Discriminative Learning over Constrained Latent Representations

Ming-Wei Chang, Dan Goldwasser, Dan Roth and Vivek Srikumar

Computer Science Department, University of Illinois at Urbana-Champaign

An one minute version of the talk

What we did

- Provide a *general recipe* for many important NLP problems
- Our algorithm: **L**earning over **C**onstrained **L**atent **R**epresentations

An one minute version of the talk

What we did

- Provide a *general recipe* for many important NLP problems
- Our algorithm: **L**earning over **C**onstrained **L**atent **R**epresentations

Example NLP problems

- Transliteration (Klementiev and Roth 2008),
- Textual entailment (RTE) (Dagan, Glickman, and Magnini 2006)
- Paraphrase identification (Dolan, Quirk, and Brockett 2004)
- Question Answering, and many more!

An one minute version of the talk

What we did

- Provide a *general recipe* for many important NLP problems
- Our algorithm: **L**earning over **C**onstrained **L**atent **R**epresentations

Example NLP problems

- Transliteration (Klementiev and Roth 2008),
- Textual entailment (RTE) (Dagan, Glickman, and Magnini 2006)
- Paraphrase identification (Dolan, Quirk, and Brockett 2004)
- Question Answering, and many more!

Problems of Interests

Binary classification tasks that require **an intermediate representation**

Example task: Paraphrase Identification

Yes/NO

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

- Q: Are sentence 1 and sentence 2 paraphrases of each other?

Example task: Paraphrase Identification

Yes/NO

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

- Q: Are sentence 1 and sentence 2 paraphrases of each other?
  - Yes, but why?

Example task: Paraphrase Identification

Yes/NO

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

- Q: Are sentence 1 and sentence 2 paraphrases of each other?
  - Yes, but why?
  - They carry the same information!

Example task: Paraphrase Identification

Yes/NO

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

- Q: Are sentence 1 and sentence 2 paraphrases of each other?
  - Yes, but why?
  - They carry the same information!
- Justifying the decision requires **an intermediate representation**

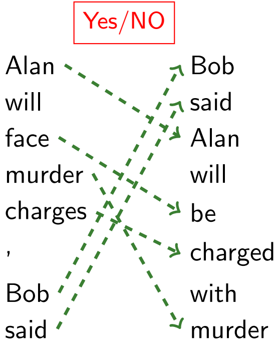
Example task: Paraphrase Identification

Yes/NO

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

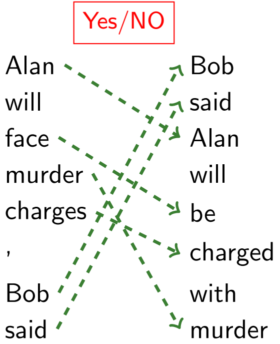
- Q: Are sentence 1 and sentence 2 paraphrases of each other?
  - Yes, but why?
  - They carry the same information!
- Justifying the decision requires **an intermediate representation**

Example task: Paraphrase Identification



- Q: Are sentence 1 and sentence 2 paraphrases of each other?
  - Yes, but why?
  - They carry the same information!
- Justifying the decision requires **an intermediate representation**
- Just an example; the real intermediate representation is more complicated

Example task: Paraphrase Identification



- Q: Are sentence 1 and sentence 2 paraphrases of each other?
  - Yes, but why?
  - They carry the same information!
- Justifying the decision requires **an intermediate representation**
- Just an example; the real intermediate representation is more complicated

Problem of interests

- Binary output problem: $y \in \{-1, 1\}$
- Intermediate representation: h
  - **Some structure that justifies the positive label**
  - The intermediate representation is **latent** (not present in the data)

Limitations of existing approaches: two-stage approach

Most systems: a two-stage approach

Stage 1: Generate the intermediate representation

Obtain intermediate representation $\longrightarrow$ Fix it (ignore the second stage) !

$$\boxed{X \longrightarrow H}$$

Limitations of existing approaches: two-stage approach

Most systems: a two-stage approach

Stage 1: Generate the intermediate representation

Obtain intermediate representation $\longrightarrow$ Fix it (ignore the second stage) !

$$\boxed{X \longrightarrow H}$$

Stage 2: Classification based on the intermediate representation

Extract features using the fixed representation and learn:

$$\boxed{\Phi(X, H) \longrightarrow Y}$$

Limitations of existing approaches: two-stage approach

Most systems: a two-stage approach

Stage 1: Generate the intermediate representation

Obtain intermediate representation $\longrightarrow$ Fix it (ignore the second stage) !

$$\boxed{X \longrightarrow H}$$

Stage 2: Classification based on the intermediate representation

Extract features using the fixed representation and learn:

$$\boxed{\Phi(X, H) \longrightarrow Y}$$

Problem: the intermediate representation **ignores** the binary task

Limitations of existing approaches: two-stage approach

Most systems: a two-stage approach

Stage 1: Generate the intermediate representation

Obtain intermediate representation → Fix it (ignore the second stage) !

