

# **Semi-Supervised Learning (SSL): Transductive SVM & Co-Training**

# Different Types of Learning

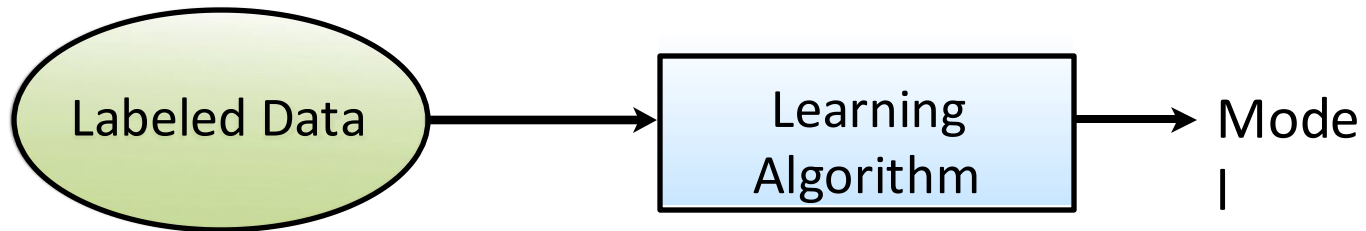
**Supervised learning** : learn from **labeled** data

**Unsupervised learning** : learn from **unlabeled** data

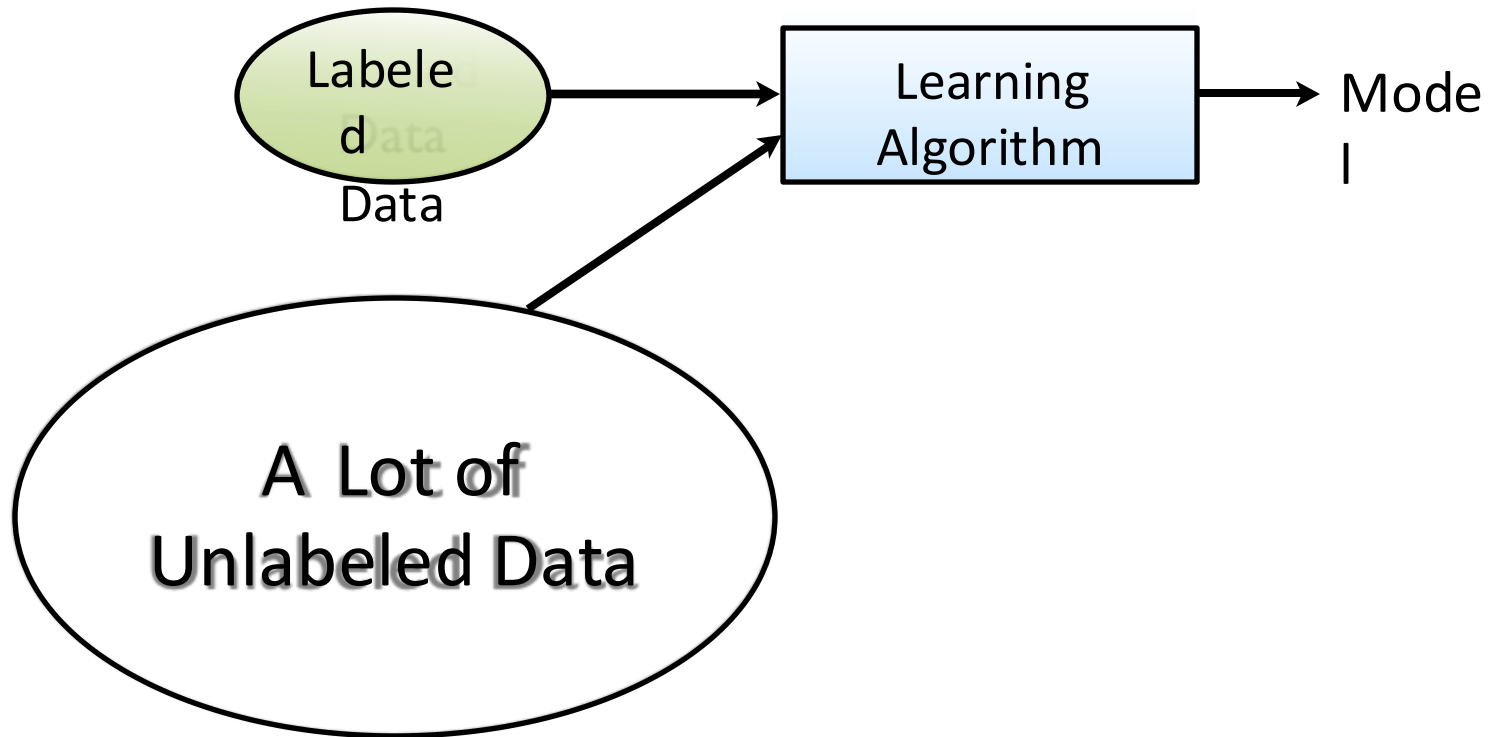
**Semi-supervised learning** : learn from both **labeled** and **unlabeled** data

| usage                     | supervised learning | semi-supervised learning | unsupervised learning |
|---------------------------|---------------------|--------------------------|-----------------------|
| $\{(x, y)\}$ labeled data | yes                 | yes                      | no                    |
| $\{x\}$ unlabeled data    | no                  | yes                      | yes                   |

# Supervised Learning



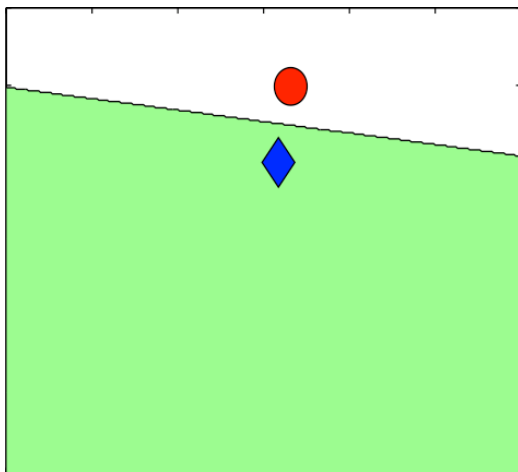
# Semi-Supervised Learning (SSL)



# Why SSL?

- Labeling is expensive and difficult
  - Human annotation is slow
  - Labels require experts
  - Imagine the cost/time of tagging millions of images!
- Unlabeled data
  - Cheap and sufficient

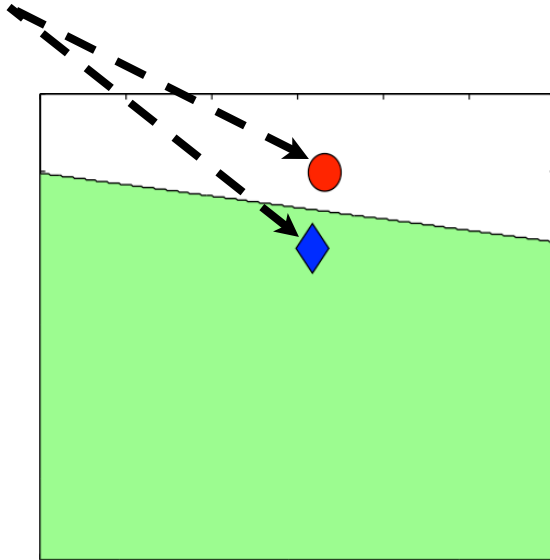
How can unlabeled data be helpful?



Without Unlabeled Data

How can unlabeled data be helpful?

