Deep Learning With Constraints

Slides by Yatin Nandwani

Learning with Constraints: *Motivation*

➔ Modern day AI == Deep Learning (DL) [Learn from Data]

Learning with Constraints: *Motivation*

- ➔ Modern day AI == Deep Learning (DL) [Learn from Data]
- ➔ Can we inject symbolic knowledge in Deep Learning? E.g.

  Person => Noun [Learn from ~~Data~~ Knowledge]*(credit: Vivek S Kumar)*

Learning with Constraints: *Motivation*

- ➔ Modern day AI == Deep Learning (DL) [Learn from Data]
- ➔ Can we inject symbolic knowledge in Deep Learning? E.g. Person => Noun [Learn from ~~Data~~ Knowledge]
- ➔ **Constraints:** One of the ways of representing symbolic knowledge. $\mathbb{1}\{y_{PER.} = 1\} \implies \mathbb{1}\{y_{Noun.} = 1\}$

Learning with Constraints: *Running Example*

- **Task:** Fine Grained Entity Typing

Learning with Constraints: *Running Example*

Input: Bag of Mentions

Sample Mention: “*Barack Obama is the President of the United States*”

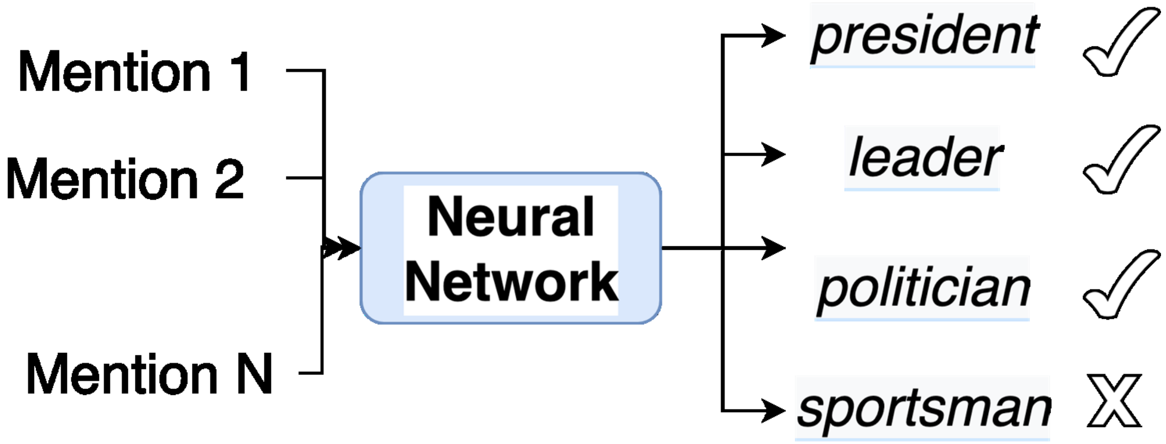
Output: *president, leader, politician...*

