Light-Supervision of Structured Prediction Energy Networks

Andrew McCallum

Pedram Rooshenas
Oregon PhD → UMass Postdoc

Aishwarya Kamath
UMass MS

David Belanger
UMass PhD → Google Brain

Greg Druck
UMass PhD → Yummly

SPENs [2016]
Generalized Expectation [Mann; Druck 2010-12]
Light-Supervision

Prior Knowledge as *Generalized Expectation*

...induces extra structural dependencies...

Structured Prediction

Complex dependencies with *SPENs*
Chapter 1

Generalized Expectation
Learning from small labeled data
Leverage unlabeled data
Family 1: Expectation Maximization

[Dempster, Laird, Rubin, 1977]
Family 2: Graph-Based Methods

[Szummer, Jaakkola, 2002]  [Zhu, Ghahramani, 2002]
Family 3: Auxiliary-Task Methods

[Ando and Zhang, 2005]
Family 4: Boundary in Sparse Region

*Transductive SVMs* [Joachims, 1999]: Sparsity measured by margin

*Entropy Regularization* [Grandvalet & Bengio, 2005]: minimize label entropy
Family 4: Boundary in Sparse Region

Transductive SVMs
[Joachims, 1999]: Sparsity measured by margin

Entropy Regularization
[Grandvalet & Bengio, 2005]: minimize label entropy

best solution?

Label Proportions
Student
Faculty

50

E[p(y)]
E[p(y|f(x))]

Label | Feature Expectations
Label Prior Expectations

[Ma, McCallum 2010; Druck, Mann, McCallum 2011, Druck McCallum 2012]

Entropy Regularization
[Grandvalet & Bengio, 2005]: minimize label entropy

best solution?
Expectations on Labels | Features
Classifying *Baseball* versus *Hockey*

**Traditional**

- Human Labeling Effort

(Semi-)Supervised Training via Maximum Likelihood

**Generalized Expectation**

- Brainstorm a few Keywords

<table>
<thead>
<tr>
<th>ball</th>
<th>field</th>
<th>bat</th>
<th>puck</th>
<th>ice</th>
<th>stick</th>
</tr>
</thead>
</table>

Semi-Supervised Training via Generalized Expectation

p(HOCKEY | “puck”) = .9
Labeling Features

~1000 unlabeled examples

features labeled . . .

hockey
base
HR
Mets

goal
Buffalo
Leafs
puck
Lemieux

ball
Oilers
Sox
Pens
runs

batting
base
NHL
Bruins
Penguins

Accuracy
85%
92%
94.5%
96%
Accuracy per Human Effort

![Graph showing test accuracy over labeling time]

- Labeling features
- Labeling instances

Test accuracy vs. Labeling time in seconds
## Prior Knowledge

### Feature labels from humans

**baseball**/**hockey** classification

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>baseball</td>
<td>hockey</td>
</tr>
<tr>
<td>hit</td>
<td>puck</td>
</tr>
<tr>
<td>braves</td>
<td>goal</td>
</tr>
<tr>
<td>runs</td>
<td>nhl</td>
</tr>
</tbody>
</table>

### many other sources

**resources on the web**

**data from related tasks**

Generalized Expectation (GE)

\[ O(x) = S(E\tilde{p}(x)) \left[ E p(y|x; g(x, y)) \right] \]

returns 1 if \( x \) contains “hit” and \( y \) is **baseball**
Generalized Expectation (GE)

\[ \mathbb{E}_p(y|x; \theta) \left[ g(x, y) \right] \]

assume general CRF \[ Lafferty \text{ et al. } 01 \]
\[ p(y|x; \theta) = \frac{1}{Z_{\theta, x}} \exp \left( \theta^\top f(x, y) \right) \]

model probability of baseball if \( x \) contains “hit”
Generalized Expectation (GE)

\[ E_{\tilde{p}(x)} \left[ E_{p(y|x;\theta)} \left[ g(x, y) \right] \right] \]