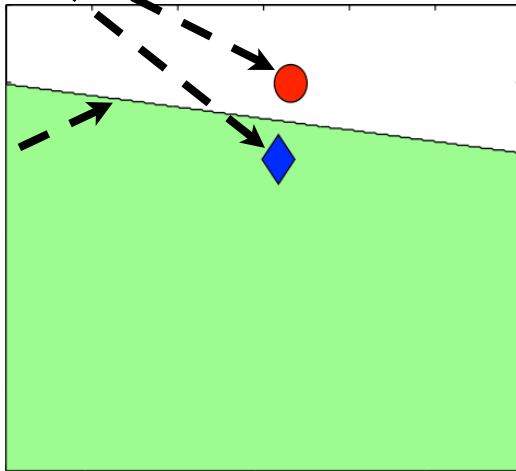
Labeled  
Instances



Without Unlabeled Data

# How can unlabeled data be helpful?

Labeled  
Instances



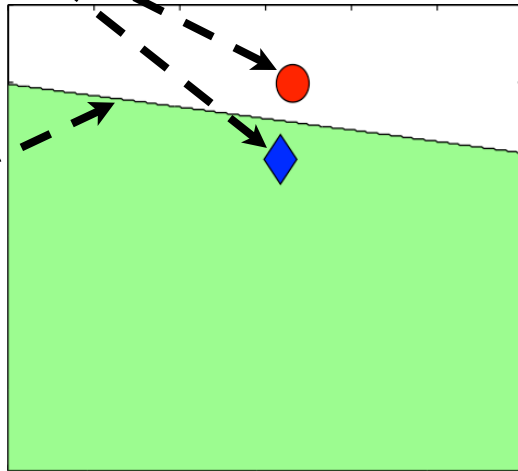
Decision  
Boundary

Without Unlabeled Data

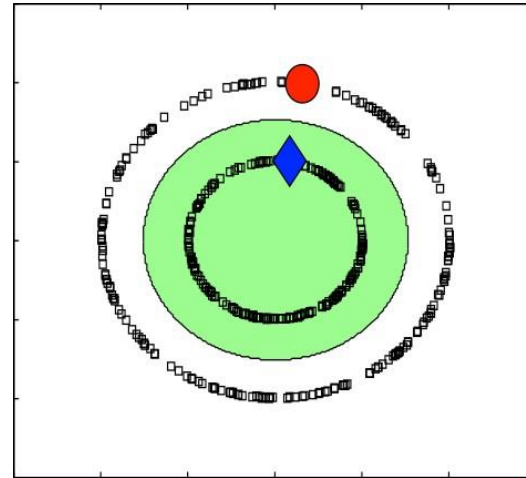


# How can unlabeled data be helpful?

Labeled  
Instance  
s

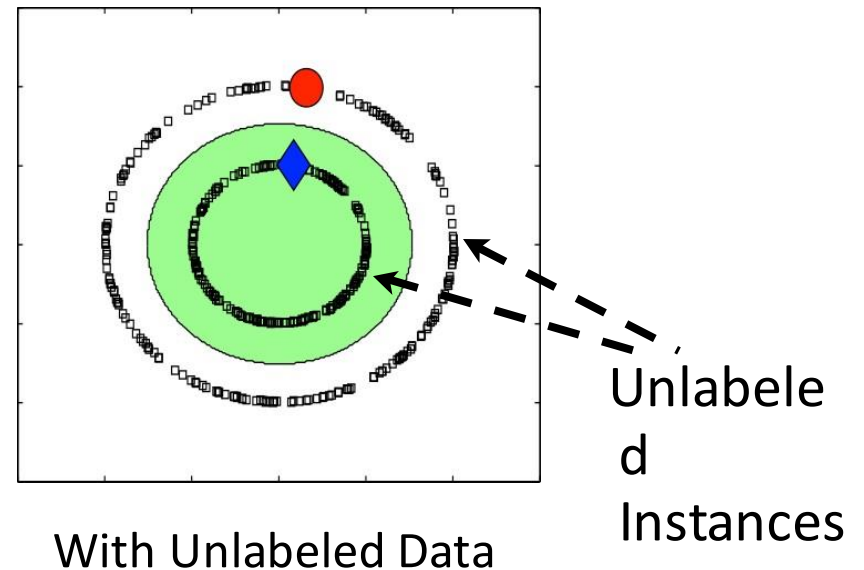
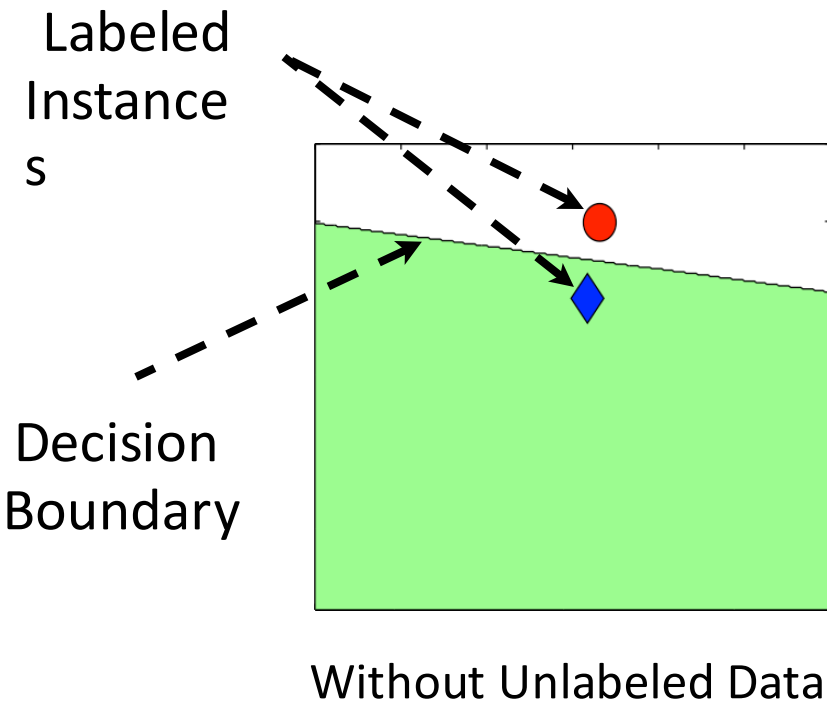