Learning with Constraints: *Running Example*

Input: Bag of Mentions

Sample Mention: *“Barack Obama is the President of the United States”*

Output: *president, leader, politician...*

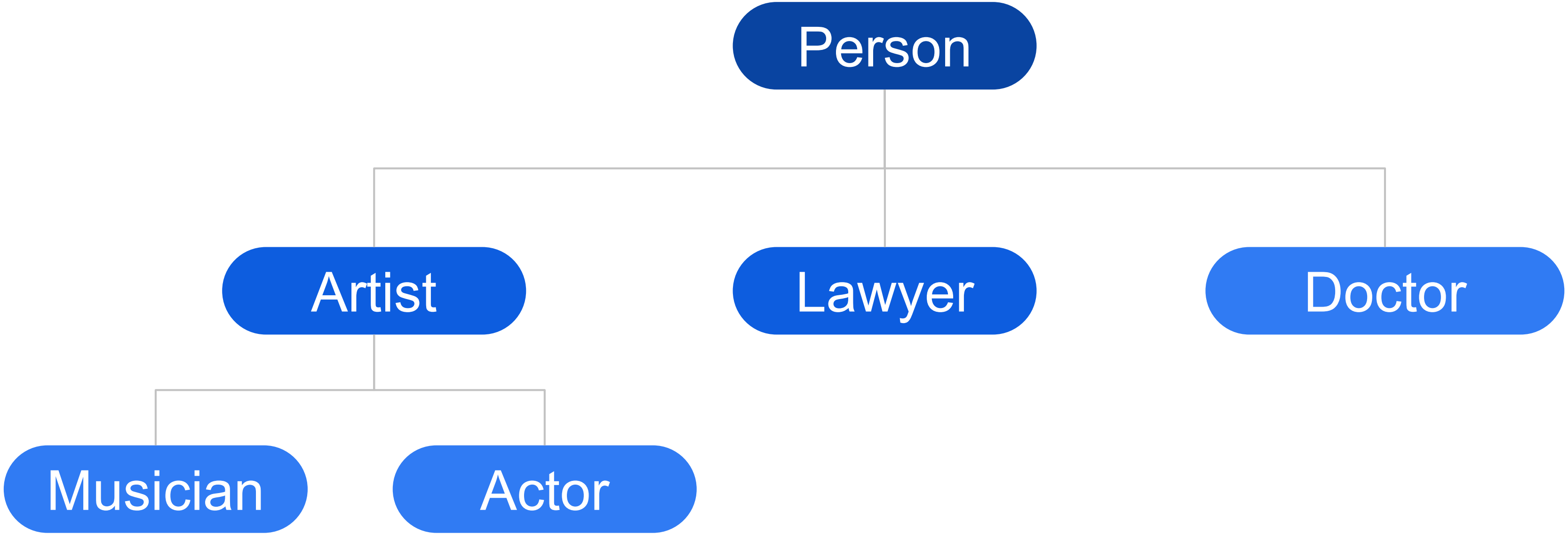


Learning with Constraints: *Running Example*

- **Constraints:** Hierarchy on Output label space

Learning with Constraints: *Running Example*

- **Constraints:** Hierarchy on Output label space



Learning with Constraints: *Representation of Constraints*

➔ Using Soft Logic

$$\mathbb{1}\{y_{ARTIST} = 1\} \implies \mathbb{1}\{y_{PERSON} = 1\}$$

Learning with Constraints: *Representation of Constraints*

→ Using Soft Logic

$$\mathbb{1}\{y_{ARTIST} = 1\} \implies \mathbb{1}\{y_{PERSON} = 1\}$$

$$(\neg\mathbb{1}\{y_{ARTIST} = 1\}) \vee (\mathbb{1}\{y_{PERSON} = 1\})$$

Learning with Constraints: *Representation of Constraints*

→ Using Soft Logic

$$\mathbb{1}\{y_{\mathit{ARTIST}} = 1\} \implies \mathbb{1}\{y_{\mathit{PERSON}} = 1\}$$

$$(\neg \mathbb{1}\{y_{\mathit{ARTIST}} = 1\}) \vee (\mathbb{1}\{y_{\mathit{PERSON}} = 1\})$$

$$(1 - p(y_{\mathit{ARTIST}})) + p(y_{\mathit{PERSON}})$$

Le

Cc

→ l

Boolean Expression	T-norm: Choice 1	T-norm: Choice 2
v	$p(v=1)$	
$\neg v$	$1-p(v=1)$	
$v_1 \vee v_2$	$\min(p(v_1=1)+p(v_2=1),1)$	$\max(p(v_1=1),p(v_2=1))$
$v_1 \wedge v_2$	$\max(p(v_1=1)+p(v_2=1)-1,0)$	$\min(p(v_1=1),p(v_2=1))$

$$\mathbb{1}\{y_{\textsc{artist}}=1\} \implies \mathbb{1}\{y_{\textsc{person}}=1\}$$

$$(\neg\mathbb{1}\{y_{\textsc{artist}}=1\}) \vee (\mathbb{1}\{y_{\textsc{person}}=1\})$$

$$(1-p(y_{\textsc{artist}}))+p(y_{\textsc{person}})$$

Learning with Constraints: *Representation of Constraints*

$$1 - p(y_{ARTIST}) + p(y_{PERSON}) = 1$$

Learning with Constraints: *Representation of Constraints*

$$1 - p(y_{ARTIST}) + p(y_{PERSON}) = 1$$

$$1 - p(y_{ARTIST}) + p(y_{PERSON}) \geq 1$$

Learning with Constraints: *Representation of Constraints*

$$1 - p(y_{ARTIST}) + p(y_{PERSON}) = 1$$

$$1 - p(y_{ARTIST}) + p(y_{PERSON}) \geq 1$$

Equivalently:

