to create noisily labeled instances
- **Idea:** if *entity1* and *entity2* are found in the same sentence and *rel(entity1,entity2)* ∈ KB → positive training instance
- Exploiting some kind of "world knowledge"
- Like **type-constraints** in sequence tagging

  (Täckström et al., 2013)




The food is good at COLING

# Take-home message

| | **Good?** | **Bad?** |
|---|---|---|
| **Semi-supervised learning** | Neighboring domains. Or with *distant* supervision. | When the CROSS-DOMAIN GULF is wide. |
| **Importance-weighting** | ? (in generative models) | In discriminative POS tagging, for example. |
| **Dropout** | When your training data is highly redundant, e.g. text classification. | In parsing, for example. |
| **Distant supervision** | When your KBs are good. | For low-resource languages. |

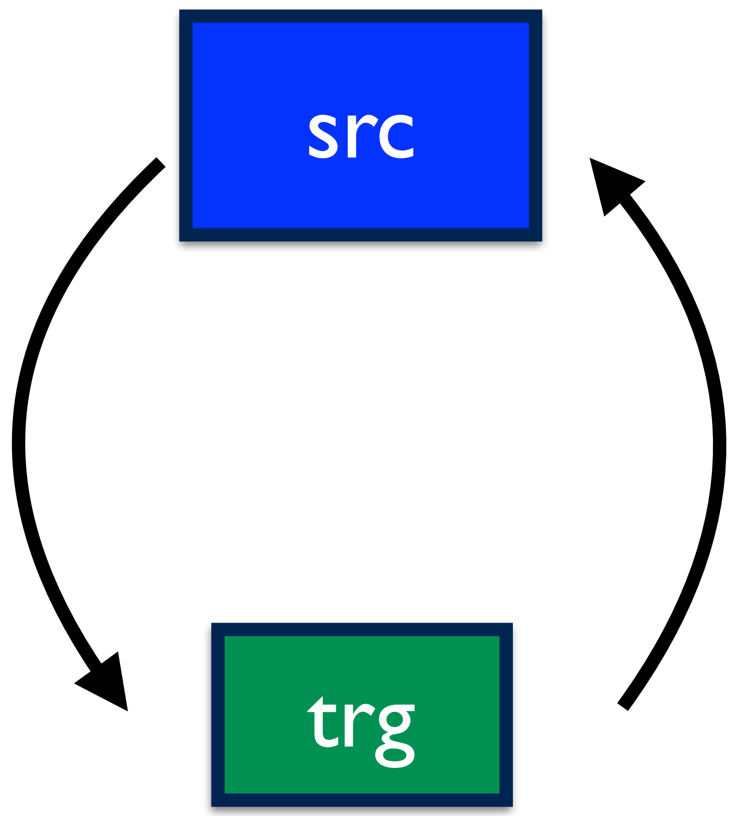
# Additional

# Baselines (for supervised DA)

| | Train on | Test on TRG |
|---|---|---|
| SRC only | src | |
| TRG only | trg | |
| UNION | src + trg | |

# Making input/training data more similar to each other

change input to
resemble src
(e.g., Aw 2006 for parsing,
Tim Baldwin's work on normalization..)




change src to
resemble trg
(e.g., van der Plas et al., 2009)

Questions?

Thanks!