Constrained Conditional Models
Learning and Inference in Natural Language Understanding

Dan Roth
Department of Computer Science
University of Illinois at Urbana-Champaign

With thanks to:
Collaborators: Ming-Wei Chang, Vasin Punyakanok, Lev Ratinov, Nick Rizzolo, Mark Sammons, Scott Yih, Dav Zimak
Funding: ARDA, under the AQUAINT program
NSF: ITR IIS-0085836, ITR IIS-0428472, ITR IIS-0085980, SoD-HCER-0613885
A DOI grant under the Reflex program; DHS
DASH Optimization (Xpress-MP)
Nice to Meet You
Learning and Inference

- Global decisions in which several local decisions play a role but there are mutual dependencies on their outcome.
  - E.g. Structured Output Problems – multiple dependent output variables

- (Learned) models/classifiers for different sub-problems
  - In some cases, not all models are available to be learned simultaneously
  - Key examples in NLP are Textual Entailment and QA
  - In these cases, constraints may appear only at evaluation time

- Incorporate models’ information, along with prior knowledge/constraints, in making coherent decisions
  - Decisions that respect the learned models as well as domain & context specific knowledge/constraints.
Inference
Comprehension

A process that maintains and updates a collection of propositions about the state of affairs.

(ENGLAND, June, 1989) - Christopher Robin is alive and well. He lives in England. He is the same person that you read about in the book, Winnie the Pooh. As a boy, Chris lived in a pretty home called Cotchfield Farm. When Chris was three years old, his father wrote a poem about him. The poem was printed in a magazine for others to read. Mr. Robin then wrote a book. He made up a fairy tale land where Chris lived. His friends were animals. There was a bear called Winnie the Pooh. There was also an owl and a young pig, called a piglet. All the animals were stuffed toys that Chris owned. Mr. Robin made them come to life with his words. The places in the story were all near Cotchfield Farm. Winnie the Pooh was written in 1925. Children still love to read about Christopher Robin and his animal friends. Most people don’t know he is a real person who is grown now. He has written two books of his own. They tell what it is like to be famous.

1. Christopher Robin was born in England.
2. Winnie the Pooh is a title of a book.
3. Christopher Robin’s dad was a magician.
4. Christopher Robin must be at least 65 now.

This is an Inference Problem
This Talk: Constrained Conditional Models

- A general inference framework that combines
  - Learning conditional models with using declarative expressive constraints
  - Within a constrained optimization framework
    - Formulate a decision process as a constrained optimization problem, or
    - Break up a complex problem into a set of sub-problems and require components’ outcomes to be consistent modulo constraints

- Has been shown useful in the context of many NLP problems
  - SRL, Summarization; Co-reference; Information Extraction
  - [Roth&Yih04,07; Punyakanok et.al 05,08; Chang et.al07,08; Clarke&Lapata06,07; Denise&Baldridge07]

- Here: focus on Learning and Inference for Structured NLP Problems
Outline

- **Constrained Conditional Models**
  - Motivation
  - Examples

- **Training Paradigms**: Investigate ways for training models and combining constraints
  - Joint Learning and Inference vs. decoupling Learning & Inference
  - Guiding Semi-Supervised Learning with Constraints
    - Features vs. Constraints
    - Hard and Soft Constraints

- **Examples**
  - Semantic Parsing
  - Information Extraction
  - Pipeline processes
Inference with General Constraint Structure [Roth & Yih'04]

Dole's wife, Elizabeth, is a native of N.C.

Some Questions:
How to guide the global inference?
Why not learn Jointly?

Models could be learned separately; constraints may come up only at decision time.
Task of Interests: Structured Output

- For each instance, assign values to a set of variables
- Output variables depends on each other

Common tasks in
  - Natural language processing
    - Parsing; Semantic Parsing; Summarization; Transliteration; Co-reference resolution,…
  - Information extraction
    - Entities, Relations,…

Many pure machine learning approaches exist
  - Hidden Markov Models (HMMs); CRFs
  - Perceptrons…

However, …
Information Extraction via Hidden Markov Models


Prediction result of a trained HMM

[AUTHOR] Lars Ole Andersen. Program analysis and specialization for the C Programming language.
[DATE] Un satisfactory results!
Strategies for Improving the Results

- (Pure) Machine Learning Approaches
  - Higher Order HMM/CRF?
  - Increasing the window size?
  - Adding a lot of new features
    - Requires a lot of labeled examples
  - What if we only have a few labeled examples?