(empirical distribution)

(can be defined as)
model’s probability that documents that contain “hit” are labeled **baseball**
Generalized Expectation (GE)

(soft) expectation constraint

\[ S(\mathbb{E}_{\tilde{p}(x)}[\mathbb{E}_p(y|x;\theta)[g(x, y)]])) \]

score function

larger score if model expectation matches prior knowledge
Generalized Expectation (GE)

Objective Function

\[ \mathcal{O}(\theta) = S\left( \mathbb{E}_{\tilde{p}(x)} \left[ \mathbb{E}_{p(y|x;\theta)} \left[ g(x, y) \right] \right] \right) + r(\theta) \]
GE Score Functions

\[ \mathcal{O}(\theta) = S(E_{\tilde{p}(x)}[E_{p(y|x;\theta)}[g(x, y)]])) + r(\theta) \]

Squared error:

\[ S_{l^2}(\theta) = -||\hat{g} - g_{\theta}||^2_2 \]

KL divergence:

\[ S_{KL}(\theta) = -\sum_q \hat{g}_q \log \frac{\hat{g}_q}{g_{\theta,q}} \]
Estimating Parameters with GE

\[ \mathcal{O}(\theta) = S(\mathbb{E}_{\tilde{p}(x)}[\mathbb{E}_p(y|x;\theta)[g(x, y)]]) + r(\theta) \]

violation term: \[ \text{KL: } v_i = \frac{\hat{g}_i}{g_{\theta i}} \]
sq. error: \[ v_i = -2(\hat{g}_i - g_{\theta i}) \]

\[ \nabla_{\theta} \mathcal{O}(\theta) = v^\top \left( \mathbb{E}_{\tilde{p}(x)}[\mathbb{E}_p(y|x;\theta)[g(x, y)f(x, y)^\top]] \right. \]

violation

\[ -\mathbb{E}_p(y|x;\theta)[g(x, y)] \mathbb{E}_p(y|x;\theta)[f(x, y)^\top] \] + \nabla_{\theta} r(\theta) \]

estimated covariance between model and constraint features
Learning About Unconstrained Features

- **Unlabeled** puck
- **Hit** puck

GE

Trained Model

- **Hit**
- **Run**
- **Pitcher**
- **Goal**
- **NHL**

Generalizes beyond prior knowledge

Learned through covariance
Generalized Expectation criteria

Easy communication with domain experts

• Inject domain knowledge into parameter estimation

• Like “informative prior”...

• ...but rather than the “language of parameters” (difficult for humans to understand)

• ...use the “language of expectations” (natural for humans)
IID Prediction

“classification” e.g. logistic regression

Example: Spam Filtering

Predicted

Y

Observed

X
Structured Prediction

e.g. “sequence labeling”

Chinese Word Segmentation

\[ O(\theta) = S(\mathbb{E}_{\tilde{p}(x)}[\mathbb{E}_{p(y|x;\theta)}[g(x, y)]])) + r(\theta) \]

Linear-chain CRF

\[ \nabla \mathbf{E} \sum_{\mathbf{y}} \sum_{i} \sum_{j} p(y_{i-1}, y_{i}, y_{j} | x; \theta)g(x, y_{j}, j)f(x, y_{i-1}, y_{i}, i) \]

marginal over three, non-consecutive positions
Natural Expectations lead to Difficult Training Inference

"AUTHOR field should be contiguous, only appearing once."

Anna Popescu (2004), "Interactive Clustering,

\[ p(y_{i-1}, y_i, y_j, y_k) \] The downfall of GE.
Chapter 2

A framework providing easier inference for complex dependencies?