$$\boxed{X \rightarrow H}$$

Stage 2: Classification based on the intermediate representation

Extract features using the fixed representation and learn:

$$\boxed{\Phi(X, H) \rightarrow Y}$$

Problem: the intermediate representation **ignores** the binary task

Limitations of existing approaches: inference

- Observation: decisions on intermediate representation are interdependent

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

Limitations of existing approaches: inference

- Observation: decisions on intermediate representation are interdependent

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

Limitations of existing approaches: inference

- Observation: decisions on intermediate representation are interdependent

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

Limitations of existing approaches: inference

- Observation: decisions on intermediate representation are interdependent

Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder

- Many frameworks use custom designed inference procedures
- Difficult to add linguistic intuition/constraints on the intermediate representation
- Difficult to generalize to other tasks

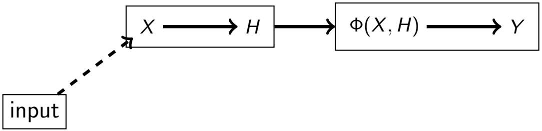
Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels

$$\boxed{X \longrightarrow H} \longrightarrow \boxed{\Phi(X, H) \longrightarrow Y}$$

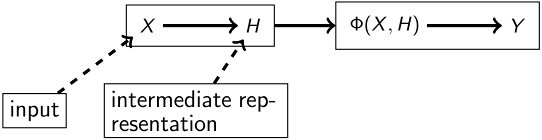
Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels



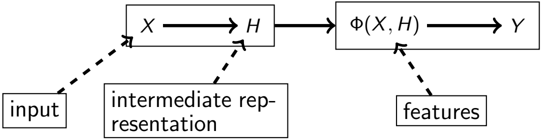
Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels



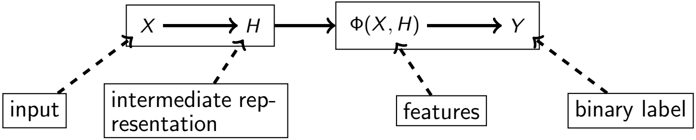
Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels



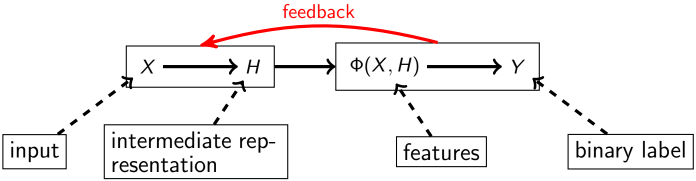
Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels



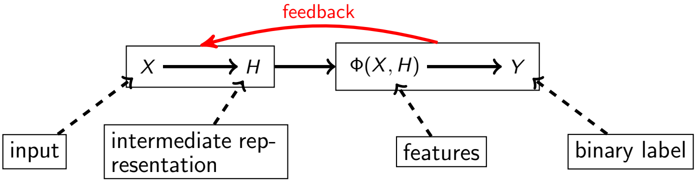
Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels



Learning Constrained Latent Representation (LCLR)

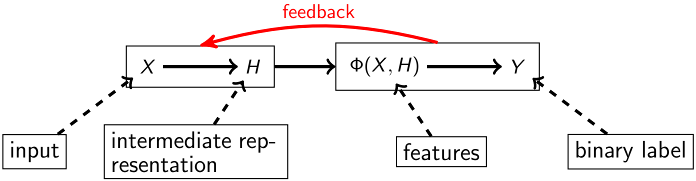
- **Property 1:** Jointly learn intermediate representations and labels



  - **Find an intermediate representation that helps the binary task**

Learning Constrained Latent Representation (LCLR)

- **Property 1:** Jointly learn intermediate representations and labels



  - **Find an intermediate representation that helps the binary task**

Property 2: Constraint-based inference for the intermediate representation

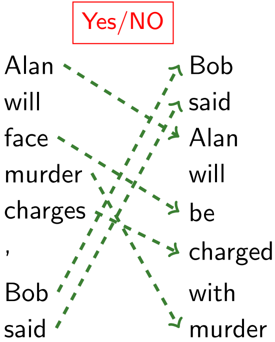
  - **Uses integer linear programming on latent variables**
  - Easy to inject constraints on *latent variables*
  - Easy to generalize to other tasks

Outline

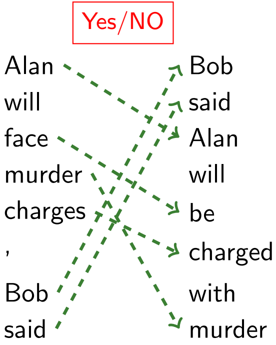
1. Motivation and Contribution
2. Property 1: Jointly learn intermediate representations and labels
3. Property 2: Constraint-based inference for the intermediate representation
4. LCLR: Putting Everything Together
5. Experiments