Without Unlabeled Data

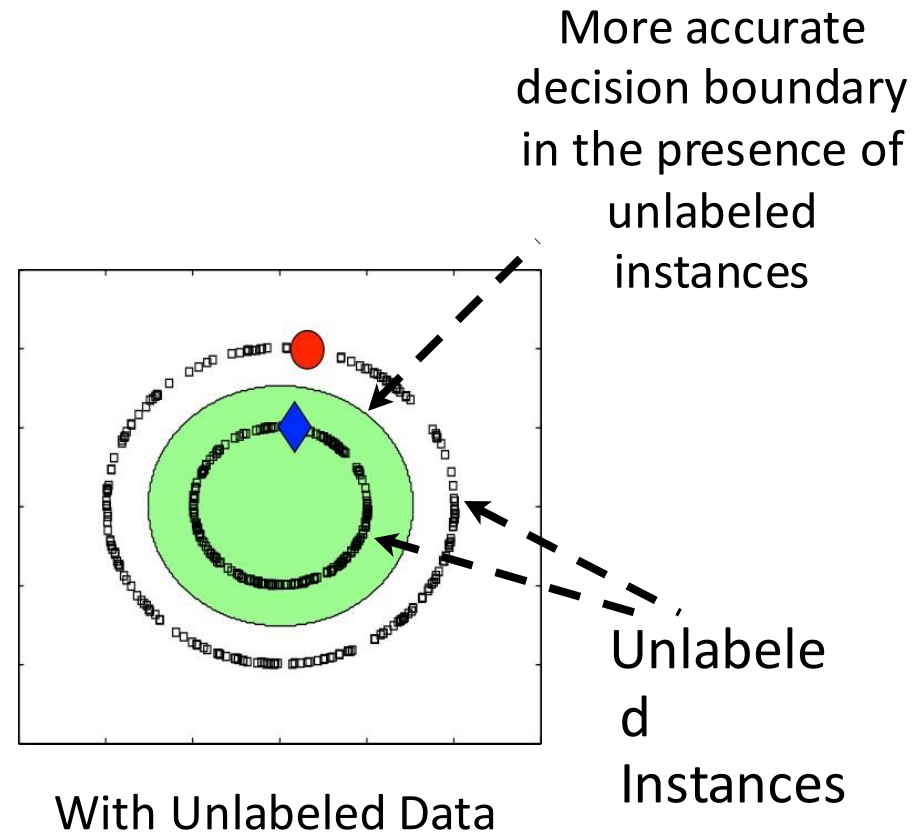
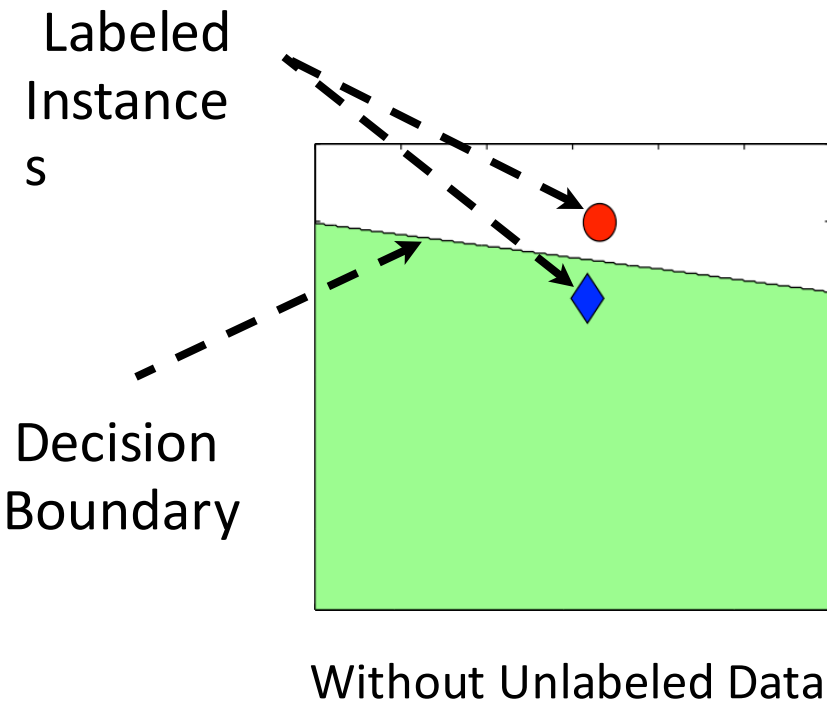


With Unlabeled Data

# How can unlabeled data be helpful?

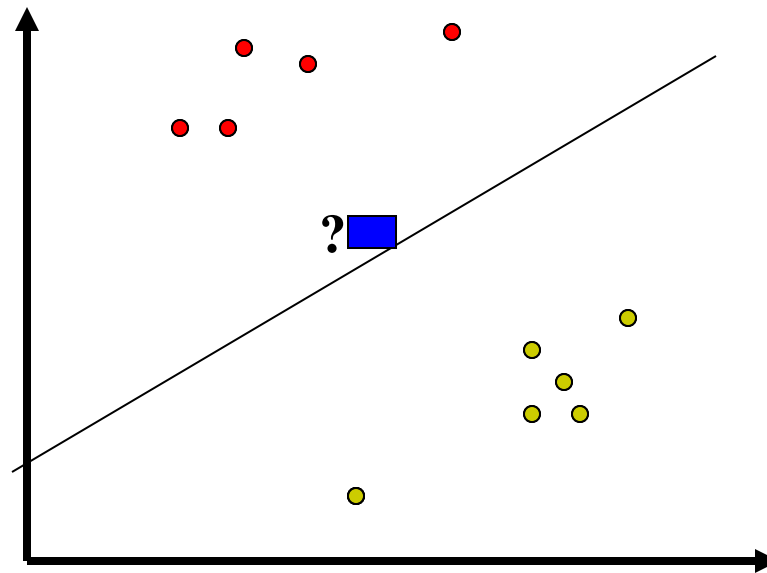


# How can unlabeled data be helpful?

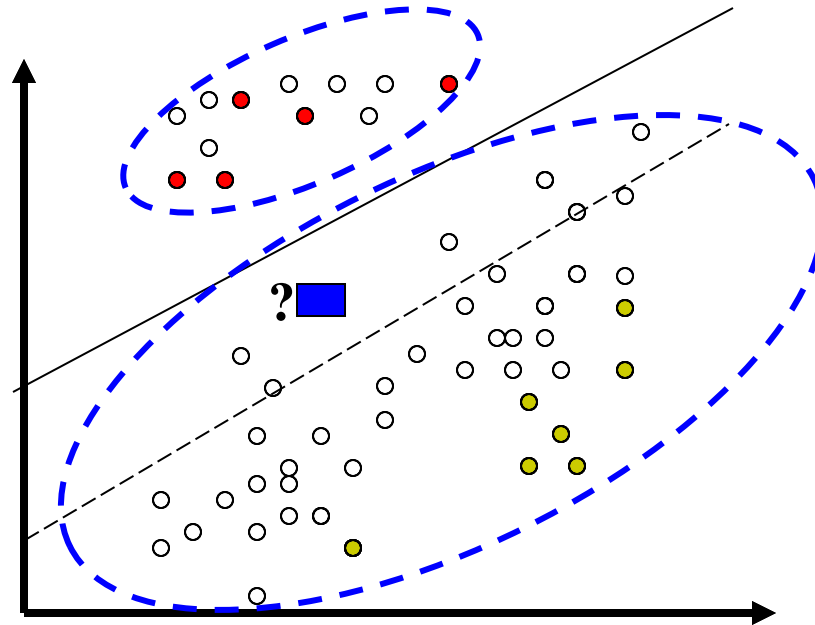


Example from [Belkin et al., JMLR 2006]

# Clustering Assumption



# Clustering Assumption



- Points with same label are connected through high density regions, thereby defining a cluster
- Clusters are separated through **low-density regions**

# Inductive vs Transductive Learning

- Inductive learning
  - Induce a decision function that works well for *all the possible examples*.
- Transductive learning
  - Find a decision function that works well for *the given test examples*
  - Problem setting
    - Given labeled data and unlabeled data
    - Find set of labels that best fit unlabeled data
    - Note that we do not extend to unseen examples

# SSL Methods

- Non-graph based SSL methods
  - Transductive SVM
  - Co-training
  - Active learning
  - MixMatch
- Graph based SSL methods
  - Label propagation
  - Belief propagation
  - Graph neural networks