Structured Prediction Energy Networks

Deep Learning
+
Structured Prediction
Structured Prediction

“classification” e.g. logistic regression

Example: Spam Filtering

\[
P(\text{Spam}| \mathbf{X}) = \max_{\text{Spam}} P(\text{Spam}, \mathbf{X})
\]

\[
P(\text{Not Spam}| \mathbf{X}) = \max_{\text{Not Spam}} P(\text{Not Spam}, \mathbf{X})
\]
Structured Prediction

*Example: Chinese Word Segmentation*

\[ E(Y;X) \]

\[ Y = \text{argmin}_Y E(Y;X) \]
Structured Prediction

e.g. “sequence labeling”

Example: Chinese Word Segmentation

\[ E(Y,Y) \]

\[ E(Y;X) \]

Feature Engineering

\[ X \]

Chinese People

Example: Chinese Word Segmentation

Y

Start

Not Start

Start

Not Start

中

国

人

民

C h i n e s e

P e o p l e
Structured Prediction

e.g. “sequence labeling”

Example: Chinese Word Segmentation

E(Y,Y)

Y

Start — Not Start — Start — Not Start

E(Y;X)

Feature Engineering

X

Chinese People

Example: Chinese Word Segmentation
Structured Prediction

e.g. “sequence labeling”

Example: Chinese Word Segmentation

E(Y,Y)

E(Y,Z;X)

Feature Engineering

Y

Not Start

Start

Not Start

Start

Not Start

Z₁

Z₂

Z₃

Z₄

X

中

国

人

民

“Hidden Unit Conditional Random Fields”
Maaten, Welling, Saul, AISTATS 2011
Structured Prediction

e.g. “sequence labeling”

Example: Chinese Word Segmentation

\[ E(Y,Y) \]

\[ E(Y,Z;X) \]

Feature Engineering

- Start
- Not Start
- Start
- Not Start

- \( Z_1 \)
- \( Z_2 \)
- \( Z_3 \)
- \( Z_4 \)
Structured Prediction

e.g. “sequence labeling”

Example: Chinese Word Segmentation

\[ E(Y, Y) \]

Dependency structure

\[ E(Y, Z; X) \]

Feature Engineering

Y

Start

Not Start

Start

Not Start

X

中

国

人

民

Chinese People

Example: Chinese Word Segmentation

羅穆尼頭號對手桑托倫在三州勝選，而金瑞契只贏得喬治亞州的初選。羅穆尼面臨的一大挑戰是，其他共和黨總統參選人目前均表
Structured Prediction

Example: Chinese Word Segmentation

\[ E(Y,Y) \]
Dependency structure

\[ E(X,Z..,Y) \]
Feature Engineering
Structured Prediction

E.g. "multi-label classification"

Example: Multi-label Document Classification

E(Y,Y)
Dependency structure

Y

E(X,Y)
Feature Engineering

barley  gold  wheat  zinc

LONDON, March 3 - The U.K. Exported 535,460 tonnes of wheat and 336,750 tonnes of barley in January, the Home Grown Cereals Authority (HGCA) said, quoting adjusted Customs and Excise figures. Based on the previous January figures issued on February 9, wheat exports increased by nearly 64,000 tonnes and barley by about 7,000 tonnes. The new figures bring cumulative wheat exports for the period July 1/February 13 to 2.99 mln tonnes, and barley to 2.96
Structured Prediction
e.g. “multi-label classification”

Example: Multi-label Image Classification

\[ E(Y,Y) \]
Dependency structure

\[ Y \]

\[ E(X,Y) \]
Feature Engineering

[Diagram showing a graph with nodes labeled 'road', 'fish', 'tree', and 'desk']
Structured Prediction

Example: Scene Understanding

$E(Y,Y)$
Dependency structure

$X$

$E(X,Y)$
Feature Engineering
Structured Prediction

Example: Scene Understanding

$E(Y,Y)$
Dependency structure

$E(X,Y)$
Feature Engineering

• **Expressivity** of dependencies
• **Parsimony** of parameterization
• **Tractability** of inference
Structured Prediction

