Guiding Semi-Supervision with Constraint-Driven Learning

Ming-Wei Chang[1] Lev Ratinov[2] Dan Roth[3]

[1]Department of Computer Science
University of Illinois at Urbana-Champaign

Paper presentation by: Drew Stone

Outline

1. Background
   - Semi-supervised learning
   - Constraint-driven learning
2. Constraint-driven learning with semi-supervision
   - Introduction
   - Model
   - Scoring
   - Constraint pipeline
   - Constraint Driven Learning (CODL) algorithm
3. Experiments

Outline

- **Question**: When labelled data is scarce, how can we take advantage of unlabelled data for training?
- **Intuition**: If there exists structure in the underlying distribution of samples, points close/clustered to one another may share labels

Semi-supervised learning

Framework

- Given a model $\mathcal{M}$ trained on labelled samples from a distribution $\mathcal{D}$ and an unlabelled set of examples $U \subseteq \mathcal{D}^m$
- Learn labels for each example in $U \longrightarrow \mathit{Learn}(U, \mathcal{M})$ and re-use examples (now labelled) to tune $\mathcal{M} \longrightarrow \mathcal{M}^*$

- Given a model $\mathcal{M}$ trained on labelled samples from a distribution $\mathcal{D}$ and an unlabelled set of examples $U \subseteq \mathcal{D}^m$
- Learn labels for each example in $U \longrightarrow Learn(U, \mathcal{M})$ and re-use examples (now labelled) to tune $\mathcal{M} \longrightarrow \mathcal{M}^*$
- **Benefit**: Access to more training data

Semi-supervised learning

Framework

- Given a model $\mathcal{M}$ trained on labelled samples from a distribution $\mathcal{D}$ and an unlabelled set of examples $U \subseteq \mathcal{D}^m$
- Learn labels for each example in $U \longrightarrow Learn(U, \mathcal{M})$ and re-use examples (now labelled) to tune $\mathcal{M} \longrightarrow \mathcal{M}^*$
- **Benefit**: Access to more training data
- **Drawback**: Learned model might drift from correct classifier if the assumptions on the distribution do not hold.

Outline

Constraint-driven learning

Introduction

- **Motivation**: Keep the learned model simple by using constraints to balance over-simplicity

Constraint-driven learning

Introduction

- **Motivation**: Keep the learned model simple by using constraints to balance over-simplicity
- **Benefits**: Simple models (less features) are more computationally efficient

- **Motivation**: Keep the learned model simple by using constraints to balance over-simplicity
- **Benefits**: Simple models (less features) are more computationally efficient
- **Intuition**: Fix a set of task-specific constraints to enable the use of a simple machine learning model but encode task-specific constraints to make both learning easier and more correct.

Constraint-driven learning

Framework

- Given an objective function

$$\mathbf{argmax}_y \; \boldsymbol{\lambda} \cdot F(x, y)$$

- Define the set of linear (non-linear) constraints $\{C_i\}_{i \leq k}$

$$C_i : \mathcal{X} \times \mathcal{Y} \longrightarrow \{0, 1\}$$

- Solve the optimization problem given the constraints

- Given an objective function

$$\mathbf{argmax}_y\ \boldsymbol{\lambda} \cdot F(x, y)$$

- Define the set of linear (non-linear) constraints $\{C_i\}_{i \leq k}$

$$C_i : \mathcal{X} \times \mathcal{Y} \longrightarrow \{0, 1\}$$

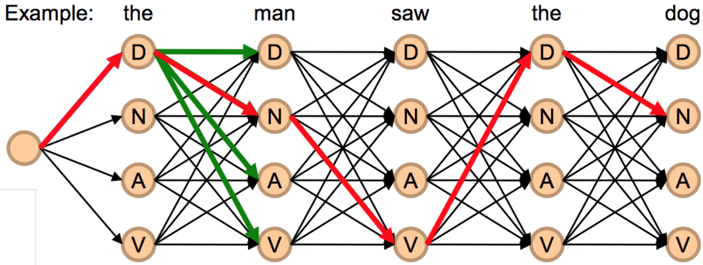
- Solve the optimization problem given the constraints
- **Any Boolean rule can be encoded as a set of linear inequalities.**

Constraint-driven learning

Example task

- Sequence tagging with HMM/CRF + global constraints

$$\mathbf{y}^* = \operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}} P(y_0)P(x_0|y_0)\prod_{i=1}^{n-1} P(y_i|y_{i-1})P(x_i|y_i)$$



Every assignment to the *y*'s is a path.

