Learning the Structure of Generative Models without Labeled Data

Stephen Bach, Bryan He, Alex Ratner, and Chris Ré
Stanford University

Motivation: Generative Models for Weak Supervision
- Curating training data has become the biggest bottleneck when developing machine learning applications
- Weak supervision sources, such as heuristic rules, distant supervision, and weak classifiers, are much less expensive
- However, weak supervision sources are noisy and conflict
- Idea: encapsulate weak supervision sources as labeling functions and model their behavior to denoise their outputs
- By modeling the true class as a latent variable, we can estimate generative parameters without ground truth

Learning Pipeline

Example Application: Information Extraction

Task: identify mentions of chemicals causing diseases in scientific literature

<table>
<thead>
<tr>
<th>Chemical</th>
<th>Disease</th>
<th>Misc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.3</td>
<td>0.2</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0.7</td>
</tr>
<tr>
<td>3</td>
<td>0.8</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Example Labeling Functions

- **LF_heuristic(x):**
  - `m = re.match('.*caused.*', x.sentence)`
  - return True if m else None

- **LF_distant_supervision(x):**
  - `in_kb = (x.chemical, x.disease) in ctd`
  - return True if in_kb else None

Empirical Results

<table>
<thead>
<tr>
<th>Application</th>
<th>Ind. F1</th>
<th>Struct. F1</th>
<th>F1 Diff</th>
<th># LF</th>
<th># Dep.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disease Tagging</td>
<td>66.3</td>
<td>68.9</td>
<td>+2.6</td>
<td>233</td>
<td>315</td>
</tr>
<tr>
<td>Chem-Disease</td>
<td>54.6</td>
<td>55.9</td>
<td>+1.3</td>
<td>33</td>
<td>21</td>
</tr>
<tr>
<td>Device Polarity</td>
<td>88.1</td>
<td>88.7</td>
<td>+0.6</td>
<td>12</td>
<td>32</td>
</tr>
</tbody>
</table>

Consistent improvements to information extraction models trained on labels estimated from generative models with learned structure

These experiments used existing labeling functions, demonstrating that modeling structure can even improve the performance of carefully developed weak supervision sources

Sample Complexity

<table>
<thead>
<tr>
<th>Selecting Correlations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selecting Overlapping Correlations</td>
</tr>
</tbody>
</table>

Efficient, exact gradient computation leads to two-order-of-magnitude speedup over MLE with Gibbs sampling

Reduction of learning time to seconds enables human-in-the-loop development of labeling functions

Analysis: Sample Complexity

**Challenge:** Marginal pseudolikelihood is nonconvex, but previous analyses of l1-regularized parameter estimation for structure learning rely on Lagrangian duality

**Assumptions:**
1. Feasible set of parameters that contains the true model
2. Over the feasible set, conditioning on a labeling function provides more information than marginalizing it out

**Theorem:** For pairwise dependencies, such as correlations, samples are sufficient to recover true dependency structure over n labeling functions with probability at least 1 - δ.

Our approach: maximize l1-regularized marginal pseudolikelihood

Since the marginal pseudolikelihood objective has only one target variable and one latent variable, efficient to compute gradient exactly

We optimize for each LF and add the dependencies with nonzero weight to the generative model

Structure Learning for Generative Models

- When domain experts write labeling functions, they often introduce statistical dependencies among them
- Incorrectly modeling dependencies leads to inaccurate estimation of true, latent classes
- Goal is to quickly identify labeling function dependencies

Key
- Latent Variable
- Target Variable
- Conditioning Variable
- Dependency
- Possible Dependency

**Structure:**

```
  +------------------+
  |                  |
  |                  |
  |  Acc             |
  |                  |
  |  Acc             |
  |                  |
  |  Acc             |
  |                  |
  |  Cor             |
  |                  |
  |  Cor             |
  |                  |
  +------------------+
```

We model functions' behavior to denoise it
We use estimated labels to train a model