Example: Scene Understanding

$E(Y,Y)$
Dependency structure

$Y$

$E(X,Y)$
Feature Engineering

$X$
Structured Prediction

Example: Scene Understanding

$E(Y,Y)$
Dependency structure

$Y$

$E(X,Y)$
Feature Engineering
Structured Prediction

Example: Scene Understanding

\[ E(Y,Y) \]
Dependency structure

\[ E(X,Y) \]
Feature Engineering
Structured Prediction

Example:

Scene Understanding

E(Y,Y)

Dependency structure

Y

E(X,Y)

Feature Engineering

X

\[ m^{(t+1)}_{i \rightarrow j}(x_j) = \sum_{x_i} \Phi_{ij}(x_i, x_j) \Phi_i(x_i) \prod_{k \in N(i), j} m^{(t)}_{k \rightarrow i}(x_i) \]
Bayesian Network

Deep Learning

Sparsely connected
Hand-designed representations
Loopy/iterated inference (typically)
Cautious about capacity
“Statistically conscientious”

Densely connected (learn connectivity)
Learned, distributed representations
Feed-forward inference (typically)
Wild about high capacity
“Wild West” 😬
$y = \sigma \left( \sum_{i} w_{1i} x_i \right) \sigma \left( W_2 \sigma \left( W_1 x \right) \right)$

$z_2 = \sigma \left( W_2 z_1 \right)$

$z_{11} = \sigma \left( \sum_{i} z_1 w_{1i} \sigma \left( W_1 x \right) \right)_{z_{11}}$
Deep Learning

\[ y = F(x; W) \]

Training Data
\[ \{x^{(i)}, y^{(i)}\}^{N}_{i=1} \]

Loss
\[ L = \sum_{i} L \left( F(x^{(i)}; W), y^{(i)} \right) \]
e.g. Squared error, Cross-entropy,…

Training
\[ \arg \min_{W} L \]

\[ W_{\text{new}} = W_{\text{old}} - \alpha \frac{\partial L(W)}{\partial W} \]

Key tools:
1. Back-propagation
2. Stochastic gradient descent
\[ y = F(x; W) \]
Back-propagation

\[ y = \sigma \left( W_3 \sigma \left( W_2 \sigma \left( W_1 x \right) \right) \right) \]

The “chain rule”

\[ \frac{\partial g \circ f}{\partial x} = \frac{\partial g}{\partial f} \cdot \frac{\partial f}{\partial x} \]

\[ \frac{\partial j \circ i \circ h \circ g \circ f}{\partial x} = \frac{\partial j}{\partial i} \frac{\partial i}{\partial h} \frac{\partial h}{\partial g} \frac{\partial g}{\partial f} \frac{\partial f}{\partial x} \]
Can get gradient of \textit{Loss} wrt parameters at any depth from 
(1) local partial derivative functions 
(2) numeric gradient from above
Example: CNNs for Object Classification in Images

Motivation for SPENs

1. Use power of deep learning for structure learning

2. Provide an alternative to graphical models.

3. Black-box interaction with model.
Structured Prediction Energy Networks

\[ E(Y,Y) = \Psi_0[y_0, y_1] + \Psi_1[y_1, y_2] + \Psi_2[y_2, y_3] \]

\[ Y \in \{0,1\} \]

\[ E(X,Z,..,Y) \]
Structured Prediction Energy Networks

$E(y,y)$

$E(y,z;x)$

[Belanger, McCallum, ICML 2016]
Structured Prediction Energy Networks

$E(y, y, z; x)$
Structured Prediction Energy Networks

Energy network

\[ E(\bar{y}; F(x)) \]

Soft prediction… found by gradient descent

\[ \bar{y}^* = \arg \min_{\bar{y} \in [0,1]^L} E(\bar{y}; F(x)) \]

Relax \( y \), to be continuous

\[ y \in \{0, 1\}^L \rightarrow \bar{y} \in [0, 1]^L \]

Feature Network

\[ F(x) \]

Energy network

\[ E(y, y, z; x) \]

Soft prediction found by gradient descent

\[ \bar{y}^* = \arg \min_{\bar{y} \in [0,1]^L} E(\bar{y}; F(x)) \]