Constraint-driven learning

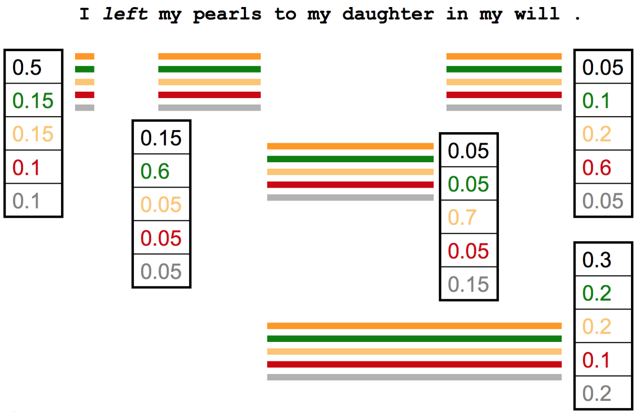
Sequence tagging constraints

- Unique label for each word
- Edges must for a path
- A verb must exist

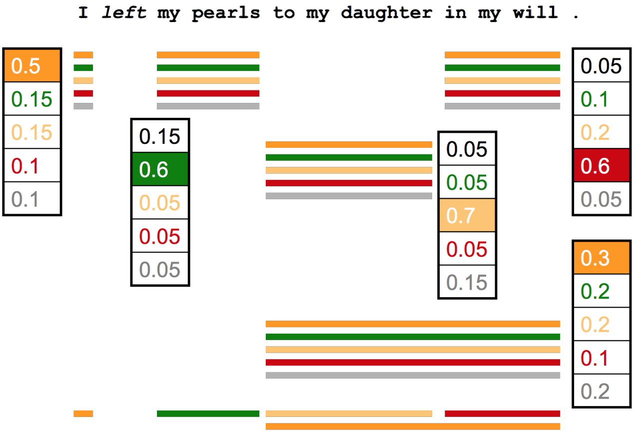
Constraint-driven learning

Example task

- Semantic role labelling with independent classifiers + global constraints

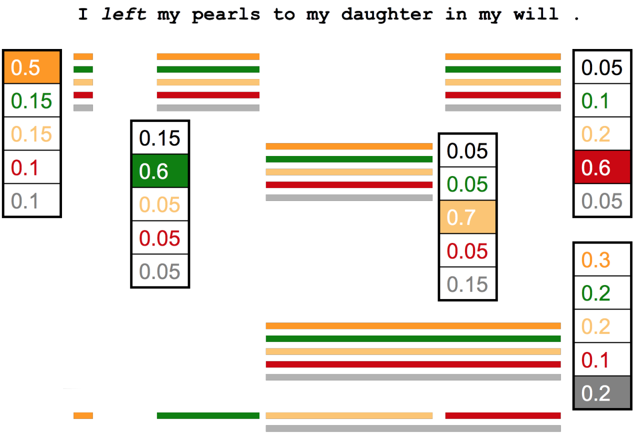


- Each verb predicate carries with it a new inference problem



Constraint-driven learning

Example task



Constraint-driven learning

SRL constraints

- No duplicate argument classes
- Unique labels
- Order constraints

Outline

Constraint-drive learning w/ semi-supervision

Introduction

- **Intuition**: Use constraints to provide better training samples from our unlabelled set of samples

Constraint-drive learning w/ semi-supervision
Introduction

- **Intuition**: Use constraints to provide better training samples from our unlabelled set of samples
- **Benefit**: Deviations from simple model only do so towards a more expressive answer, since constraints guide the model

- Consider the constraint over $\{0,1\}^n$ of 1^*0^*. For every possible sequence $\{1^*010^*\}$, there are 2 good fixes $\{1^*110^*, 1^*000^*\}$. What is the best approach for learning?
- Consider the constraint, state transitions must occur on punctuation marks. There could be many good sequences abiding by this constraint.