$$p(y_{ARTIST}) - p(y_{PERSON}) \leq 0$$

Learning with Constraints: *Representation of Constraints*

Define:

$$f_k^i = p(y_{ARTIST}) - p(y_{PERSON})$$

kth Constraint

ith Data point

Inequality Constraint:

$$f_k^i \leq 0$$

Learning with Constraints: *Formulation*

Unconstrained Problem

$$\min_{w} L(w)$$

$L(w)$: Any standard loss function, say Cross Entropy

Learning with Constraints: *Formulation*

Unconstrained Problem

$$\min_{w} L(w)$$

$L(w)$: Any standard loss function, say Cross Entropy

Constrained Problem

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \leq 0; \;\; \forall 1 \leq i \leq m; \;\; \forall 1 \leq k \leq K$$

Learning with Constraints: *Formulation*

Constrained Problem

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \leq 0; \;\; \forall 1 \leq i \leq m; \;\; \forall 1 \leq k \leq K$$

Where:

m: Size of training data

K: Number of Constraints

Learning with Constraints: *Formulation*

Constrained Problem

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \le 0, \; \forall 1 \le i \le m, \; \forall 1 \le k \le K$$

$$\mathcal{L}(w, \Lambda) = L(w) + \sum_{i=1}^{m} \sum_{k=1}^{K} \lambda_k^i f_k^i(w)$$

Learning with Constraints: *Formulation*

Constrained Problem

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \leq 0; \;\; \forall 1 \leq i \leq m; \;\; \forall 1 \leq k \leq K$$

Where:

m: Size of training data

K: Number of Constraints

Issue:

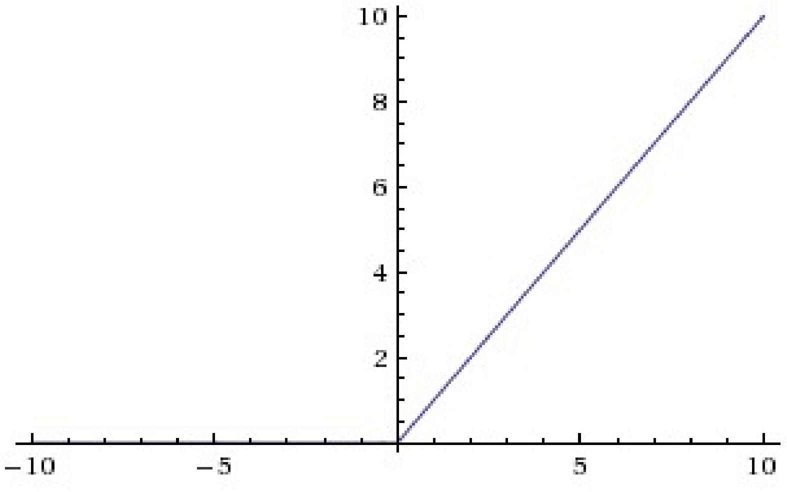
O(mK) #constraints

i.e. ***mK*** Lagrange Multipliers!

Learning with Constraints: *Reduce # Constraints*

$$H(c) = c \ \ for \ \ c \geq 0, \quad \text{and} \quad 0 \quad \text{for} \quad c < 0$$

H(c)



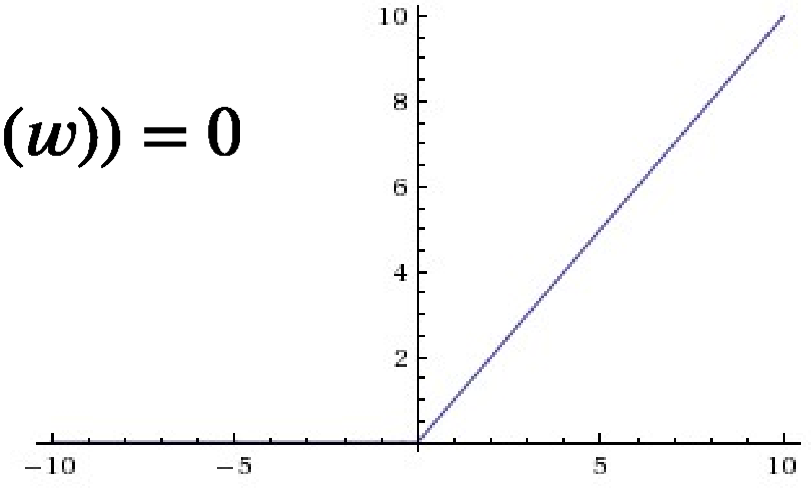
Learning with Constraints: *Reduce # Constraints*

$$H(c) = c \;\; for \;\; c \geq 0, \quad \text{and} \quad 0 \quad \text{for} \quad c < 0$$

$$f_k^i(w) \leq 0 \qquad \equiv \qquad H(f_k^i(w)) = 0$$

Equivalent

H(c)