Relax \( y \), to be continuous

\[ y \in \{0, 1\}^L \rightarrow \bar{y} \in [0, 1]^L \]

Feature Network

\[ F(x) \]

Energy network

\[ E(y, y, z; x) \]
SPEN Inference Graph

Inference Step 1
- \( y_0 \)
  - \( E(y_0), \frac{\partial E}{\partial y_0} \)

Inference Step 2
- \( y_1 \)
  - \( E(y_1), \frac{\partial E}{\partial y_1} \)

Inference Step 3
- \( y_2 \)

Repeat...

Gradient step
- \( E(y_0), \frac{\partial E}{\partial y_0} \)
- \( E(y_1), \frac{\partial E}{\partial y_1} \)
Inference Step 1
- \( \text{cached features} \)
  - \( \text{feature network} \)
  - \( \text{initialization network} \)
  - \( \text{energy network} \)
  - \( y_0 \)

Inference Step 2
- \( y_1 \)
- \( \text{energy network} \)
- \( \frac{\partial E}{\partial y_1} \)
- \( E(y_1) \)

Inference Step 3
- \( y_2 \)
- \( \text{energy network} \)
- \( \frac{\partial E}{\partial y_2} \)
- \( E(y_2) \)

Repeat...
- \( \text{gradient step} \)
- \( \text{gradient step} \)
Gradient used to Modify Inputs

“A Neural Algorithm for Artistic Style”
[Gatys et al. 2015]

SPENs use similar idea:
Optimize energy using backprop all the way down to the raw pixels.
Learning Algorithm 1: Structured SVM

Training Loss: $\mathcal{L} = \sum_{x^{(i)}, y^{(i)}} \max_{\bar{y}} \left[ \Delta(y^{(i)}, \bar{y}) - \left( E_{\bar{y}}(\bar{y}; x^{(i)}) - E_{\bar{y}}(y^{(i)}; x^{(i)}) \right) \right] +$ 

Penalty must be differentiable

Model’s energy difference

search requires

Loss-Augmented Inference

$$\arg \min_{\bar{y}} \left( -\Delta(y^{(i)}, \bar{y}) + E_{\bar{y}}(\bar{y}; x^{(i)}) \right)$$

Penalty must be differentiable

Stochastic Gradient

$$\frac{\partial \mathcal{L}}{\partial W}$$

(Taskar et al., 2004; Tsochantaridis et al., 2004)
Learning Algorithm 2: End-to-end “backprop through inference”

Training Loss:

\[
\mathcal{L} = \sum_{i} \sum_{i} L \left( \mathbf{y}^{(i)}, \text{Algorithm}^{(i)} \right)_{\mathbf{W}} \left( \mathbf{x}^{(i)}, \bar{y}^{(i)} \right)
\]

Direct Risk Minimization

\[
\bar{y}^* = \bar{y}^{[0]} + \sum_{t=1}^{T} \alpha_t \frac{\partial}{\partial \bar{y}} EW \left( \mathbf{x}, \bar{y}^{[t-1]} \right)
\]

sum over “time steps” of inference

\[
\frac{\partial \mathcal{L}}{\partial \mathbf{W}} = \frac{\partial \mathcal{L}}{\partial \bar{y}^*} \frac{\partial \bar{y}^*}{\partial \mathbf{W}} = \sum_{t=1}^{T} \alpha_t \frac{\partial \mathcal{L}}{\partial \bar{y}^*} \left( \frac{\partial}{\partial \mathbf{W}} \frac{\partial}{\partial \bar{y}} EW \left( \mathbf{x}, \bar{y}^{[t-1]} \right) \right)
\]

sum over “time steps” of inference

Direct application of:

Justin Domke, AISTATS, 2012. "Generic Methods for Optimization-Based Modeling"

Belanger, McCallum, ICML 2017
Learning Algorithm 2 Graph

\[ \frac{\partial L}{\partial y} \]

\[ L(y, y^{(i)}) \]

Hessian-vector product

Domke, 2012. *Generic Methods for Optimization Based Modeling*
Chapter 3

Light Supervision training of Structured Prediction Energy Networks

(Turing complete!)