Outline

- Consider a data and output domain $\mathcal{X}$, $\mathcal{Y}$ and a distribution $\mathcal{D}$ defined over $\mathcal{X} \times \mathcal{Y}$
- Input/output pairs $(x, y) \in \mathcal{X} \times \mathcal{Y}$ are given as sequence pairs:

$$x = (x_1, \cdots, x_N),\ y = (y_1, \cdots, y_M)$$

- We wish to find a structured output classifier $h : \mathcal{X} \longrightarrow \mathcal{Y}$ that uses a structured scoring function $f : \mathcal{X} \times \mathcal{Y} \longrightarrow \mathbb{R}$ to choose the most likely output sequence, where the correct output sequence is given the highest score
- We are given (or define) a set of constraints $\{C_i\}_{i \leq K}$

$$C_i : \mathcal{X} \times \mathcal{Y} \longrightarrow \{0, 1\}$$

Outline

Constraint-driven learning w/ semi-supervision

Scoring

- Scoring rule:

$$f(x, y) = \sum_{i=1}^{M} \lambda_i f_i(x, y) = \boldsymbol{\lambda} \cdot F(x, y) \equiv \boldsymbol{w}^T \cdot \phi(x, y)$$

- Compatible for any linear discriminative and generative model such as HMMs and CRFs
- Local feature functions $\{f_i\}_{i \leq M}$ allow for tractable inference by capturing contextual structure
  - Space of structured pairs could be huge
  - Locally there are smaller distances to account for

Outline

Constraint-driven learning w/ semi-supervision

Sample constraints

(a)-Citations

1) Each field must be a consecutive list of words, and can appear at most once in a citation.
2) State transitions must occur on punctuation marks.
3) The citation can only start with author or editor.
4) The words *pp., pages* correspond to *PAGE*.
5) Four digits starting with 20xx and 19xx are *DATE*.
6) Quotations can appear only in titles.
7) The words *note, submitted, appear* are *NOTE*.
8) The words *CA, Australia, NY* are *LOCATION*.
9) The words *tech, technical* are *TECH_REPORT*.
10) The words *proc, journal, proceedings, ACM* are *JOURNAL* or *BOOKTITLE*.
11) The words *ed, editors* correspond to *EDITOR*.

- Define $\mathbf{1}_{C(x)} \subseteq \mathcal{Y}$ as the set of output sequences that assign the value 1 for a given (x, y)
- Our objective function then becomes

$$\mathbf{argmax}_{y \in \mathbf{1}_{C(x)}} \boldsymbol{\lambda} \cdot F(x, y)$$

Constraint-driven learning w/ semi-supervision

Constraint pipeline (hard)

- Define $\mathbf{1}_{C(x)} \subseteq \mathcal{Y}$ as the set of output sequences that assign the value 1 for a given (x, y)
- Our objective function then becomes

$$\mathbf{argmax}_{y \in \mathbf{1}_{C(x)}} \boldsymbol{\lambda} \cdot F(x, y)$$

- **Intuition**: Find best output sequence y that maximizes the score, fitting the hard constraints

Constraint-driven learning w/ semi-supervision

Constraint pipeline (soft)

- Given a suitable distance metric between an output sequence and the space of outputs respecting a single constraint
- Given a set of soft constraints $\{C_i\}_{i \leq K}$ with penalties $\{\rho_i\}_{i \leq K}$, we get a new objective function:

$$\textbf{argmax}_y \ \boldsymbol{\lambda} \cdot F(x, y) - \sum_{i=1}^{K} \rho_i \cdot d(y, \mathbf{1}_{C_i(x)})$$

Constraint-driven learning w/ semi-supervision

Constraint pipeline (soft)

- Given a suitable distance metric between an output sequence and the space of outputs respecting a single constraint
- Given a set of soft constraints $\{C_i\}_{i \leq K}$ with penalties $\{\rho_i\}_{i \leq K}$, we get a new objective function:

$$\mathbf{argmax}_y \ \boldsymbol{\lambda} \cdot F(x, y) - \sum_{i=1}^{K} \rho_i \cdot d(y, \mathbf{1}_{C_i(x)})$$