Outline

The intuition behind the joint approach



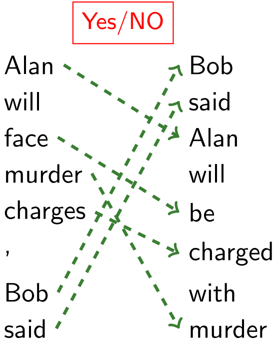
The intuition behind the joint approach



intermediate representation $\Leftrightarrow \{1, -1\}$

- Only positive examples have good intermediate representations
- **No** negative example has a good intermediate representation

The intuition behind the joint approach

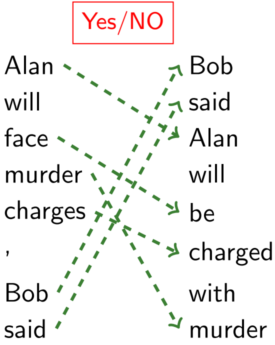


intermediate representation $\Leftrightarrow \{1, -1\}$

- Only positive examples have good intermediate representations
- **No** negative example has a good intermediate representation

x: a sentence pair
h: an alignment between two sentences
$\mathcal{H}(\mathbf{x})$: all possible alignments for **x**

The intuition behind the joint approach

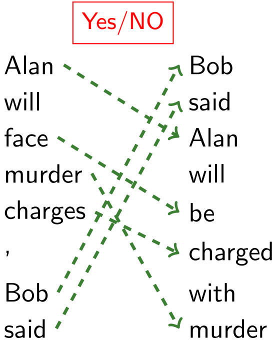


intermediate representation $\Leftrightarrow \{1, -1\}$

- Only positive examples have good intermediate representations
- **No** negative example has a good intermediate representation

$\mathbf{x}$: a sentence pair, **weight vector**: $\mathbf{u}$
h: an alignment between two sentences
$\mathcal{H}(\mathbf{x})$: all possible alignments for $\mathbf{x}$

The intuition behind the joint approach



intermediate representation $\Leftrightarrow \{1, -1\}$

- Only positive examples have good intermediate representations
- **No** negative example has a good intermediate representation

x: a sentence pair, **weight vector**: **u**
h: an alignment between two sentences
$\mathcal{H}(\mathbf{x})$: all possible alignments for **x**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$

Geometric interpretation: **the case of two examples**

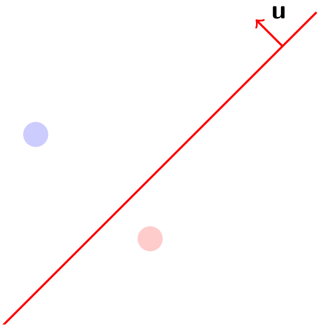
- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$

Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$

Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$



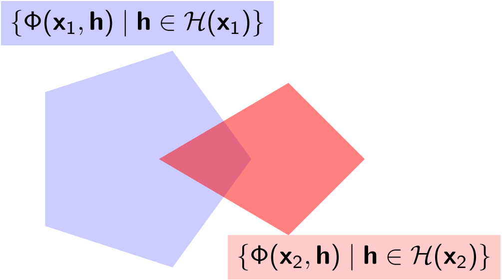
Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$

$$\{\Phi(\mathbf{x}_1, \mathbf{h}) \mid \mathbf{h} \in \mathcal{H}(\mathbf{x}_1)\}$$

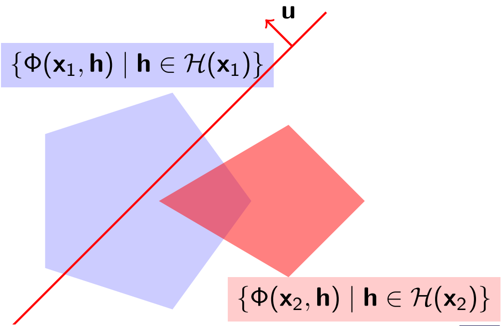
Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$



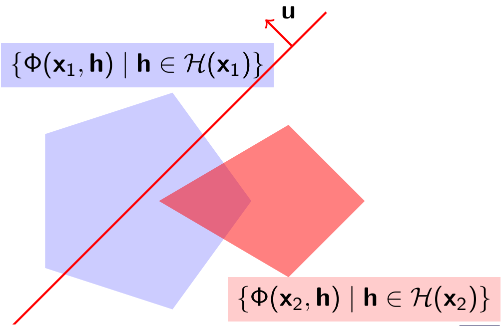
Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$



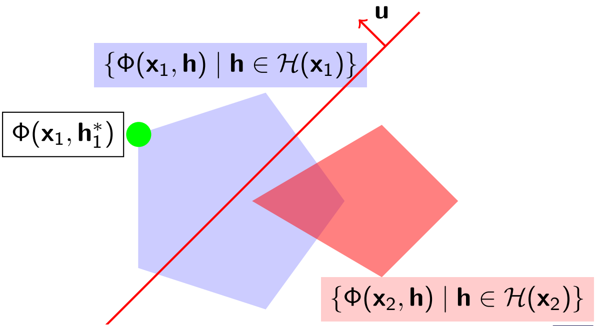
Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$