Learning with Constraints: *Reduce # Constraints*

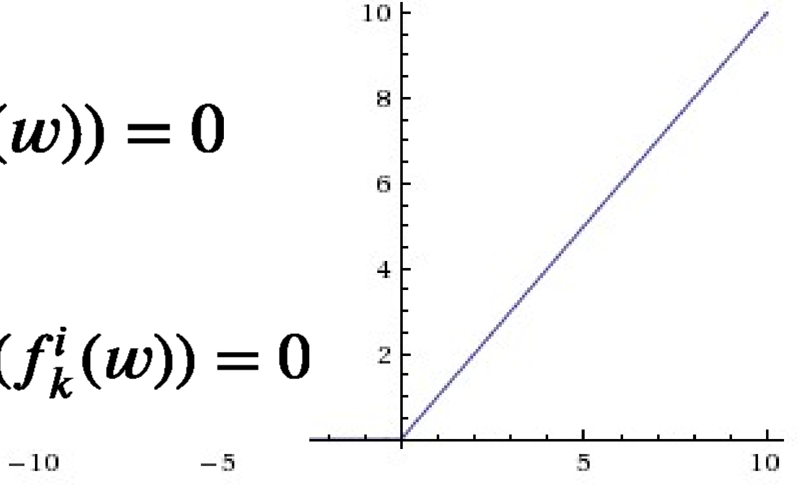
$$H(c) = c \quad for \quad c \geq 0, \quad \text{and} \quad 0 \quad \text{for} \quad c < 0$$

$$f_k^i(w) \leq 0 \qquad \equiv \qquad H(f_k^i(w)) = 0$$

Equivalent

$$\forall i : H(f_k^i(w)) = 0 \qquad \equiv \qquad \sum_i H(f_k^i(w)) = 0$$

H(c)



Learning with Constraints: *Reduce # Constraints*

Originally:

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \leq 0; \;\; \forall 1 \leq i \leq m; \;\; \forall 1 \leq k \leq K$$

Learning with Constraints: *Reduce # Constraints*

Originally:

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \leq 0; \;\; \forall 1 \leq i \leq m; \;\; \forall 1 \leq k \leq K$$

Now:

Define: $h_k(w) = \sum_i H(f_k^i(w))$

$$\min_{w} L(w) \quad \text{subject to} \quad h_k(w) = 0; \;\; \forall 1 \leq k \leq K$$

Learning with Constraints: *Reduce # Constraints*

Originally:

$$\min_{w} L(w) \quad \text{subject to} \quad f_k^i(w) \leq 0; \;\; \forall 1 \leq i \leq m; \;\; \forall 1 \leq k \leq K$$

Now:

Define: $h_k(w) = \sum_i H(f_k^i(w))$

O(K) #constraints

$$\min_{w} L(w) \quad \text{subject to} \quad h_k(w) = 0; \;\; \forall 1 \leq k \leq K$$

Learning with Constraints

$$\min_{w} L(w) \quad \text{subject to} \quad h_k(w) = 0; \quad \forall 1 \leq k \leq K$$

$$\mathcal{L}(w; \Lambda) = L(w) + \sum_{k=1}^{K} \lambda_k h_k(w)$$

Learning with Constraints: *Experiments*

Typenet

	MAP Scores			Constraint Violations		
Scenario	5% Data	10% Data	100% Data	5% Data	10% Data	100% Data
B	68.6			22,715		
B+H	68.71			22,928		
B+C						
B+S						

Learning with Constraints: *Experiments*

Typenet

	MAP Scores			Constraint Violations		
Scenario	**5% Data**	**10% Data**	**100% Data**	**5% Data**	**10% Data**	**100% Data**
B	68.6			22,715		
B+H	68.71			22,928		
B+C	80.13			**25**		
B+S	**82.22**			41		