- **Goal**: Find best output sequence under our model that violates the least amount of constraints

Constraint-driven learning w/ semi-supervision

Constraint pipeline (soft)

- Given a suitable distance metric between an output sequence and the space of outputs respecting a single constraint
- Given a set of soft constraints $\{C_i\}_{i \le K}$ with penalties $\{\rho_i\}_{i \le K}$, we get a new objective function:

$$\textbf{argmax}_y \ \boldsymbol{\lambda} \cdot F(x, y) - \sum_{i=1}^{K} \rho_i \cdot d(y, \mathbf{1}_{C_i(x)})$$

- **Goal**: Find best output sequence under our model that violates the least amount of constraints
- **Intuition**: Given a learned model, $\boldsymbol{\lambda} \cdot F(x, y)$, we can bias its decisions using the amount by which the output sequence violates each soft constraint
- **Option**: Use minimal Hamming distance to a sequence

$$d(y, \mathbf{1}_{C_i(x)}) = \textbf{min}_{y' \in C_i(x)} H(y, y')$$

Constraint-driven learning w/ semi-supervision

Distance example

Lars	Ole	Andersen	.
AUTH	AUTH	EDITOR	EDITOR
$\Phi_c(y_1)=0$	$\Phi_c(y_2)=0$	$\Phi_c(y_3)=1$	$\Phi_c(y_4)=0$

Lars	Ole	Andersen	.
AUTH	BOOK	EDITOR	EDITOR
$\Phi_c(y_1)=0$	$\Phi_c(y_2)=1$	$\Phi_c(y_3)=1$	$\Phi_c(y_4)=0$

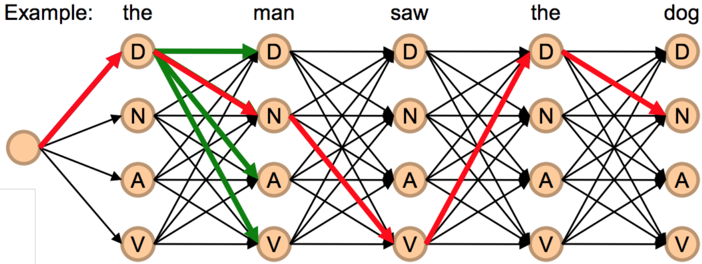
- Taken from *CCM-NAACL-12-Tutorial*

$$d(y, \mathbf{1}_{C_i(x)}) = \sum_{j=1}^{M} \phi_{C_i}(y_j) \longrightarrow \text{count violations}$$

Constraint-driven learning

Recall: sequence tagging

$$\mathbf{y}^* = \operatorname*{argmax}_{\mathbf{y} \in \mathcal{Y}} P(y_0) P(x_0 | y_0) \prod_{i=1}^{n-1} P(y_i | y_{i-1}) P(x_i | y_i)$$



Every assignment to the *y*'s is a path.

Constraint-driven learning

Recall: sequence tagging

$$\text{maximize} \quad \sum_{y \in \mathcal{Y}} \lambda_{0,y} 1_{\{y_0 = y\}} + \sum_{i=1}^{n-1} \sum_{y \in \mathcal{Y}} \sum_{y' \in \mathcal{Y}} \lambda_{i,y,y'} 1_{\{y_i = y \,\wedge\, y_{i-1} = y'\}}$$

$$\lambda_{0,y} = \log(P(y)) + \log(P(x_0|y))$$
$$\lambda_{i,y,y'} = \log(P(y|y')) + \log(P(x_i|y))$$

subject to

$$\sum_{y \in \mathcal{Y}} 1_{\{y_0 = y\}} = 1$$

Unique label for each word

$$\forall y, \quad 1_{\{y_0 = y\}} = \sum_{y' \in \mathcal{Y}} 1_{\{y_0 = y \,\wedge\, y_1 = y'\}}$$

$$\forall y, i > 1 \quad \sum_{y' \in \mathcal{Y}} 1_{\{y_{i-1} = y' \,\wedge\, y_i = y\}} = \sum_{y'' \in \mathcal{Y}} 1_{\{y_i = y \,\wedge\, y_{i+1} = y''\}}$$