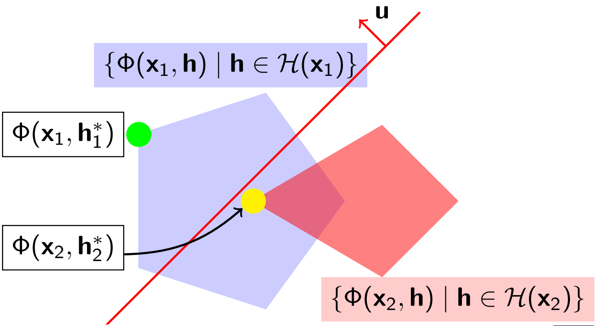
Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$



Geometric interpretation: **the case of two examples**

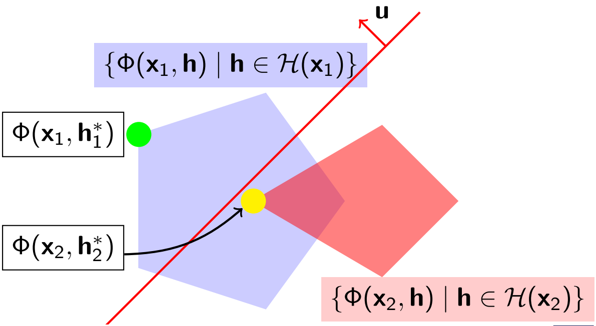
- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$



Geometric interpretation: **the case of two examples**

- Pair $\mathbf{x}_1$ is positive
  - There must exist a good explanation that justifies the positive label
  - $\exists \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_1, \mathbf{h}) \geq 0$
- Pair $\mathbf{x}_2$ is negative
  - No explanation is good enough to justify the positive label
  - $\forall \mathbf{h}, \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$, or $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}_2, \mathbf{h}) \leq 0$

- The prediction function: $\max_{\mathbf{h}} \mathbf{u}^T \Phi(\mathbf{x}, \mathbf{h})$



Outline

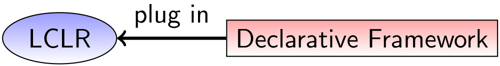
Integer Linear Programming for LCLR

Declarative Framework

- Why is a declarative framework important?
  - No more custom-designed inference procedures
  - Easy to generalize to other tasks
  - Easy to inject constraints and linguistic intuition

Declarative Framework

- Why is a declarative framework important?
  - No more custom-designed inference procedures
  - Easy to generalize to other tasks
  - Easy to inject constraints and linguistic intuition



Integer Linear Programming for LCLR

Declarative Framework

- Why is a declarative framework important?
  - No more custom-designed inference procedures
  - Easy to generalize to other tasks
  - Easy to inject constraints and linguistic intuition

LCLR $\xleftarrow{\text{plug in}}$ Declarative Framework

Paraphrasing

Model input as graphs. G_a: the first sentence. G_b: the second sentence.

- Each vertex in G_a can be mapped to at most one vertex in G_b (vice versa)
- Each edge in G_a can be mapped to at most one edge in G_b (vice versa)
- Edge mapping is active iff the corresponding node mappings are active

Integer Linear Programming for LCLR

Declarative Framework

- Why is a declarative framework important?
  - No more custom-designed inference procedures
  - Easy to generalize to other tasks
  - Easy to inject constraints and linguistic intuition
  - **Check out the CCM tutorial!**

LCLR ← plug in ← Declarative Framework

Paraphrasing

Model input as graphs. G_a: the first sentence. G_b: the second sentence.

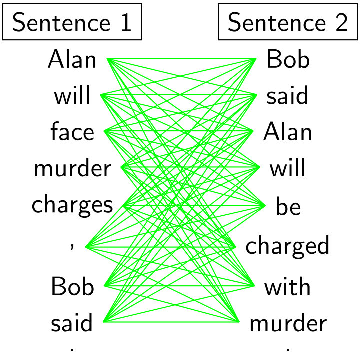
- Each vertex in G_a can be mapped to at most one vertex in G_b (vice versa)
- Each edge in G_a can be mapped to at most one edge in G_b (vice versa)
- Edge mapping is active iff the corresponding node mappings are active

Finding intermediate representation using ILP

Sentence 1	Sentence 2
Alan	Bob
will	said
face	Alan
murder	will
charges	be
,	charged
Bob	with
said	murder
.	.