# **Transductive Support Vector Machine (TSVM)**



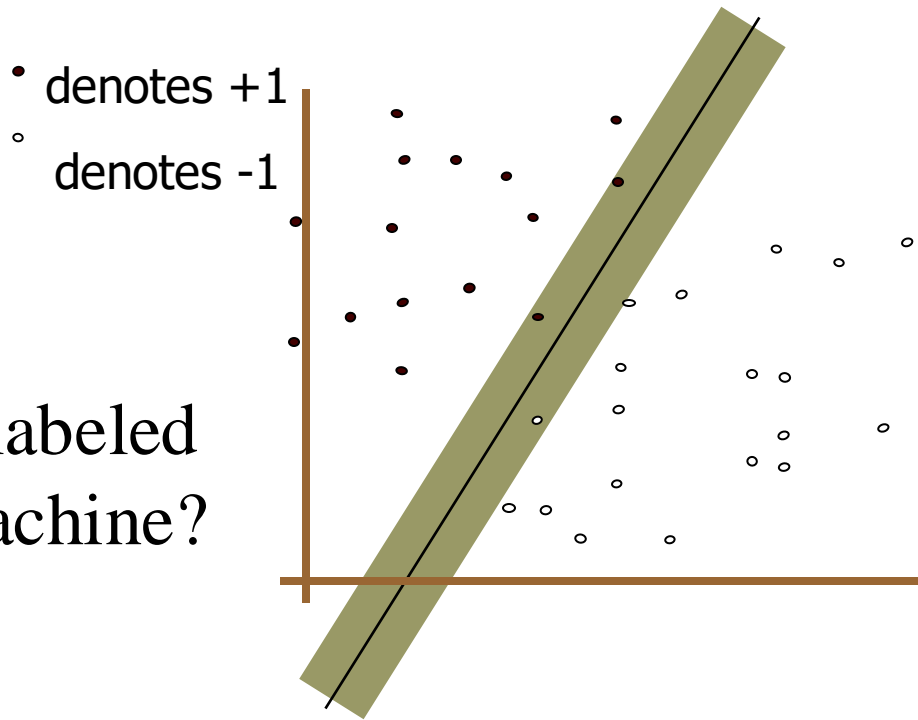
# Cluster Assumption vs. Maximum Margin

- Support Vector Machine (SVM)
  - Maximum margin classifier

→ low density around decision boundary

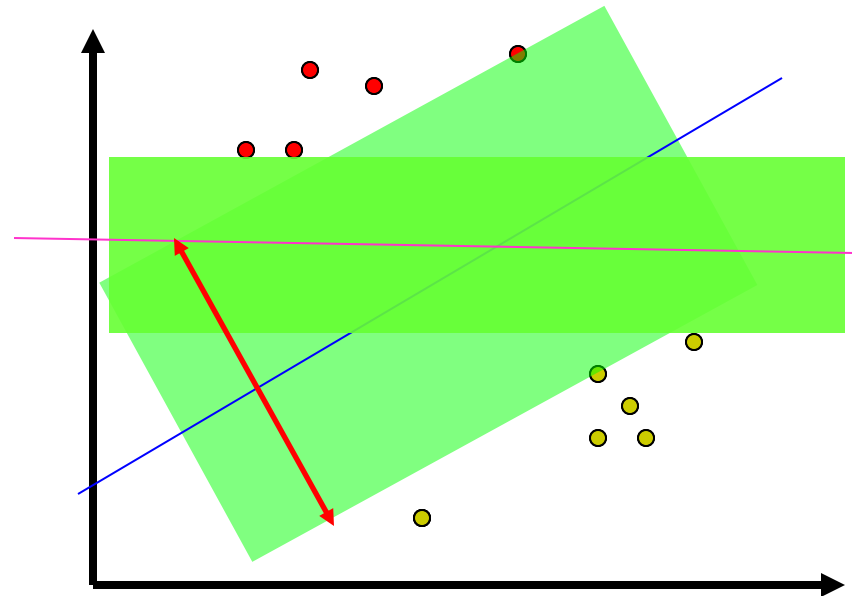
→ Cluster assumption

- What about using the unlabeled data in support vector machine?



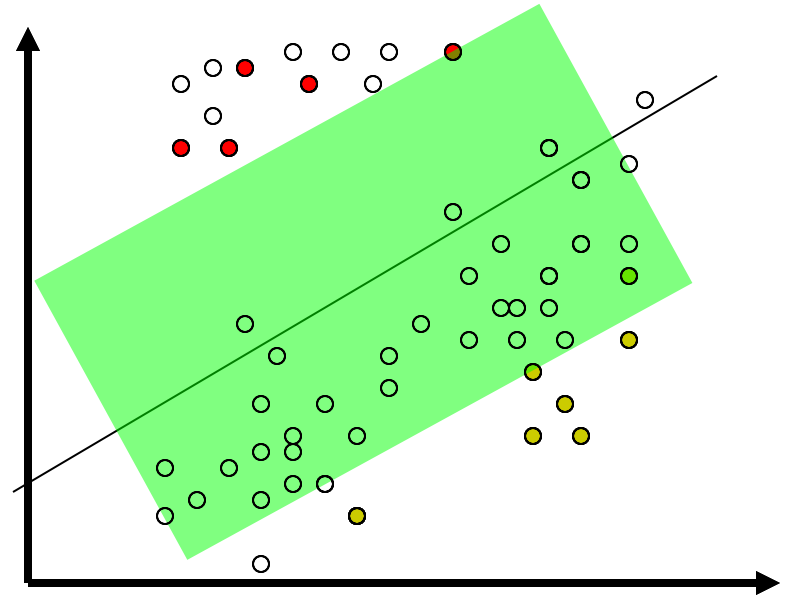
# Transductive SVM

- Decision boundary given a small number of labeled examples
- Support vector machine
  - Maximum margin classifier



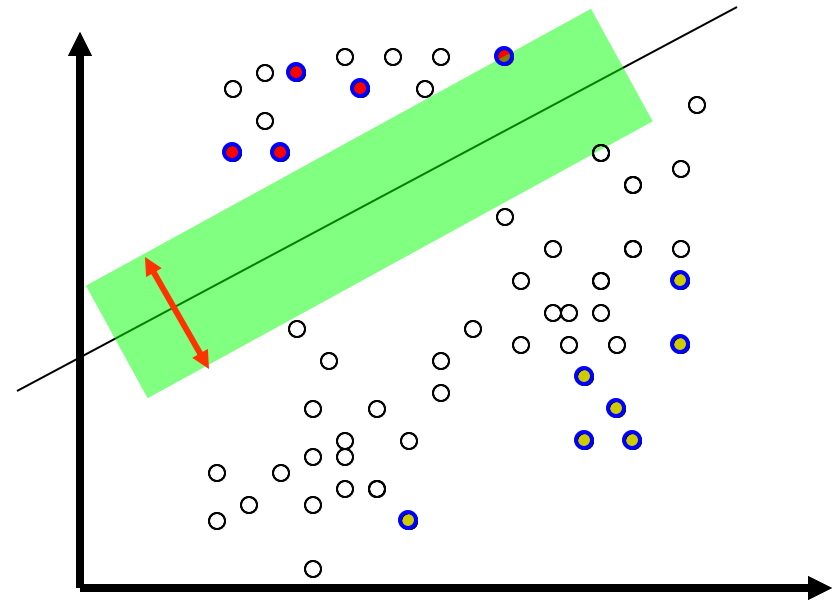
# Transductive SVM

- Decision boundary given a small number of labeled examples
- How to change decision boundary given both labeled and unlabeled examples ?