- Increasing the model complexity

- Any other options?
  - Humans can immediately tell bad outputs
  - The output does not make sense
Information extraction without Prior Knowledge


Prediction result of a trained HMM

[TITLE] Violates lots of natural constraints!
Examples of Constraints

- Each field must be a **consecutive list of words and can appear at most once** in a citation.

- State transitions must occur on **punctuation marks**.

- The citation can only start with **AUTHOR** or **EDITOR**.

- The words **pp., pages** correspond to **PAGE**.

- Four digits starting with **20xx and 19xx** are **DATE**.

- **Quotations** can appear only in **TITLE**.

- …….

  - **Easy to express pieces of “knowledge”**

  - **Non Propositional; May use Quantifiers**
Information Extraction with Constraints

- Adding constraints, we get correct results!
  - Without changing the model

- [AUTHOR]: Lars Ole Andersen
- [TITLE]: Program analysis and specialization for the C Programming language
- [TECH-REPORT]: PhD thesis
- [INSTITUTION]: DIKU, University of Copenhagen
- [DATE]: May, 1994
Problem Setting

- Random Variables $Y$:
  - $Y_1$
  - $Y_2$
  - $Y_3$
  - $Y_4$
  - $Y_5$
  - $Y_6$
  - $Y_7$
  - $Y_8$

- Conditional Distributions $P$ (learned by models/classifiers)
- Constraints $C$—any Boolean function defined on partial assignments (possibly: + weights $W$)

Goal: Find the “best” assignment
  - The assignment that achieves the highest global performance.
  - This is an Integer Programming Problem

$Y^* = \arg\max_Y P \cdot Y$ (subject to constraints $C$)
Formal Model

\[ \text{argmax } \lambda \cdot F(x, y) \]

- Weight Vector for “local” models
- A collection of Classifiers; Log-linear models (HMM, CRF) or a combination

Subject to constraints

Penalty for violating the constraint

(soft) constraints component

How far away is \( y \) from a “legal” assignment

How to solve?
This is an Integer Linear Program
Solving using ILP packages gives an exact solution.
Search techniques are also possible

How to train?
How to decompose global objective function?
Should we incorporate constraints in the learning process?
Example: Semantic Role Labeling

Who did what to whom, when, where, why,…

I left my pearls to my daughter in my will.

\[ I_{A_0} \text{ left } [my \text{ pearls}]_{A_1} [to \text{ my daughter}]_{A_2} [in \text{ my will}]_{AM-Loc}. \]

- **A0**  Leaver
- **A1**  Things left
- **A2**  Benefactor
- **AM-LOC**  Location

I left my pearls to my daughter in my will.

**Special Case (structured output problem):**

- here, all the data is available at one time;
- in general, classifiers might be learned from different sources, at different times, at different contexts.

**Implications on training paradigms**

**Overlapping arguments**

If A2 is present, A1 must also be present.
Semantic Role Labeling (2/2)

- PropBank [Palmer et. al. 05] provides a large human-annotated corpus of semantic verb-argument relations.
  - It adds a layer of generic semantic labels to Penn Tree Bank II.
  - (Almost) all the labels are on the constituents of the parse trees.

- Core arguments: A0-A5 and AA
  - different semantics for each verb
  - specified in the PropBank Frame files

- 13 types of adjuncts labeled as AM-arg
  - where arg specifies the adjunct type
Algorithmic Approach

- **Identify** argument candidates
  - Pruning [Xue & Palmer, EMNLP’04]
  - Argument Identifier
    - Binary classification (SNoW)

- **Classify** argument candidates
  - Argument Classifier
    - Multi-class classification (SNoW)

- **Inference**
  - Use the estimated probability distribution given by the argument classifier
  - Use structural and linguistic constraints
  - Infer the optimal global output

---

I left my nice pearls to her

Identify Vocabulary

**EASY**

Identify candidate arguments

Inference over (old and new) Vocabulary
Inference

- The output of the argument classifier often violates some constraints, especially when the sentence is long.

- Finding the **best legitimate output** is formalized as an optimization problem and solved via Integer Linear Programming.  
  