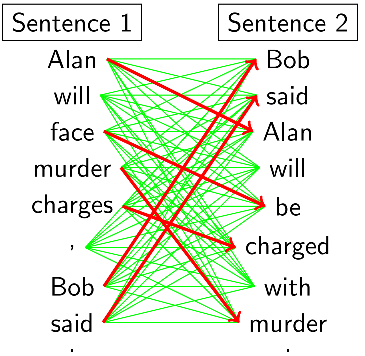
- **We need this because of the formulation. You do not need to parse the symbols in this page**

Finding intermediate representation using ILP



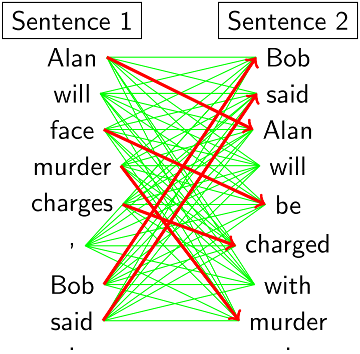
- **We need this because of the formulation. You do not need to parse the symbols in this page**
- $\Gamma(x)$, the set of all "parts" that **x** can generate $|\Gamma(x)| = 8\text{x}8 = 64$

Finding intermediate representation using ILP



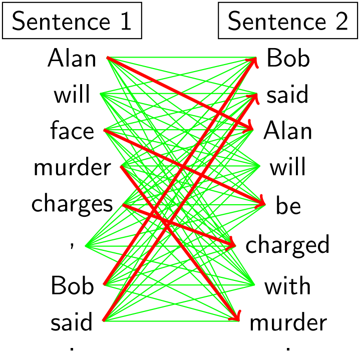
- **We need this because of the formulation. You do not need to parse the symbols in this page**
- $\Gamma(x)$, the set of all "parts" that $\mathbf{x}$ can generate $|\Gamma(x)| = 8\text{x}8 = 64$
- Rewrite $\mathbf{h} \in \{0, 1\}^{64}$ as a binary vector $\mathbf{h} = \{0, 0, 0, \ldots, 1, 0, 0, 1, 1\}$

Finding intermediate representation using ILP



- **We need this because of the formulation. You do not need to parse the symbols in this page**
- $\Gamma(x)$, the set of all "parts" that $\mathbf{x}$ can generate $|\Gamma(x)| = 8\text{x}8 = 64$
- Rewrite $\mathbf{h} \in \{0,1\}^{64}$ as a binary vector $\mathbf{h} = \{0,0,0,\ldots,1,0,0,1,1\}$
- A feature vector $\Phi_s(\mathbf{x})$ for every part h_s

Finding intermediate representation using ILP



- **We need this because of the formulation. You do not need to parse the symbols in this page**
- $\Gamma(x)$, the set of all "parts" that $\mathbf{x}$ can generate $|\Gamma(x)| = 8\text{x}8 = 64$
- Rewrite $\mathbf{h} \in \{0,1\}^{64}$ as a binary vector $\mathbf{h} = \{0,0,0,\ldots,1,0,0,1,1\}$
- A feature vector $\Phi_s(\mathbf{x})$ for every part h_s

Inference Problem = ILP formulation (pink box)

$$\max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T\Phi(\mathbf{x},\mathbf{h}) = \max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s\Phi_s(\mathbf{x})$$

Outline

1. Motivation and Contribution
2. Property 1: Jointly learn intermediate representations and labels
3. Property 2: Constraint-based inference for the intermediate representation
4. LCLR: Putting Everything Together
5. Experiments

LCLR: The objective function

- **Review:** Logistic Regression and Support Vector Machine
  - Decision Function: $f(\mathbf{x}, \mathbf{u}) \geq 0$

LCLR: The objective function

- **Review:** Logistic Regression and Support Vector Machine
  - Decision Function: $f(\mathbf{x}, \mathbf{u}) \geq 0$
  - Objective Function:

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C \sum_{i=1}^{l} \ell(-y_i\, f(\mathbf{x}, \mathbf{u})\,)$$

LCLR: The objective function

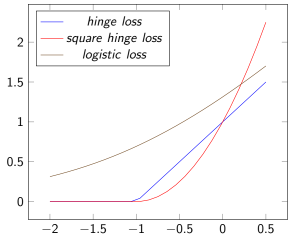
- **Review:** Logistic Regression and Support Vector Machine
  - Decision Function: $\mathbf{u}^T \Phi(\mathbf{x}) \geq 0$
  - Objective Function:

$$\min_{\mathbf{u}} \frac{1}{2} \|\mathbf{u}\|^2 + C \sum_{i=1}^{l} \ell(-y_i \mathbf{u}^T \Phi(\mathbf{x}_i))$$

LCLR: The objective function

- **Review:** Logistic Regression and Support Vector Machine
  - Decision Function: $\mathbf{u}^T \Phi(\mathbf{x}) \geq 0$
  - Objective Function:

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C \sum_{i=1}^{l} \ell(-y_i \, \mathbf{u}^T \Phi(\mathbf{x}_i) \,)$$



LCLR: The objective function

- **L**earning over **C**onstrained **L**atent **R**epresentations
  - Decision Function (**ILP**): $f(\mathbf{x}, \mathbf{u}) \geq 0$

LCLR: The objective function

- **L**earning over **C**onstrained **L**atent **R**epresentations
  - Decision Function (**ILP**): $f(\mathbf{x}, \mathbf{u}) \geq 0$
  - Objective Function

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C \sum_{i=1}^{l} \ell(-y_i\, f(\mathbf{x}, \mathbf{u})\,)$$