Edges that are chosen must form a path

$$\cancel{1_{\{y_0 = \text{"NN"}\}} = 1} \qquad \cancel{1_{\{y_0 = \text{"DT"} \,\wedge\, y_1 = \text{"JJ"}\}} = 1}$$

$$\cancel{1_{\{y_0 = \text{"DT"} \,\wedge\, y_1 = \text{"JJ"}\}} = 1} \qquad \cancel{1_{\{y_1 = \text{"NN"} \,\wedge\, y_2 = \text{"VB"}\}} = 1}$$

Constraint-driven learning

Recall: sequence tagging

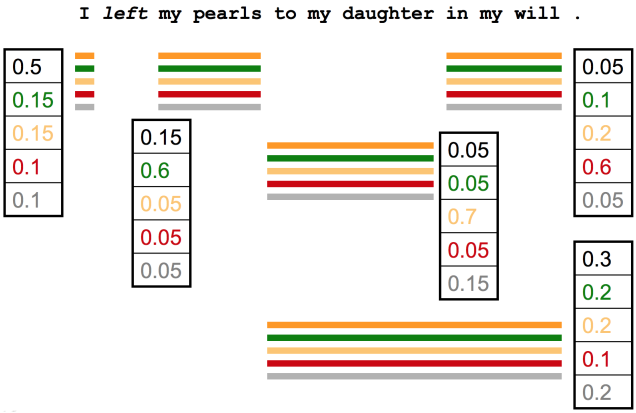
(a) [*AUTHOR* Lars Ole Andersen .] [*TITLE* Program analysis and specialization for the C programming language .] [*TECH-REPORT* PhD thesis ,] [*INSTITUTION* DIKU , University of Copenhagen ,] [*DATE* May 1994 .]

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

- (a) correct citation parsing
- (b) HMM predicted citation parsing
- Adding punctuation state transition constraint returns correct parsing under the same HMM

Constraint-driven learning

Recall: SRL



Constraint-driven learning

Recall: SRL

$$\text{maximize} \quad \sum_{i=0}^{n-1} \sum_{y \in \mathcal{Y}} \lambda_{\mathbf{x}_i,y} 1_{\{y_i=y\}}$$

$$\text{where} \quad \lambda_{\mathbf{x},y} = \lambda \cdot F(\mathbf{x},y) = \lambda_y \cdot F(\mathbf{x})$$

subject to

$$\forall i, \sum_{y \in \mathcal{Y}} 1_{\{y_i=y\}} = 1$$

$$\forall y \in \mathcal{Y}, \sum_{i=0}^{n-1} 1_{\{y_i=y\}} \leq 1$$

$$\forall y \in \mathcal{Y}_R, \sum_{i=0}^{n-1} 1_{\{y_i=y=\text{“R-Ax”}\}} \leq \sum_{i=0}^{n-1} 1_{\{y_i=\text{“Ax”}\}}$$

$$\forall j, y \in \mathcal{Y}_C, 1_{\{y_j=y=\text{“C-Ax”}\}} \leq \sum_{i=0}^{j} 1_{\{y_i=\text{“Ax”}\}}$$

Constraint-driven learning

Recall: SRL

I *left* my pearls to my daughter in my will .

[I]$_{A0}$ *left* [my pearls]$_{A1}$ [to my daughter]$_{A2}$ [in my will]$_{AM\text{-}LOC}$.

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

I *left* my pearls to my daughter in my will .

Outline

Constraint-driven learning w/ semi-supervision

CODL algorithm

Input:

$Cycles$: learning cycles
$Tr = \{x, y\}$: labeled training set.
U: unlabeled dataset
F: set of feature functions.
$\{\rho_i\}$: set of penalties.
$\{C_i\}$: set of constraints.
γ: balancing parameter with the supervised model.
$learn(Tr, F)$: supervised learning algorithm
Top-K-Inference:
returns top-K labeled scored by the cost function (1)

CODL:

```
1.  Initialize λ0 = learn(Tr, F).
2.  λ = λ0.
3.  For Cycles iterations do:
4.      T = φ
5.      For each x ∈ U
6.          {(x, y^1), ..., (x, y^K)} =
7.              Top-K-Inference(x, λ, F, {C_i}, {ρ_i})
8.          T = T ∪ {(x, y^1), ..., (x, y^K)}
9.      λ = γλ0 + (1 − γ)learn(T, F)
```