  [Punyakanok et. al 04, Roth & Yih 04;05]

- Input:
  - The probability estimation (by the argument classifier)
  - Structural and linguistic constraints

- Allows incorporating expressive (non-sequential) constraints on the variables (the arguments types).
Integer Linear Programming Inference

- For each argument $a_i$
  - Set up a Boolean variable: $a_{i,t}$ indicating whether $a_i$ is classified as $t$

- Goal is to maximize
  - $\sum_i \text{score}(a_i = t) a_{i,t}$
  - Subject to the (linear) constraints

- If $\text{score}(a_i = t) = P(a_i = t)$, the objective is to find the assignment that maximizes the expected number of arguments that are correct and satisfies the constraints.

The Constrained Conditional Model is completely decomposed during training.
Constraints

- No duplicate argument classes
  \[ \sum_{a \in \text{POTARG}} x_{\{a = A0\}} \leq 1 \]

- R-ARG
  \[ \forall a_2 \in \text{POTARG}, \sum_{a \in \text{POTARG}} x_{\{a = A0\}} \geq x_{\{a_2 = R-A0\}} \]

- C-ARG
  \[ \forall a_2 \in \text{POTARG}, \\
  \sum_{(a \in \text{POTARG}) \land (a \text{ is before } a_2)} x_{\{a = A0\}} \geq x_{\{a_2 = C-A0\}} \]

- Many other possible constraints:
  - Unique labels
  - No overlapping or embedding
  - Relations between number of arguments; order constraints
  - If verb is of type A, no argument of type B

Any Boolean rule can be encoded as a linear constraint.

If there is an R-ARG phrase, there is an ARG Phrase

If there is an C-ARG phrase, there is an ARG before it

Universally quantified

LBJ: allows a developer to encode constraints in FOL; these are compiled into linear inequalities automatically.

Joint inference can be used also to combine different SRL Systems.
Semantic Role Labeling

Semantic Role Labeling Output

Input Text:
A car bomb that exploded outside the U.S. military base in Beniji killed 11 Iraqi citizens.

Result: Complete!
□ General Explanation of Argument Labels

<table>
<thead>
<tr>
<th>Role</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>bomb [A1]</td>
</tr>
<tr>
<td>car</td>
<td></td>
</tr>
<tr>
<td>bomb</td>
<td></td>
</tr>
<tr>
<td>that</td>
<td>bomb (Reference) [R-A1]</td>
</tr>
<tr>
<td>exploded</td>
<td>V: explode</td>
</tr>
<tr>
<td>outside</td>
<td>location [AM-LOC]</td>
</tr>
<tr>
<td>the</td>
<td></td>
</tr>
<tr>
<td>U.S.</td>
<td></td>
</tr>
<tr>
<td>military</td>
<td>temporal [AM-TMP]</td>
</tr>
<tr>
<td>base</td>
<td></td>
</tr>
<tr>
<td>in</td>
<td></td>
</tr>
<tr>
<td>Beniji</td>
<td>location [AM-LOC]</td>
</tr>
<tr>
<td>killed</td>
<td>V: kill</td>
</tr>
<tr>
<td>11</td>
<td>corpse [A1]</td>
</tr>
<tr>
<td>Iraqi</td>
<td></td>
</tr>
<tr>
<td>citizens</td>
<td></td>
</tr>
</tbody>
</table>

Semantic parsing reveals several relations in the sentence along with their arguments.

This approach produces a very good semantic parser. F1~90%
Easy and fast: ~7 Sent/Sec (using Xpress-MP)

Top ranked system in CoNLL’05 shared task
Key difference is the Inference

Screen shot from a CCG demo http://L2R.cs.uiuc.edu/~cogcomp
Outline

- Constrained Conditional Models
  - Motivation
  - Examples

Training Paradigms: Investigate ways for training models and combining constraints
  - Joint Learning and Inference vs. decoupling Learning & Inference
  - Guiding Semi-Supervised Learning with Constraints
    - Features vs. Constraints
    - Hard and Soft Constraints

Examples
  - Semantic Parsing
  - Information Extraction
  - Pipeline processes
Eyeing the huge market potential, currently led by Google, Yahoo took over search company Overture Services Inc. last year.

Is it true that...

(Textual Entailment)

Yahoo acquired Overture
Overture is a search company
Google is a search company
Google owns Overture

Semantic Role Labeling
Punyakanok et. al’05,08

Inference for Entailment
Braz et. al’05, 07

Phrasal verb paraphrasing
[Connor&Roth’07]

Entity matching [Li et. al, AAAI’04, NAACL’04]

Phrasal verb paraphrasing
[Connor&Roth’07]
Training Paradigms that Support Global Inference

- Incorporating general constraints (Algorithmic Approach)
  - Allow both statistical and expressive declarative constraints
  - Allow non-sequential constraints (generally difficult)

Coupling vs. Decoupling Training and Inference.