LCLR: The objective function

- **L**earning over **C**onstrained **L**atent **R**epresentations
  - Decision Function (**ILP**): $\max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s \Phi_s(\mathbf{x}) \geq 0$
  - Objective Function

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C\sum_{i=1}^{l} \ell(-y_i \max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s \Phi_s(\mathbf{x}))$$

LCLR: The objective function

- **L**earning over **C**onstrained **L**atent **R**epresentations
  - Decision Function (**ILP**): $\max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s\Phi_s(\mathbf{x}) \geq 0$
  - Objective Function

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C\sum_{i=1}^{l} \ell(-y_i \max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s\Phi_s(\mathbf{x}))$$

Beyond standard LR/SVM

Solves an inference problem (max) to select **h** (also affect features)

Challenges in optimizing the objective function

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C\sum_{i=1}^{l} \ell(-y_i \max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s \Phi_s(\mathbf{x}))$$

$\boxed{\text{Not a regular LR/SVM}}$

- LCLR has an inference procedure inside the minimization problem

Challenges in optimizing the objective function

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C\sum_{i=1}^{l} \ell(-y_i \max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s \Phi_s(\mathbf{x}))$$

Not a regular LR/SVM

- LCLR has an inference procedure inside the minimization problem

No shortcut

- Find the best representation for all examples
- Obtain a new weight vector using a LR/SVM package with the updated representations.
- Repeat.

Challenges in optimizing the objective function

$$\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}\|^2 + C\sum_{i=1}^{l} \ell(-y_i \max_{\mathbf{h}\in\mathcal{H}} \mathbf{u}^T \sum_{s\in\Gamma(\mathbf{x})} h_s \Phi_s(\mathbf{x}))$$

Not a regular LR/SVM

- LCLR has an inference procedure inside the minimization problem

No shortcut

- Find the best representation for all examples
- Obtain a new weight vector using a LR/SVM package with the updated representations.
- Repeat.

Does not minimize the objective function

LCLR: optimization procedure

Algorithm

1: Find the best intermediate representations for **positive examples**

2: Find the weight vector with this intermediate representation

- Still need to do inference for negative examples
- **Not a regular SVM problem even in this step!**

3: Repeat!

LCLR: optimization procedure

Algorithm

1: Find the best intermediate representations for **positive examples**

2: Find the weight vector with this intermediate representation

- Still need to do inference for negative examples
- **Not a regular SVM problem even in this step!**

3: Repeat!

This algorithm converges when ℓ is monotonically increasing and convex.

LCLR: optimization procedure

Algorithm

1: Find the best intermediate representations for **positive examples**

2: Find the weight vector with this intermediate representation

- Still need to do inference for negative examples
- **Not a regular SVM problem even in this step!**

3: Repeat!

This algorithm converges when ℓ is monotonically increasing and convex.

Properties of the algorithm: Asymmetric nature

- Asymmetry between positive and negative examples
- Converting a non-convex problem into a series of smaller convex problems

Comparison to other latent variable frameworks

Inference procedure

- Other frameworks often use application-specific inference.
- LCLR allows you to add constraints and generalize to other tasks.

Comparison to other latent variable frameworks

Inference procedure

- Other frameworks often use application-specific inference.
- LCLR allows you to add constraints and generalize to other tasks.

Learning

- Not only for SVM. Many different loss functions can be used.
- Dual coordinate descent methods and cutting plane method
  - Fewer parameters to tune. Allows parallel inference procedure.

Comparison to other latent variable frameworks

Inference procedure

- Other frameworks often use application-specific inference.
- LCLR allows you to add constraints and generalize to other tasks.

Learning

- Not only for SVM. Many different loss functions can be used.
- Dual coordinate descent methods and cutting plane method
  - Fewer parameters to tune. Allows parallel inference procedure.

CRF-like latent variable framework

- LCLR can use logistic regression and have a probabilistic interpretation
- LCLR solves the "max" problem. CRF-like models solves the "sum" problem. **"Max" enables adding constraints.** Jump

Outline

Experimental setting

Tasks

- Transliteration: Is named entity B a transliteration of A?
- Textual Entailment: Can sentence A entail sentence B?
- Paraphrase Identification

Goal of experiments

- Determine if a joint approach be better than a two-stage approach?