- Constraints are used in inference not learning
- Maximum likelihood estimation of $\boldsymbol{\lambda}$ (EM algorithm)
- Semi-supervision occurs on line 7,9 (i.e. for each unlabelled sample, we generate K training examples)
- All unlabelled samples are re-labelled each training cycle, unlike "self-training" perspective

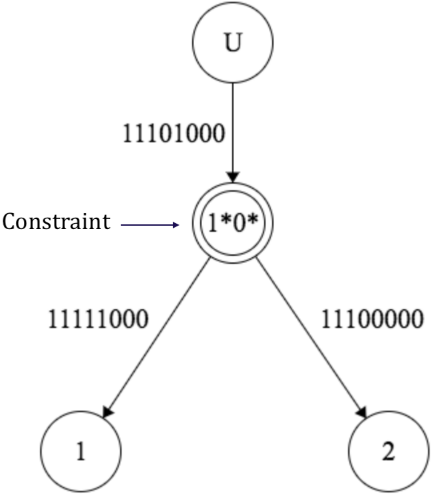
Constraint-driven learning w/ semi-supervision

Expectation Maximization: Top K-hard EM

- EM procedures typically assign a distribution over all input/output pairs for a given unlabelled $x \in \mathcal{X}$
- Instead, we choose top $K, \ y \in \mathcal{Y}$ maximizing our soft objective function and assign uniform probability to each output
- Constraints mutate distribution each step

Constraint-driven learning w/ semi-supervision

Motivation for $K > 1$



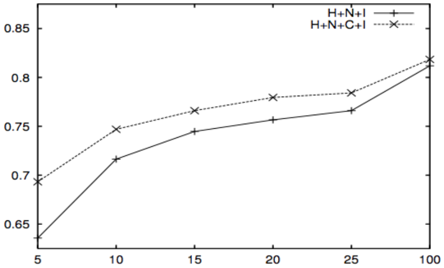
- There may be multiple ways to correct constraint-violating samples
- We have access to more training data

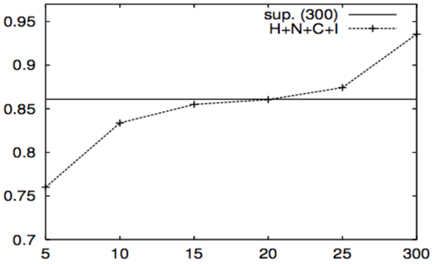
Experiments

Citations

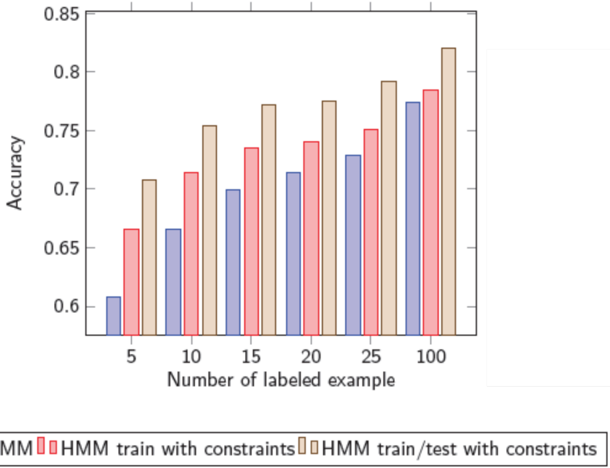
(a)- Citations

N	Inf.	**sup.**	**H**	**H&W**	**H&W&C** (Top-1)	**H&W&C** (Top-K)
5	no I	55.1	60.9	63.6	70.6	71.0
	I	66.6	69.0	72.5	76.0	77.8
300	no I	86.1	80.7	87.1	88.2	88.2
	I	92.5	89.6	93.4	93.6	93.5





Experiments
HMM



Pictures taken from:

- *CCM-NAACL-12-Tutorial*
- Chang, M. W., Ratinov, L., & Roth, D. (2007, June). Guiding semi-supervision with constraint-driven learning. In ACL (pp. 280-287).