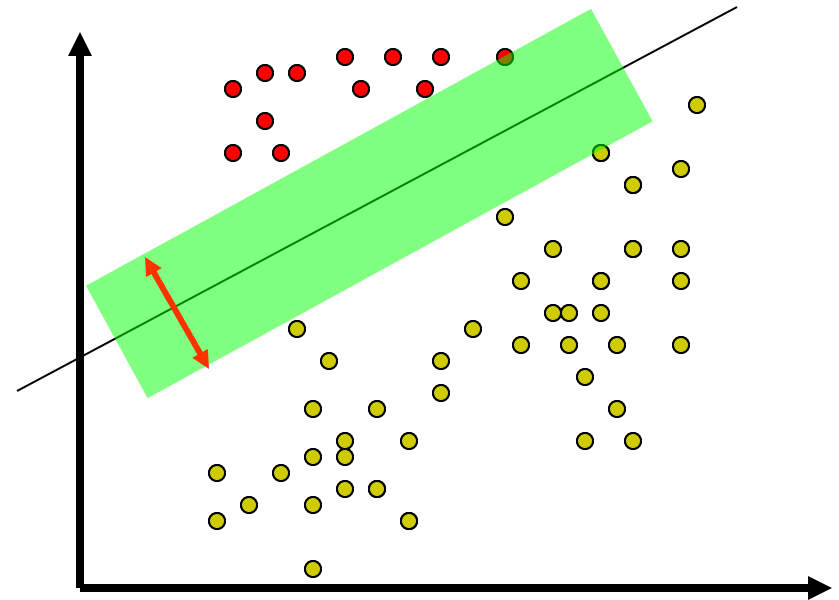
# Transductive SVM

- Decision boundary given a small number of labeled examples
- Move the decision boundary to place with low density
  - Maximum margin



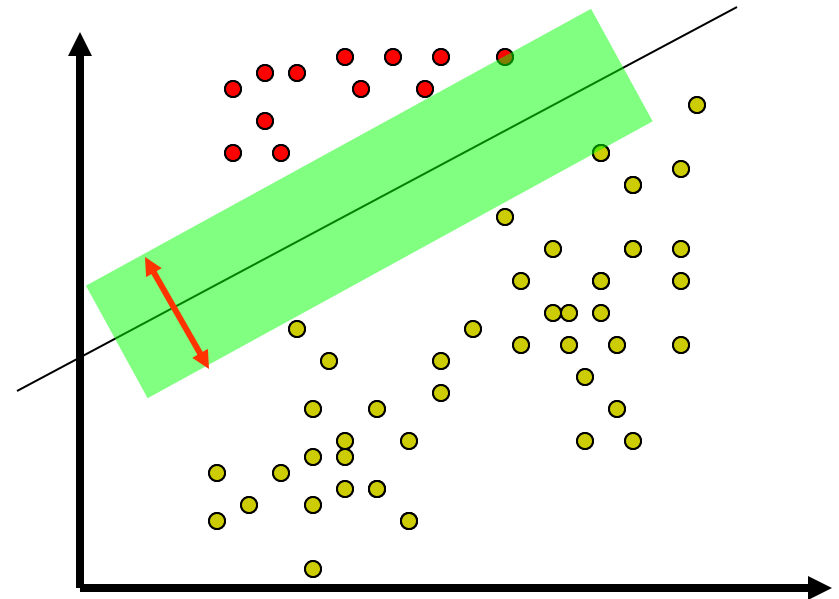
# Transductive SVM

- Decision boundary given a small number of labeled examples
- Move the decision boundary to place with low density
- Classification results



# Transductive SVM

- Decision boundary given a small number of labeled examples
- Move the decision boundary to place with low density
- Classification results
- How to formulate this idea?



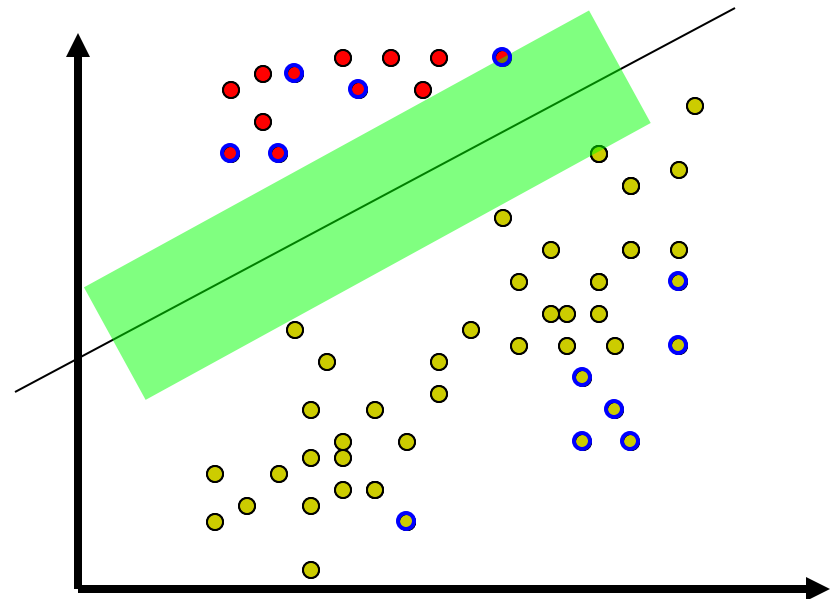
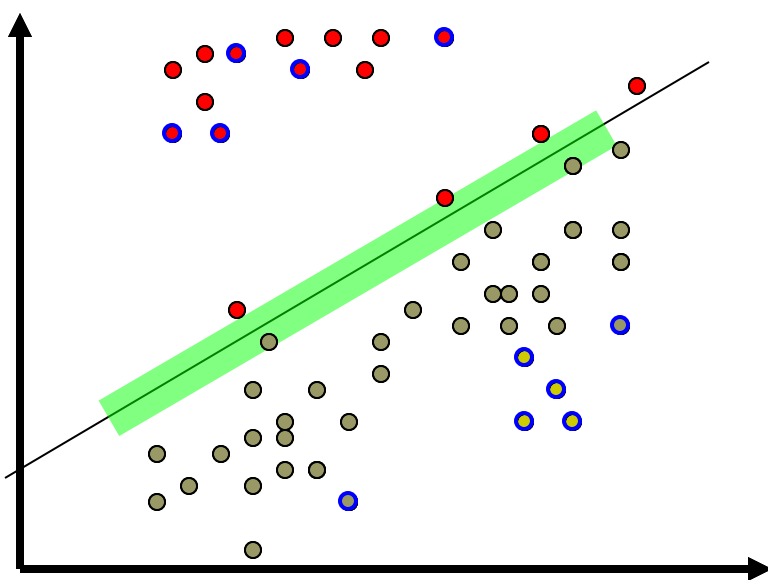
# Transductive SVM: Formulation

- Labeled data L:  $L = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$
- Unlabeled data D:  $D = \{(x_{n+1}), (x_{n+2}), \dots, (x_{n+m})\}$
- Maximum margin principle for *mixture of labeled and unlabeled data*
  - **Step I:** Given label assignment of each unlabeled data, compute its maximum margin
  - **Step II:** Given the maximum margin of unlabeled data, update the label assignment

# Transductive SVM

Different label assignment for unlabeled data

→ different maximum margin





# Transductive SVM: Formulation

**Original SVM**

A binary variables for  
label of each example

**Transductive SVM**

$$\{\mathbf{w}^*, b^*\} = \underset{\mathbf{w}, b}{\operatorname{argmin}} \mathbf{w} \cdot \mathbf{w}$$

$$\left. \begin{array}{l} y_1 \left( \mathbf{w} \cdot \mathbf{x}_1 + b \right) \geq 1 \\ y_2 \left( \mathbf{w} \cdot \mathbf{x}_2 + b \right) \geq 1 \\ \dots \\ y_n \left( \mathbf{w} \cdot \mathbf{x}_n + b \right) \geq 1 \end{array} \right\} \begin{array}{l} \text{labeled} \\ \text{examples} \end{array}$$