1. Human writes arbitrary prior knowledge (SPEN)
2. Learn model with arbitrary dependencies.
3. Efficient inference by gradient descent.
Human writes arbitrary prior knowledge…

“AUTHOR field should be contiguous, only appearing once.”

…as a scoring function $V(x=\text{citation}, \ y=\text{labeling})$

\[
\text{score} = 0 \\
\text{score} -= 1 \text{ foreach AUTHOR non-contiguous} \\
\text{score} -= 1 \text{ if has both JOURNAL & BOOKTITLE} \\
\text{score} -= 1 \text{ foreach “using” not in TITLE} \\
\text{score} -= 1 \text{ foreach [A–Z]\.
not AUTHOR|EDITOR} \\
\text{score} -= 1 \text{ if PUBLISHER before JOURNAL . . .}
\]

(like rule-based AI before ML was popular)
Why use ML if we get a ruled-based scoring function?

- Doesn’t generalize
  - examines just a few features
  - SPENs will learn correlated features, labels.

- No inference procedure just scores for given \((x,y)\)
  - stochastic optimization is slow
  - SPENs provide gradient-descent inference
Learning Algorithm 3: “ranking successive gradient steps”

\[
\text{Training Loss} = \sum_{x \in \mathcal{D}} [\alpha(V(y_h, x) - V(y_l, x)) - E_w(y_h, x) + E_w(y_l, x)]^+
\]
Preliminary Experiments

(...much more work and comparisons in future...)
Weak-Sup SPEN: simple test
Multi-label Document Classification

\(x = \text{Medical bag-of-words}\)

\[
\text{[amount, cystourethrogram, diagnosed, episode, evaluate, exam, fever, grade, growth, hematuria, infection, interval, kidney, left, lower, occurred, patient, pole, previously, purpose, reflux, renal, scar, scarring, small, study, tract, urinary, vesicoureteral, voiding, year]}
\]

\(y = \text{multiple ICM-9-CD codes}\)

\[
\text{[593-70, 599-00]}
\]

\(x = \text{Human background knowledge}\)

Keyword descriptions of ICM-9-CD codes. (Not gathering any labeled correlation knowledge.)

- 593-70: vesicoureteral, reflux, unspecified, nephropathy
- V79-99: viral, chlamydial, infection, conditions, unspecified
- 753-00: renal, agenesis, dysgenesis

Scoring function gives +1 for each label:keyword cooccurrence.

\[
V(y^i, x^i) = \sum_j I(l_j \in y^i) I(|x^i \cap w_j| > 0) - \gamma \max(|y^i| - 1, 0)
\]

Label, Keyword matches

Sparsity constraint
Does the SPEN generalize over the human scoring function?

ICM-9-CD code data set, evaluate **F1 of label set**

<table>
<thead>
<tr>
<th>Human Scoring Function, Exhaustive Search</th>
<th>SPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>N(\leq 1)</td>
<td>N(\leq 2)</td>
</tr>
<tr>
<td>15.5</td>
<td>18.3</td>
</tr>
</tbody>
</table>

(~10x faster)
Weak-Sup SPEN: better test Citation Field Extraction

\[ x = \text{Citation Token Sequence} \]


\[ y = \text{Seq. of Labels } \in |14| \]

AUTHOR AUTHOR YEAR TITLE TITLE EDITOR, EDITOR EDITOR BOOKTITLE, BOOKTITLE PUBLISHER PUBLISHER LOCATION

\[ x = \text{Human background knowledge} \]

Human-written scoring function. 50 lines of code. Written in \(~1\) hour.

score \(\leftarrow 1\) foreach AUTHOR non-contiguous
score \(\leftarrow 1\) if has both JOURNAL & BOOKTITLE
score \(\leftarrow 1\) foreach “using” not in TITLE

\(~4000\) unlabeled examples, 0 labeled.