- Incorporating global constraints is important but
- Should it be done only at evaluation time or also at training time?
- How to decompose the objective function and train in parts?
- Issues related to:
  - Modularity, efficiency and performance, availability of training data
  - Problem specific considerations
Training in the presence of Constraints

**General Training Paradigm:**

- **First Term:** Learning from data (could be further decomposed)
- **Second Term:** Guiding the model by constraints
- Can choose if constraints’ weights trained, when and how, or taken into account only in evaluation.

\[
\arg\max_y \lambda \cdot F(x, y) - \sum_{i=1}^{K} \rho_i d(y, 1_{C_i(x)})
\]
L+I: Learning plus Inference

Training w/o Constraints

Testing: Inference with Constraints

IBT: Inference-based Training

Cartoon: each model can be more complex and may have a view on a set of output variables.

Learning the components together!
Perceptron-based Global Learning

True Global Labeling

Apply Predictions:

\[ f_1(x) \]
\[ f_2(x) \]
\[ f_3(x) \]
\[ f_4(x) \]
\[ f_5(x) \]

Which one is better? When and Why?
Claims

- When the local modes are “easy” to learn, L+I outperforms IBT.
  - In many applications, the components are identifiable and easy to learn (e.g., argument, open-close, PER).
- Only when the local problems become difficult to solve in isolation, IBT outperforms L+I, but needs a larger number of training examples.
  - When data is scarce, problems are not easy and constraints can be used, along with a “weak” model, to label unlabeled data and improve model.

L+I: cheaper computationally; modular
IBT is better in the limit, and other extreme cases.

- Other training paradigms are possible
- Pipeline-like Sequential Models:
  - Identify a preferred ordering among components
  - Learn k-th model jointly with previously learned models
Bound Prediction

- Local \( \varepsilon \leq \varepsilon_{\text{opt}} + \left( \frac{d \log m + \log 1/\delta}{m} \right)^{1/2} \)
- Global \( \varepsilon \leq 0 + \left( \frac{cd \log m + c^2d + \log 1/\delta}{m} \right)^{1/2} \)

L+I vs. IBT: the more identifiable individual problems are, the better overall performance is with L+I

Indication for hardness of problem

\( \varepsilon_{\text{opt}} = 0 \)

Simulated Data
Relative Merits: SRL

In some cases problems are hard due to lack of training data. Semi-supervised learning

**Semantic Role Labeling**

- **L+I** is better.
- When the problem is artificially made harder, the tradeoff is clearer.

**Difficulty of the learning problem**  
(Num features)

- **Hard**
- **Easy**
Outline

- **Constrained Conditional Models**
  - Motivation
  - Examples

- **Training Paradigms:** Investigate ways for training models and combining constraints
  - Joint Learning and Inference vs. decoupling Learning & Inference
  - Guiding Semi-Supervised Learning with Constraints
    - Features vs. Constraints
    - Hard and Soft Constraints

- **Examples**
  - Semantic Parsing
  - Information Extraction
  - Pipeline processes
Information extraction without **Prior Knowledge**


**Prediction result of a trained HMM**

[**AUTHOR**] Lars Ole Andersen
[**TITLE**] Program analysis and specialization for the C Programming language
[**EDITOR**] 
[**BOOKTITLE**] 
[**TECH-REPORT**] PhD thesis
[**INSTITUTION**] DIKU, University of Copenhagen
[**DATE**] May 1994

Violates lots of **natural constraints**!
Examples of Constraints

- Each field must be a consecutive list of words and can appear at most once in a citation.

- State transitions must occur on punctuation marks.

- The citation can only start with **AUTHOR** or **EDITOR**.

- The words *pp.*, *pages* correspond to **PAGE**.

- Four digits starting with 20xx and 19xx are **DATE**.

- *Quotations* can appear only in **TITLE**.

- ......