Constraints for  
unlabeled data

$$\{\mathbf{w}^*, b^*\} \Rightarrow \underset{y_{n+1}, \dots, y_{n+m}}{\operatorname{argmin}} \underset{\mathbf{w}, b}{\operatorname{argmin}} \mathbf{w} \cdot \mathbf{w}$$

$$\left. \begin{array}{l} y_1 \left( \mathbf{w} \cdot \mathbf{x}_1 + b \right) \geq 1 \\ y_2 \left( \mathbf{w} \cdot \mathbf{x}_2 + b \right) \geq 1 \\ \dots \\ y_n \left( \mathbf{w} \cdot \mathbf{x}_n + b \right) \geq 1 \end{array} \right\} \begin{array}{l} \text{labeled} \\ \text{examples} \end{array}$$

$$\left. \begin{array}{l} y_{n+1} \left( \mathbf{w} \cdot \mathbf{x}_{n+1} + b \right) \geq 1 \\ \dots \\ y_{n+m} \left( \mathbf{w} \cdot \mathbf{x}_{n+m} + b \right) \geq 1 \end{array} \right\} \begin{array}{l} \text{unlabeled} \\ \text{examples} \end{array}$$

# Introducing Slack Variables

$$\{\mathbf{w}^*, b^*\} = \underset{y_{n+1}, \dots, y_{n+m}}{\operatorname{argmin}} \underset{\mathbf{w}, b}{\operatorname{argmin}} \mathbf{w} \cdot \mathbf{w} + \boxed{\sum_{i=1}^n \xi_i} + \boxed{\sum_{i=1}^n \eta_i}$$

$$\left. \begin{array}{l} y_1 \left( \mathbf{w} \cdot \mathbf{x}_1 + b \right) \geq 1 - \xi_1 \\ y_2 \left( \mathbf{w} \cdot \mathbf{x}_2 + b \right) \geq 1 - \xi_2 \\ \dots \\ y_n \left( \mathbf{w} \cdot \mathbf{x}_n + b \right) \geq 1 - \xi_n \end{array} \right\} \begin{array}{l} \text{labeled} \\ \text{examples} \end{array} \quad \left. \begin{array}{l} y_{n+1} \left( \mathbf{w} \cdot \mathbf{x}_{n+1} + b \right) \geq 1 - \eta_1 \\ \dots \\ y_{n+m} \left( \mathbf{w} \cdot \mathbf{x}_{n+m} + b \right) \geq 1 - \eta_m \end{array} \right\} \begin{array}{l} \text{unlabeled} \\ \text{examples} \end{array}$$

- No longer convex optimization problem
- How to optimize transductive SVM?
  - Alternating optimization

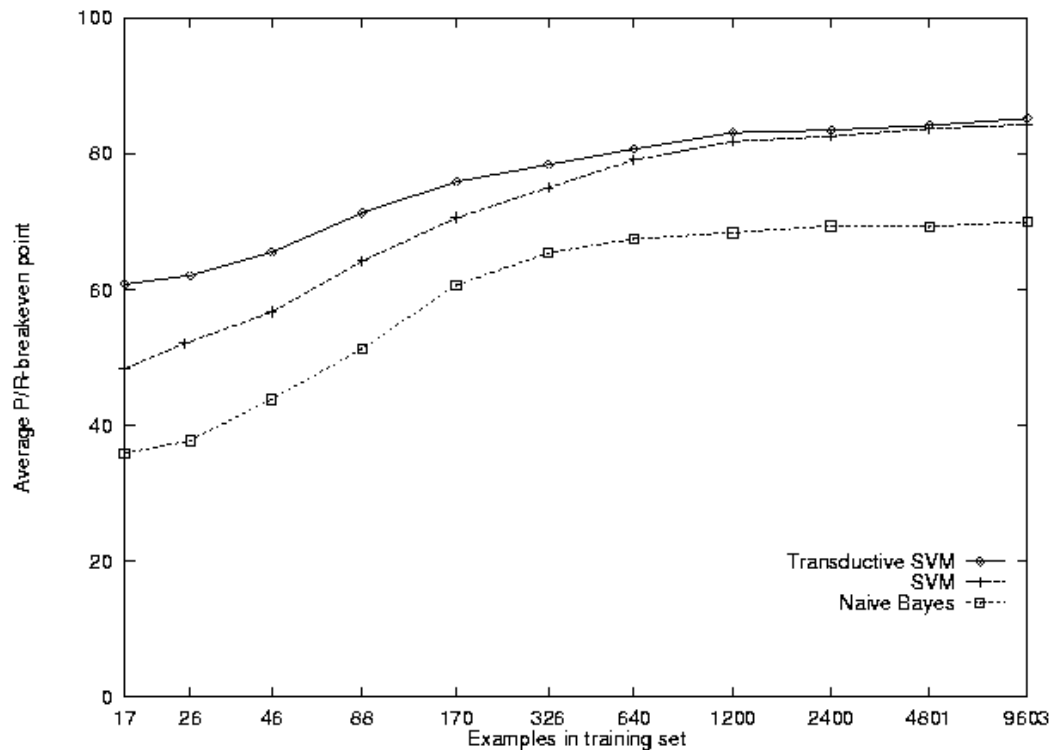
# Alternating Optimization

$$\{\mathbf{w}^*, b^*\} = \underset{y_{n+1}, \dots, y_{n+m}}{\operatorname{argmin}} \underset{\mathbf{w}, b}{\operatorname{argmin}} \mathbf{w} \cdot \mathbf{r} + \sum_{i=1}^n \xi_i + \sum_{i=1}^n \eta_i$$

$$\left. \begin{array}{l} y_1 \left( \mathbf{w} \cdot \mathbf{x}_1 + b \right) \geq 1 - \xi_1 \\ y_2 \left( \mathbf{w} \cdot \mathbf{x}_2 + b \right) \geq 1 - \xi_2 \\ \dots \\ y_n \left( \mathbf{w} \cdot \mathbf{x}_n + b \right) \geq 1 - \xi_n \end{array} \right\} \begin{array}{l} \text{labeled} \\ \text{examples} \end{array} \quad \left. \begin{array}{l} y_{n+1} \left( \mathbf{w} \cdot \mathbf{x}_{n+1} + b \right) \geq 1 - \eta_1 \\ \dots \\ y_{n+m} \left( \mathbf{w} \cdot \mathbf{x}_{n+m} + b \right) \geq 1 - \eta_m \end{array} \right\} \begin{array}{l} \text{unlabeled} \\ \text{examples} \end{array}$$

- Step 1: fix  $y_{n+1}, \dots, y_{n+m}$ , learn weights  $\mathbf{w} = (\bar{\mathbf{w}}, b)$
- Step 2: fix weights  $\mathbf{w}$ , predict  $y_{n+1}, \dots, y_{n+m}$

# Text Classification Results (Joachims 99)



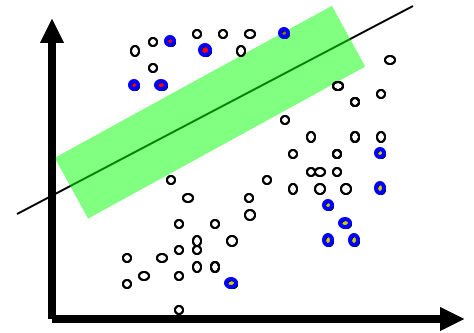
10 categories from  
the Reuter collection

3299 test documents

1000 informative words  
selected by MI criterion

# Summary

- Based on maximum margin principle
- Classification margin is decided by
  - Labeled data
  - Unlabeled data with assigned labels
- High computational cost
  - Variants: Low Density Separation (LDS), Semi-Supervised Support Vector Machine (S3VM)



# Co-Training

# Co-training [Blum & Mitchell, 1998]

- Classify web pages into
  - category for students and category for professors
- Two views of web pages
  - Content
    - “I am currently the second year Ph.D. student ...”
  - Hyperlinks
    - “My advisor is ...”
    - “My students: ...”