Two-stage approach versus LCLR

- Exactly **the same** features and definition of latent structures
  - Our two-stage approach uses a domain-dependent heuristic to find an intermediate representation
  - LCLR finds the intermediate representation automatically
- Initialization of LCLR: two-stage

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	⋆			
Our two-stage		⋆		
Our **LCLR**	⋆	⋆		

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	$\star$		N/A	89.4
Our two-stage		$\star$		
Our **LCLR**	$\star$	$\star$		

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	$\star$		N/A	89.4
Our two-stage		$\star$	80.0	85.7
Our **LCLR**	$\star$	$\star$		

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	$\star$		N/A	89.4
Our two-stage		$\star$	80.0	85.7
Our **LCLR**	$\star$	$\star$	**92.3**	**95.4**

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	$\star$		N/A	89.4
Our two-stage		$\star$	80.0	85.7
Our **LCLR**	$\star$	$\star$	**92.3**	**95.4**

Entailment System	Joint	ILP	**Acc**
Median of TAC 2009 systems			
Our two-stage		$\star$	
Our **LCLR**	$\star$	$\star$	

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	⋆		N/A	89.4
Our two-stage		⋆	80.0	85.7
Our **LCLR**	⋆	⋆	**92.3**	**95.4**

Entailment System	Joint	ILP	**Acc**
Median of TAC 2009 systems			61.5
Our two-stage		⋆	
Our **LCLR**	⋆	⋆	

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	$\star$		N/A	89.4
Our two-stage		$\star$	80.0	85.7
Our **LCLR**	$\star$	$\star$	**92.3**	**95.4**

Entailment System	Joint	ILP	**Acc**
Median of TAC 2009 systems			61.5
Our two-stage		$\star$	65.0
Our **LCLR**	$\star$	$\star$	

Experimental results

Transliteration System	Joint	ILP	**Acc**	**MRR**
(Goldwasser and Roth 2008)	$\star$		N/A	89.4
Our two-stage		$\star$	80.0	85.7
Our **LCLR**	$\star$	$\star$	**92.3**	**95.4**

Entailment System	Joint	ILP	**Acc**
Median of TAC 2009 systems			61.5
Our two-stage		$\star$	65.0
Our **LCLR**	$\star$	$\star$	**66.8**

Paraphrase Identification

Paraphrase System	Joint	ILP	**Acc**
Experiments using (Dolan, Quirk, and Brockett 2004)			
(Qiu, Kan, and Chua 2006)			72.00
(Das and Smith 2009)	⋆		73.86
(Wan, Dras, Dale, and Paris 2006)			75.60
Our two-stage		⋆	
Our **LCLR**	⋆	⋆	

Paraphrase Identification

Paraphrase System	Joint	ILP	**Acc**
Experiments using (Dolan, Quirk, and Brockett 2004)			
(Qiu, Kan, and Chua 2006)			72.00
(Das and Smith 2009)	⋆		73.86
(Wan, Dras, Dale, and Paris 2006)			75.60
Our two-stage		⋆	76.23
Our **LCLR**	⋆	⋆	**76.41**

Paraphrase Identification

Paraphrase System	Joint	ILP	**Acc**
Experiments using (Dolan, Quirk, and Brockett 2004)			
(Qiu, Kan, and Chua 2006)			72.00
(Das and Smith 2009)	⋆		73.86
(Wan, Dras, Dale, and Paris 2006)			75.60
Our two-stage		⋆	76.23
Our **LCLR**	⋆	⋆	**76.41**
Experiments using **Noisy data set**			
Our two-stage		⋆	
Our **LCLR**	⋆	⋆	

Paraphrase Identification

Paraphrase System	Joint	ILP	**Acc**
Experiments using (Dolan, Quirk, and Brockett 2004)			
(Qiu, Kan, and Chua 2006)			72.00
(Das and Smith 2009)	⋆		73.86
(Wan, Dras, Dale, and Paris 2006)			75.60
Our two-stage		⋆	76.23
Our **LCLR**	⋆	⋆	**76.41**
Experiments using **Noisy data set**			
Our two-stage		⋆	72.00
Our **LCLR**	⋆	⋆	**72.75**

Conclusions

LCLR = Constraint-based Inference + Large Margin Learning

Contributions

- LCLR joint approach is better than two-stage approaches
- LCLR allows the use of constraints on latent variables
- A novel learning framework

Conclusions

LCLR = Constraint-based Inference + Large Margin Learning

Contributions

- LCLR joint approach is better than two-stage approaches
- LCLR allows the use of constraints on latent variables
- A novel learning framework

Bonus: Learning Structures with Indirect Supervision

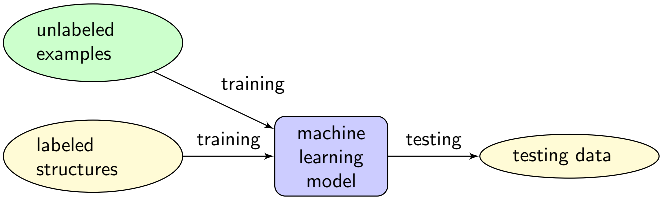
- Easy to get **binary** labeled data can be used to improve learning **structures**!
- Check out our ICML paper this year!

Thank you!

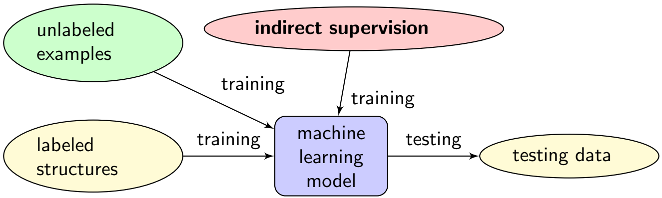
Thank you!!