**Scoring function advice:**

- Penalties only, so 0 = best.
- Can use varying magnitudes, -1, -5, -10.
- Debug with some stochastic optimization.
### Citation Field Extraction Accuracy

<table>
<thead>
<tr>
<th>Method</th>
<th>Token accuracy</th>
<th>Time sec/citation</th>
<th>Ave. V() score</th>
</tr>
</thead>
<tbody>
<tr>
<td>GE [Mann &amp; McCallum '10]</td>
<td>37%</td>
<td>?</td>
<td>N/A</td>
</tr>
<tr>
<td>V search 10</td>
<td>34%</td>
<td>14</td>
<td>-1.86</td>
</tr>
<tr>
<td>V search 100</td>
<td>39%</td>
<td>170</td>
<td>-0.98</td>
</tr>
<tr>
<td>V search 1000</td>
<td>42%</td>
<td>1240</td>
<td>-0.62</td>
</tr>
<tr>
<td>SPEN</td>
<td>52%</td>
<td>0.0008</td>
<td>~ -20</td>
</tr>
</tbody>
</table>

![Graph showing average score over training iterations](image)

---

**Example text**


**V search 100 output**

```
<table>
<thead>
<tr>
<th>AUTHOR</th>
<th>TITLE</th>
<th>AUTHOR</th>
<th>NOTE</th>
<th>DATE</th>
<th>PAGES</th>
</tr>
</thead>
</table>
```

**SPEN output**

```
<table>
<thead>
<tr>
<th>AUTHOR</th>
<th>TITLE</th>
<th>TITLE</th>
<th>TITLE</th>
<th>TITLE</th>
<th>PAGES</th>
</tr>
</thead>
</table>
```
Related Work

• Deep Value Networks…
  [Gygli, Norouzi, Angelova 2017 ICML]
  - Matching magnitude (rather than just ranking).
  - Hurts accuracy? 5% vs SPEN’s 52%

• Constraint-Driven Learning
  [Chang, Ratinov, Roth 2007 ACL]
  - Supervised training ➔ Pseudo-label data w/ constraints

• Snorkel: Rapid Training Data Creation with Weak Supervision
  [Ratner, Bach, Ehrenberg, Fries, Wu, Ré 2017 VLDB]
  - Rules ➔ Pseudo-labeled data ➔ Supervised (self) training

• Label-Free Supervision of NNs w/ … Domain Knowledge
  [Stewart, Ermon 2017 AAAI]
  - Constraints ➔ Loss function ➔ Train feed-forward NN.
GE Related Work

- **Generalized Expectation**
  - Distribution Matching: Quadrianto et al. (2009)
  - Constraint Driven Learning: Carlson et al. (2010)

- **Posterior Regularization**
  - MAP approximation
  - MAP approximation
  - Variational approximation; Jensen’s inequality
  - Gracă, Ganchev, Taskar (2007)

- **Measurements**
  - Variational approximation
  - Log $\mathbb{E}[p_N(b|\phi)] \approx \log p_N(b|\mathbb{E}[\phi])$
  - Liang, Jordan, Klein (2009)

- **Related Work**
  - Carlson et al. (2010)
  - Graça, Ganchev, Taskar (2007)
  - Mann, Druck, McCallum (2007)
  - Liang, Jordan, Klein (2009)
Summary

• **Generalized Expectation**
  • Learning from unlabeled data + “labeled features”
  • Hard to do inference

• **Structured Prediction Energy Networks**
  • Representation learning for output variables
  • Test-time inference by gradient descent
  • New SPEN training method: Ranking

• **Experiments**
  • Multi-label Classification: ICM-9
  • Sequence labeling: Citation field extraction

• **Next**
  • Training on corpus-wide expectations.
  • Interactive tools for score function development.