# Co-training for Semi-Supervised Learning

Betty H.C. Cheng



Professor in Computer Science and Engineering.

Ph.D., [University of Illinois at Urbana-Champaign](#)

## TEACHING INFORMATION:

- [Teaching Statement](#)
- Recent teaching assignments
  - [NSC840 Writing](#) (Summer 2002)
  - [CSE870 Advanced Software Engineering](#) (Spring 2003)
  - [CSE914 Topics in Formal Methods for Software Development](#)
  - [CSE470 Software Engineering](#) (Fall 2001)
- Useful Links for Students
  - [Programming Language Notes](#) (including Compiler module)
  - [Flex Documentation](#) (Lexical Analyzer)
    - [Flex Lab Notes and Directory](#)
  - [Bison Documentation](#) (Parser Generator)
    - [Bison Lab Notes and Directory](#)

## Research Personnel

- **Doctoral Students:**
  - [Laura Campbell](#) (PhD, expected October 2003)
  - [Min Deng](#) (PhD student)
  - Scott Fleming (PhD student)
  - [Sascha Konrad](#) (PhD student)
  - [Zhenxiao Yang](#) (PhD student)
  - [Ji Zhang](#) (PhD student)

## Software Engineering and Network Systems Laboratory

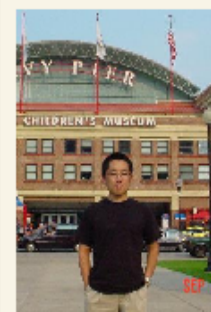


Sascha Konrad  
1510 S Shore Dr A2  
East Lansing, MI 48823  
USA  
Cell Phone: 1-517-974-9399  
Work Phone: 1-517-353-4638  
<http://www.cse.msu.edu/~konrads/>  
Email: [konradsa@cse.msu.edu](mailto:konradsa@cse.msu.edu)

[Curriculum Vitae](#)

[PGP Key](#)

For AOL Instant Messenger users:  
[Add me to your list](#)  
[Send me a message](#)



Zhenxiao Yang

Doctoral Student, [Computer Science and Engineering](#),  
[Michigan State University](#)

Advisor: [Dr. Betty H.C. Cheng](#)

(Sep., 2002, Chicago, IL)

[C.V.](#) [Research Friends](#) [Reads](#) [GoCountry](#) [ReachMe](#)



# Co-training for Semi-Supervised Learning

Betty H.C. Cheng



Professor in Computer Science and Engineering.

Ph.D., [University of Illinois at Urbana-Champaign](#)

## TEACHING INFORMATION:

- [Teaching Statement](#)
- Recent teaching assignments
  - [NSC840 Writing](#) (Summer 2002)
  - [CSE870 Advanced Software Engineering](#) (Spring 2003)
  - [CSE914 Topics in Formal Methods for Software Development](#)
  - [CSE470 Software Engineering](#) (Fall 2001)
- Useful Links for Students
  - [Programming Language Notes](#) (including Compiler module)
  - [Flex Documentation](#) (Lexical Analyzer)
    - [Flex Lab Notes and Directory](#)
  - [Bison Documentation](#) (Parser Generator)
    - [Bison Lab Notes and Directory](#)

## Research Personnel

- Doctoral Students:
  - [Laura Campbell](#) (PhD, expected October 2003)
  - [Min Deng](#) (PhD student)
  - Scott Fleming (PhD student)
  - [Sascha Konrad](#) (PhD student)
  - [Zhenxiao Yang](#) (PhD student)
  - [Ji Zhang](#) (PhD student)

It is easier to  
classify this web  
page using  
hyperlinks

It is easy to  
classify the type of  
this web page  
based on its  
content

## Software Engineering and Network Systems Laboratory

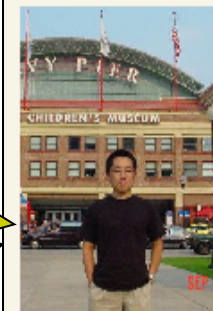


Sascha Konrad  
1510 S Shore Dr A2  
East Lansing, MI 48823  
USA  
Cell Phone: 1-517-974-9399  
Work Phone: 1-517-353-4638  
<http://www.cse.msu.edu/~konradsa/>  
Email: [konradsa@cse.msu.edu](mailto:konradsa@cse.msu.edu)

[Curriculum Vitae](#)

[PGP Key](#)

For AOL Instant Messenger users:  
[Add me to your list](#)  
[Send me a message](#)



Zhenxiao Yang

Doctoral Student, [Computer Science and Engineering](#),  
[Michigan State University](#)

Advisor: [Dr. Betty H.C. Cheng](#)

(Sep., 2002, Chicago, IL)

[C.V.](#) [Research Friends](#) [Reads](#) [GoCountry](#) [ReachMe](#)

# Co-training

- Two representation for each web page

## Content representation:

(doctoral, student, computer, university...)



Zhenxiao Yang

Doctoral Student, [Computer Science and Engineering](#),  
[Michigan State University](#)

Advisor: [Dr. Betty H.C. Cheng](#)  
(Sep., 2002, Chicago, IL)

---

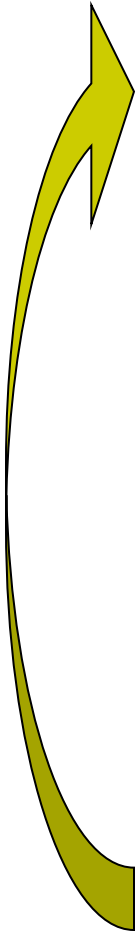
[C.V.](#) [Research](#) [Friends](#) [Reads](#) [GoCountry](#) [ReachMe](#)

## Hyperlink representation:

Inlinks: Prof. Cheng

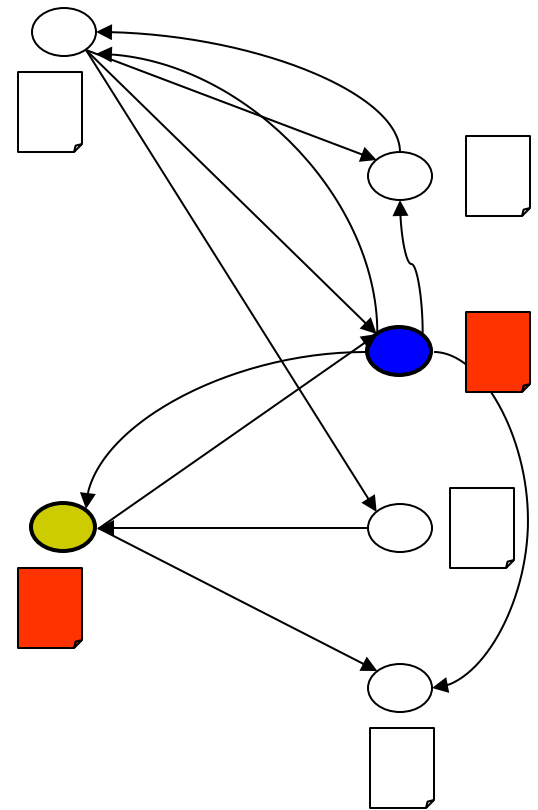
Oulinks: Prof. Cheng

# Co-training: Classification Scheme

- 
- Train a content-based classifier using labeled web pages
    - Apply the content-based classifier to classify unlabeled web pages
  - Label the web pages that have been confidently classified
  - Train a hyperlink based classifier using the web pages that are initially labeled and labeled by the content-based classifier
    - Apply the hyperlink-based classifier to classify unlabeled web pages
  - Label the web pages that have been confidently classified