Snorkel: Open source implementation available at http://snorkel.stanford.edu

Learning the Structure of Generative Models without Labeled Data

Stephen Bach, Bryan He, Alex Ratner, and Chris Ré
Stanford University

Motivation: Generative Models for Weak Supervision
- Curating training data has become the biggest bottleneck when developing machine learning applications
- Weak supervision sources, such as heuristic rules, distant supervision, and weak classifiers, are much less expensive
- However, weak supervision sources are noisy and conflict
- Idea: encapsulate weak supervision sources as labeling functions and model their behavior to denoise their outputs
- By modeling the true class as a latent variable, we can estimate generative parameters without ground truth

Learning Pipeline

Example Application: Information Extraction

Task: identify mentions of chemicals causing diseases in scientific literature

<table>
<thead>
<tr>
<th>Chemical</th>
<th>Disease</th>
<th>Misc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.3</td>
<td>0.2</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>0.7</td>
</tr>
<tr>
<td>3</td>
<td>0.8</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Example Labeling Functions

- **LF_heuristic(x):**
  - `m = re.match('.*caused.*', x.sentence)`
  - return True if m else None

- **LF_distant_supervision(x):**
  - `in_kb = (x.chemical, x.disease) in ctd`
  - return True if in_kb else None

Empirical Results

<table>
<thead>
<tr>
<th>Application</th>
<th>Ind. F1</th>
<th>Struct. F1</th>
<th>F1 Diff</th>
<th># LF</th>
<th># Dep.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disease Tagging</td>
<td>66.3</td>
<td>68.9</td>
<td>+2.6</td>
<td>233</td>
<td>315</td>
</tr>
<tr>
<td>Chem-Disease</td>
<td>54.6</td>
<td>55.9</td>
<td>+1.3</td>
<td>33</td>
<td>21</td>
</tr>
<tr>
<td>Device Polarity</td>
<td>88.1</td>
<td>88.7</td>
<td>+0.6</td>
<td>12</td>
<td>32</td>
</tr>
</tbody>
</table>

Consistent improvements to information extraction models trained on labels estimated from generative models with learned structure

These experiments used existing labeling functions, demonstrating that modeling structure can even improve the performance of carefully developed weak supervision sources

Sample Complexity

<table>
<thead>
<tr>
<th>Selecting Correlations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selecting Overlapping Correlations</td>
</tr>
</tbody>
</table>

Efficient, exact gradient computation leads to two-order-of-magnitude speedup over MLE with Gibbs sampling

Reduction of learning time to seconds enables human-in-the-loop development of labeling functions

Analysis: Sample Complexity

**Challenge:** Marginal pseudolikelihood is nonconvex, but previous analyses of l1-regularized parameter estimation for structure learning rely on Lagrangian duality

**Assumptions:**
1. Feasible set of parameters that contains the true model
2. Over the feasible set, conditioning on a labeling function provides more information than marginalizing it out

**Theorem:** For pairwise dependencies, such as correlations, samples are sufficient to recover true dependency structure over n labeling functions with probability at least 1 - δ.

Our approach: maximize l1-regularized marginal pseudolikelihood

Since the marginal pseudolikelihood objective has only one target variable and one latent variable, efficient to compute gradient exactly

We optimize for each LF and add the dependencies with nonzero weight to the generative model

Structure Learning for Generative Models

- When domain experts write labeling functions, they often introduce statistical dependencies among them
- Incorrectly modeling dependencies leads to inaccurate estimation of true, latent classes
- Goal is to quickly identify labeling function dependencies

Key
- Latent Variable
- Target Variable
- Conditioning Variable
- Dependency
- Possible Dependency

**Structure:**

```
  +------------------+
  |                  |
  |                  |
  |  Acc             |
  |                  |
  |  Acc             |
  |                  |
  |  Acc             |
  |                  |
  |  Cor             |
  |                  |
  |  Cor             |
  |                  |
  +------------------+
```

We model functions' behavior to denoise it
We use estimated labels to train a model

Snorkel: Open source implementation available at http://snorkel.stanford.edu