Learning with Constraints: *Experiments*

Typenet

	MAP Scores			Constraint Violations		
Scenario	**5% Data**	**10% Data**	**100% Data**	**5% Data**	**10% Data**	**100% Data**
B	68.6	69.2	70.5	22,715	21,451	22,359
B+H	68.71	69.31	71.77	22,928	21,157	24,650
B+C	80.13	81.36	**82.80**	**25**	45	**12**
B+S	**82.22**	**83.81**		41	**26**	

Semi-Supervised Learning

- Supervised Data

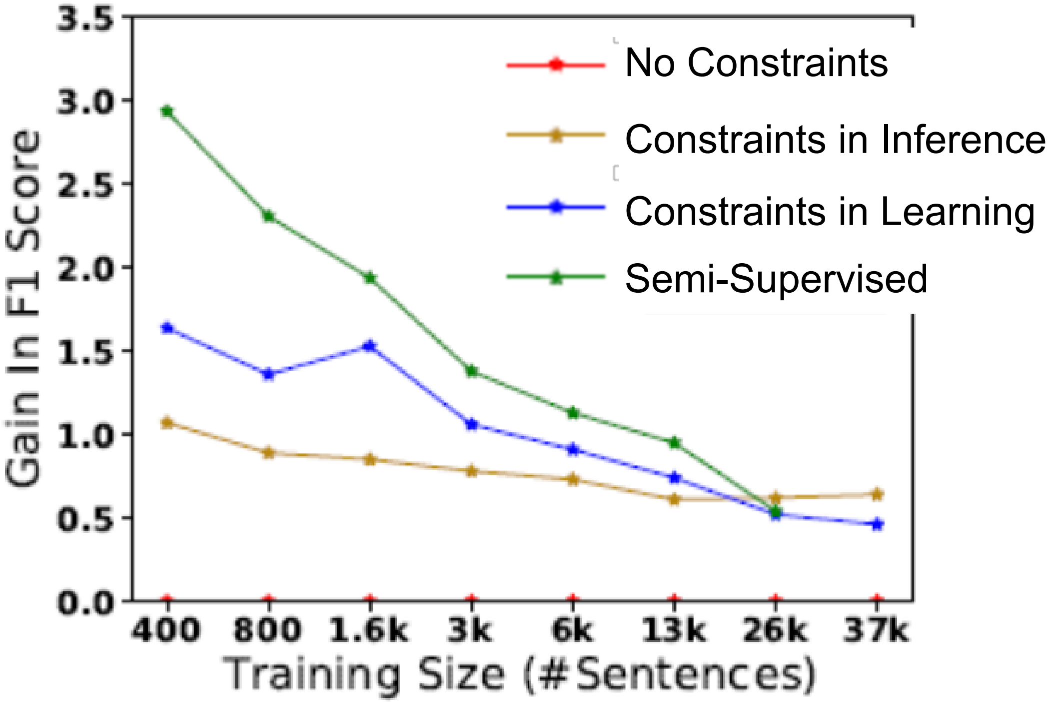
$$\mathcal{L}(w;\Lambda) = L(w) + \sum_{k=1}^{K} \lambda_k h_k(w)$$

- Unsupervised Data

$$\mathcal{L}(w;\Lambda) = \sum_{k=1}^{K} \lambda_k h_k(w)$$

Results (Multi Task NER-POS)

[Nandwani et al, NeurIPS 2019]



(a) Avg. Gain in F1 Score Over Baseline.

Test Time

	Test Time
Constraints in Training	115 sec
Constraints in Inference	2,895 sec

More Results

[Nandwani et al, NeurIPS 2019]

- Fine-Grained Entity Typing

% Data	5%	10%	100%	5%	10%	100%
Baseline	68.6	69.2	70.5	22,715	21,451	22,359
Const. L	**78.4**	**80.6**	**83.5**	**186**	**95**	**97**

- Semantic Role Labeling

% Data	1%	5%	10%	1%	5%	10%
Baseline	62.7	72.6	75.3	19,317	11,718	10,570
Const. L	**66.0**	**73.7**	**76.0**	**9,231**	**6,436**	**6,140**

More Results

[Kolluru et al, EMNLP 2020, Gupta et al, ArXiv 2022]

- Open Information Extraction

Algos	AUC	F1
Baseline	33.7	52.4
Constrained Learning	**35.7**	**54**

- Info. Extraction from Tables in Research Papers

Algos	ID F1	Tuple F1	Mat. F1
GNN	78.7	69.3	60.9
Constrained Learning of GNN	**82.4**	**70.1**	**63.5**