**Easy to express pieces of “knowledge”**

**Non Propositional; May use Quantifiers**
Information Extraction with Constraints

- Adding constraints, we get correct results!
- Without changing the model

- [AUTHOR] Lars Ole Andersen.
- [TITLE] Program analysis and specialization for the C Programming language.
- [INSTITUTION] DIKU, University of Copenhagen.
Features Versus Constraints

\[ f_{\Phi,C}(x, y) = \sum w_i \phi_i(x, y) - \sum \rho_i d_{C_i}(x, y). \]

- \( \phi_i : X \times Y \rightarrow \mathbb{R} \);
- \( C_i : X \times Y \rightarrow \{0,1\} \);
- \( d : X \times Y \rightarrow \mathbb{R} \);

- In principle, constraints and features can encode the same properties.
- In practice, they are very different.

- Features
  - Local, short distance properties – to allow tractable inference.
  - Propositional (grounded):
    - E.g. True if: “the” followed by a Noun occurs in the sentence.”

- Constraints
  - Global properties
  - Quantified, first order logic expressions
    - E.g. True if: “all \( y_i \)s in the sequence \( y \) are assigned different values.”
Encoding Prior Knowledge

- Consider encoding the knowledge that:
  - Entities of type A and B cannot occur simultaneously in a sentence

- The “Feature” Way
  - Results in higher order HMM, CRF
  - May require designing a model tailored to knowledge/constraints
  - Large number of new features: might require more labeled data
  - Wastes parameters to learn indirectly knowledge we have.

- The Constraints Way
  - Keeps the model simple; add expressive constraints directly
  - A small set of constraints
  - Allows for decision time incorporation of constraints
Outline

- Constrained Conditional Models
  - Motivation
  - Examples

- Training Paradigms: Investigate ways for training models and combining constraints
  - Joint Learning and Inference vs. decoupling Learning & Inference
  - Guiding Semi-Supervised Learning with Constraints
    - Features vs. Constraints
    - Hard and Soft Constraints

- Examples
  - Semantic Parsing
  - Information Extraction
  - Pipeline processes
Guiding Semi-Supervised Learning with Constraints

- In traditional Semi-Supervised learning the model can drift away from the correct one.
- Constraints can be used
  - At decision time, to bias the objective function towards favoring constraint satisfaction.
  - At training to improve labeling of un-labeled data (and thus improve the model)
Training Strategies

\[ f_{\Phi,C}(x,y) = \sum w_i \phi_i(x,y) - \sum \rho_i d_C(x,y). \]

- Hard Constraints or Weighted Constraints
  - Hard constraints: set penalties to \textit{infinity}
    - No more \textit{degrees} of violation
  - Weighted Constraints
    - Need to figure out penalties values

- Factored / Jointed Approaches
  - Factored Models \((L+I)\)
    - Learn model weights and constraints’ penalties \textit{separately}
  - Joint Models \((IBT)\)
    - Learn the model weights and constraints’ penalties \textit{jointly}
  - \(L+I\) vs \(IBT\): [Punyakanon et. al. 05]

Training Algorithms:
- \(L+ CI, L+ wCI\)
- \(CIBT, wCIBT\)
Factored (L+I) Approaches

- Learning model weights
  - HMM

- Constraints Penalties
  - Hard Constraints: infinity
  - Weighted Constraints:
    - $\rho_i = -\log P\{\text{Constraint } C_i \text{ is violated in training data}\}$
Joint Approaches

**Algorithm 1 IBT training: CIBT & wCIBT**

**Require:** $D$ is the training dataset, $K$ is the number of constraints, $M$ is the number of iterations

1: for $i = 1 \ldots K$ do
2: \hspace{1em} \text{if (hardConstraints) then $\rho_i = \infty$ else $\rho_i = 0$}
3: \hspace{1em} end for
4: for $i = 1 \ldots M$ do
5: \hspace{2em} for $(x, y^*) \in D$ do
6: \hspace{3em} $\hat{y} = \arg\max_y [\sum w_i \phi_i(x, y) - \sum \rho_i d_C(x, y)]$
7: \hspace{3em} $w = w + \Phi(x, y^*) - \Phi(x, \hat{y})$
8: \hspace{3em} \text{if weightedConstraints then}
9: \hspace{4em} $\rho = \rho + d_C(x, y^*) - d_C(x, \hat{y})$
10: \hspace{3em} end if
11: \hspace{2em} end for
12: end for
Semi-supervised Learning with Constraints

\[ \lambda = \text{learn}(T) \]

For \( N \) iterations do

\[ T = \emptyset \]

For each \( x \) in unlabeled dataset

\[ \{y_1, \ldots, y_K\} \leftarrow \text{InferenceWithConstraints}(x, C, \lambda) \]

\[ T = T \cup \{(x, y_i)\}_{i=1}^{K} \]

\[ \lambda = \gamma \lambda + (1 - \gamma) \text{learn}(T) \]

[Chang, Ratinov, Roth, ACL’07]

Supervised learning algorithm parameterized by \( \lambda \).