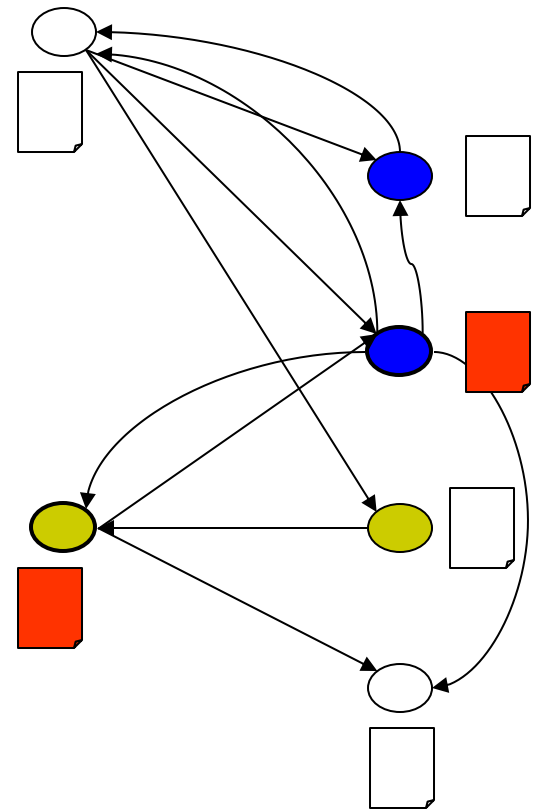
# Co-training

- Train a content-based classifier using labeled examples



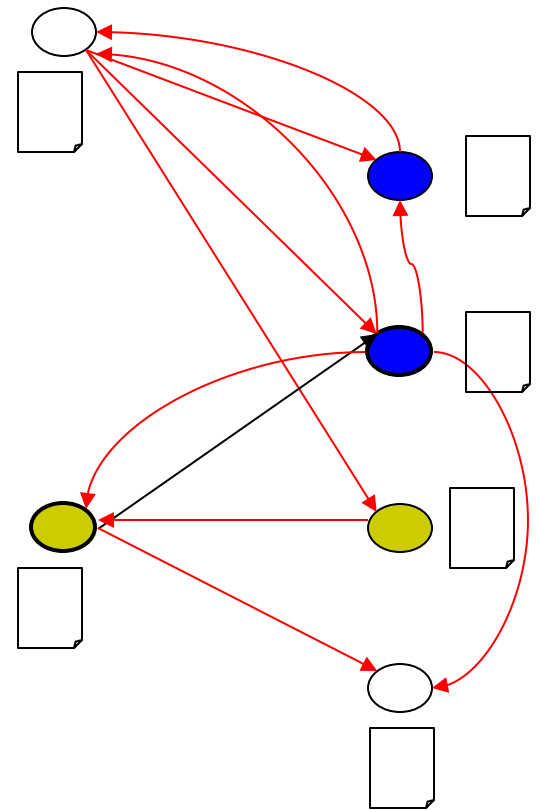
# Co-training

- Train a content-based classifier using labeled examples
- Label the unlabeled examples that are confidently classified



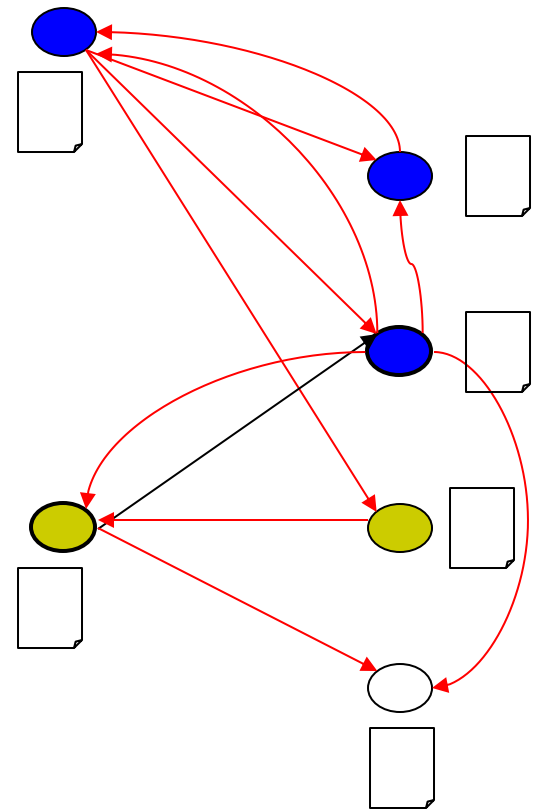
# Co-training

- Train a content-based classifier using labeled examples
- Label the unlabeled examples that are confidently classified
- Train a hyperlink-based classifier




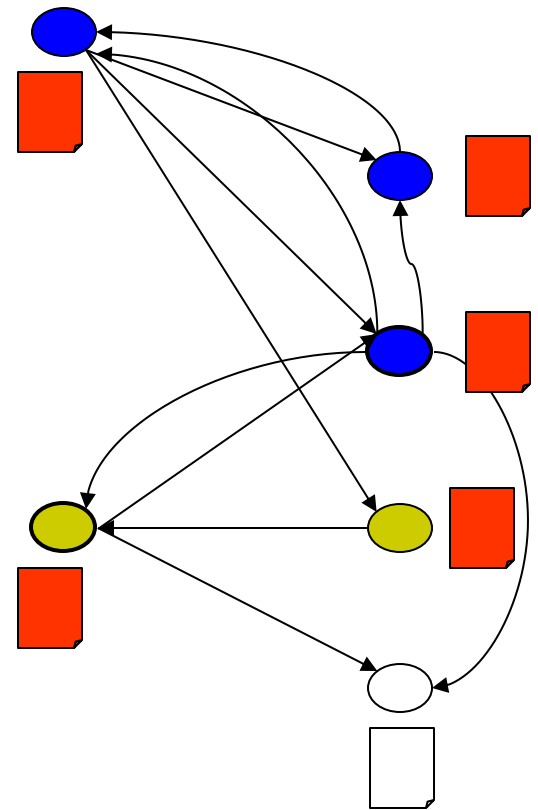
# Co-training

- Train a content-based classifier using labeled examples
- Label the unlabeled examples that are confidently classified
- Train a hyperlink-based classifier
- Label the unlabeled examples that are confidently classified



# Co-training

- 
- Train a content-based classifier using labeled examples
  - Label the unlabeled examples that are confidently classified
  - Train a hyperlink-based classifier
  - Label the unlabeled examples that are confidently classified





# Text Classification Results

(Blum & Mitchell 98)

|                     | Page-based classifier | Hyperlink-based classifier | Combined classifier |
|---------------------|-----------------------|----------------------------|---------------------|
| Supervised training | 12.9                  | 12.4                       | 11.1                |
| Co-training         | 6.2                   | 11.6                       | 5.0                 |

Table 2: Error rate in percent for classifying web pages as course home pages. The top row shows errors when training on only the labeled examples. Bottom row shows errors when co-training, using both labeled and unlabeled examples.

# Summary

- Assume two views of objects
  - Two sufficient representations
- Key idea
  - Augment training examples of one view by exploiting the classifier of the other view
- Extension to multiple view
- Challenge: find equivalent views