- Our learning code is available: the **JLIS** package
- `http://l2r.cs.uiuc.edu/~cogcomp/software.php`

Main Idea: Learning with indirect supervision

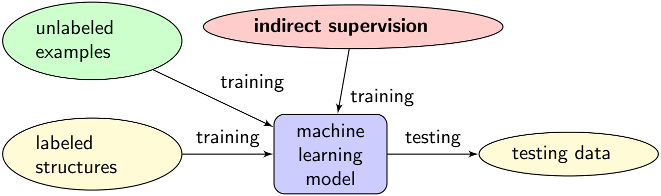


Main Idea: Learning with indirect supervision



Indirect supervision: the supervision form that does not tell you the target output directly

Main Idea: Learning with indirect supervision

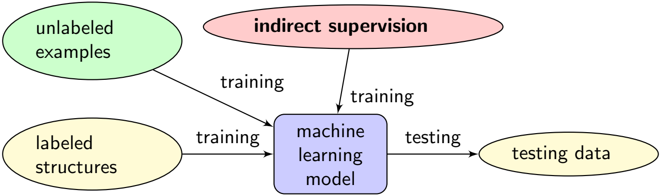


Indirect supervision: the supervision form that does not tell you the target output directly

Advantage of using indirect supervision

- Can directly use human/domain knowledge to improve the model
- Allow us to use supervision signals that are a lot easier to obtain than labeling structures
- Use *existing* labeled data for the related tasks

Main Idea: Learning with indirect supervision



Indirect supervision: the supervision form that does not tell you the target output directly

Advantage of using indirect supervision

- Can directly use human/domain knowledge to improve the model
- Allow us to use supervision signals that are a lot easier to obtain than labeling structures
- Use *existing* labeled data for the related tasks

Indirect supervision greatly reduce the supervision effort!

Compared to CRF-like latent variable framework

CRF-like latent variable framework

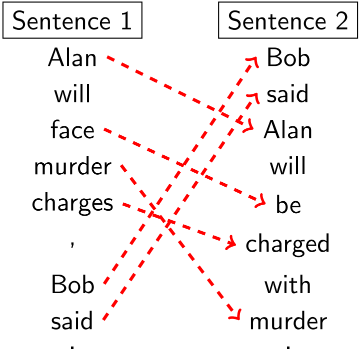
$$P(y=1|\mathbf{x}) = \sum_{\mathbf{h}} P(y=1,\mathbf{h}|\mathbf{x}) = \frac{\sum_{\mathbf{h}} \exp(\mathbf{u}^T\phi(\mathbf{x},\mathbf{h},y=1))}{\sum_{\mathbf{h},y} \exp(\mathbf{u}^T\phi(\mathbf{x},\mathbf{h},y))}$$

LCLR with logistic loss

$$P(y=1|\mathbf{x}) = \frac{\max_{\mathbf{h}} \exp(\mathbf{u}^T\phi(\mathbf{x},\mathbf{h}))}{1+\max_{\mathbf{h}} \exp(\mathbf{u}^T\phi(\mathbf{x},\mathbf{h}))}$$

- Difference 1: LCLR only models the "goodness"
  - This is important for many NLP problems, where only positive examples have good representations.
- Difference 2: LCLR only need to solve the max inference
  - Sometimes calculating sum is a lot harder!!
- Jump back

Paraphrase Identification: Revisited



- **Left**: The intermediate representation is not expressive enough
  - For example, "word ordering" is a problem
- The real setting
  - Input: two word sequence $\longrightarrow$ two graphs.
  - We used Stanford Parser to construct dependency parse trees for each sentence

Integer Linear Programming to solve the graph matching problem

- Four types of sub-structure: node matching, node-deletion, edge matching, edge-deletion
- Add constraints to enforce consistency
  - edge matching if and only if the corresponding nodes are matched

- Jump Back

Dagan, I., O. Glickman, and B. Magnini (Eds.) (2006).
The PASCAL Recognising Textual Entailment Challenge.

Das, D. and N. A. Smith (2009).
Paraphrase identification as probabilistic quasi-synchronous recognition.
In *ACL*.

Dolan, W., C. Quirk, and C. Brockett (2004).
Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources.
In *COLING*.

Goldwasser, D. and D. Roth (2008).
Active sample selection for named entity transliteration.
In *ACL*.
Short Paper.

Klementiev, A. and D. Roth (2008).
Named entity transliteration and discovery in multilingual corpora.
In C. Goutte, N. Cancedda, M. Dymetman, and G. Foster (Eds.), *Learning Machine Translation*.

Qiu, L., M.-Y. Kan, and T.-S. Chua (2006).
Paraphrase recognition via dissimilarity significance classification.
In *EMNLP*.

Wan, S., M. Dras, R. Dale, and C. Paris (2006).
Using dependency-based features to take the p̈ara-farceöut of paraphrase.
In *Proc. of the Australasian Language Technology Workshop (ALTW)*.