Inference based augmentation of the training set (feedback) (inference with constraints).

Learn from new training data. Weigh supervised and unsupervised model.
Outline

- Constrained Conditional Model
  - Feature v.s Constraints
  - Inference
  - Training
  - Semi-supervised Learning

Results

- Discussion
Results on Factored Model -- Citations

In all cases:
semi = 1000 unlabeled examples.

In all cases: Significantly better results than existing results [Chang et. al. ’07]
Results on Factored Model -- Advertisements
Hard Constraints vs. Weighted Constraints

Constraints are close to perfect

(a)-Citations

<table>
<thead>
<tr>
<th>Training samples</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weighted constraints</td>
<td>77.09</td>
<td>81.25</td>
<td>85.00</td>
<td>94.51</td>
</tr>
<tr>
<td>Hard constraints</td>
<td>78.18</td>
<td>81.11</td>
<td>85.16</td>
<td>92.80</td>
</tr>
</tbody>
</table>

(b)-Advertisement

<table>
<thead>
<tr>
<th>Training samples</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>300</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weighted constraints</td>
<td>71.46</td>
<td>75.61</td>
<td>77.76</td>
<td>82.06</td>
</tr>
<tr>
<td>Hard constraints</td>
<td>69.91</td>
<td>73.46</td>
<td>75.25</td>
<td>79.59</td>
</tr>
</tbody>
</table>

Labeled data might not follow the constraints
**Factored vs. Jointed Training**

- **Using the best models for both settings**
  - Factored training: HMM + weighted constraints
  - Jointed training: Perceptron + weighted constraints
  - Same feature set

- **Without constraints**
  - Factored Model is better
  - More labeled data with a small # of examples

Agrees with earlier results in the supervised setting ICML’05, IJCAI’5

<table>
<thead>
<tr>
<th>Training</th>
<th>Discriminative</th>
<th>Maximum Likelihood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L</td>
<td>wCIBT</td>
</tr>
<tr>
<td>5</td>
<td>50.14</td>
<td>61.65</td>
</tr>
<tr>
<td>10</td>
<td>59.90</td>
<td>69.64</td>
</tr>
<tr>
<td>20</td>
<td>68.26</td>
<td>78.46</td>
</tr>
<tr>
<td>300</td>
<td>89.83</td>
<td>93.89</td>
</tr>
</tbody>
</table>
Value of Constraints in Semi-Supervised Learning

Objective function: \( f_{\Phi,C}(x, y) = \sum w_i \phi_i(x, y) - \sum \rho_i d_{C_i}(x, y) \).

Learning w/o Constraints: 300 examples.
Learning w/ 10 Constraints

Constraints are used to bootstrap a semi-supervised learner. Poor model + constraints used to annotate unlabeled data, which in turn is used to keep training the model.

Factored model.
Summary: Constrained Conditional Models

\[ y^* = \arg\max_y \sum w_i \phi(x; y) \]

- Linear objective functions
- Typically \( \phi(x,y) \) will be local functions, or \( \phi(x,y) = \phi(x) \)

- Clearly, there is a joint probability distribution that represents this mixed model.

- We would like to:
  - Learn a simple model or several simple models
  - Make decisions with respect to a complex model

Key difference from MLNs which provide a concise definition of a model, but the whole joint one.
Conclusion

- **Constrained Conditional Models combine**
  - Learning conditional models with using declarative expressive constraints
  - Within a constrained optimization framework

- **Use constraints! The framework supports:**
  - A clean way of incorporating constraints to bias and improve decisions of supervised learning models
    - Significant success on several NLP and IE tasks (often, with ILP)
  - A clean way to use (declarative) prior knowledge to guide semi-supervised learning

---

**LBJ (Learning Based Java):** [http://L2R.cs.uiuc.edu/~cogcomp](http://L2R.cs.uiuc.edu/~cogcomp)

A modeling language for Constrained Conditional Models. Supports programming along with building learned models, high level specification of constraints and inference with constraints
Questions?

- Thank you