Snorkel: Rapid Training Data Creation with Weak Supervision

Michael Zhang, Jacob Klegar, Tomoka Kan

Logo source: https://www.snorkel.org/
Background & Motivation: Weakly Supervised Learning
Current Supervised Learning Paradigm

Collect data (X)
Current Supervised Learning Paradigm

Collect data (X) + Get ground-truth labels
Current Supervised Learning Paradigm

Collect data (X)  Get ground-truth labels
For Deep Learning Applications...

High quality, manually annotated training data
For Deep Learning Applications...

High quality, manually annotated training data

State-of-the-art deep learning model
For Deep Learning Applications...

High quality, manually annotated training data

State-of-the-art deep learning model
For Deep Learning Applications...

High quality, manually annotated training data + State-of-the-art deep learning model = Profit!
Problem: Manual Annotation is Expensive
Problem: Manual Annotation is Expensive

Slow, Costly
Problem: Manual Annotation is Expensive

- Slow
- Costly
- Require Experts
- Error-prone
Problem: Manual Annotation is Expensive

- Slow, Costly
- Require Experts, Error-prone
- Inflexible
Can we use *noisier* training data and still train *high-performance* models?
Can we use *noisier training data* and still *train high-performance models*?
Weakly Supervised Learning
Weakly Supervised Learning

Distant Supervision
Weakly Supervised Learning

Distant Supervision       Other Models
Weakly Supervised Learning

Distant Supervision

Other Models

Crowdsourcing
Weakly Supervised Learning

Distant Supervision

Other Models

Crowdsourcing

Pattern Matching

[^\^]*?
Weakly Supervised Learning

Distant Supervision  Other Models  Crowdsourcing

Pattern Matching  Expert Heuristics

if A then B
Currently done in an ad hoc manner
Snorkel tries to streamline this
Snorkel Pipeline
def label_function(x, a):
    if x < a:
        return True
    elif x > 2 * a:
        return False
    else:
        return None
Snorkel Pipeline

```python
def label_function(x, a):
    if x < a:
        return True
    elif x > 2 * a:
        return False
    else:
        return None
```

Write labeling functions

Generate probabilistic training labels
def label_function(x, a):
    if x < a:
        return True
    elif x > 2 * a:
        return False
    else:
        return None
Step 1: Labeling Functions
Labeling Functions, a Key Abstraction

$x \in \mathcal{X}$

data point
Labeling Functions, a Key Abstraction

\[ x \in \mathcal{X} \rightarrow \lambda \]

data point \hspace{1cm} labeling function
Labeling Functions, a Key Abstraction

\( x \in \mathcal{X} \rightarrow \lambda \rightarrow \tilde{y} \in \mathcal{Y} \cup \mathcal{O} \)

data point \hspace{1cm} labeling function \hspace{1cm} weak label
Labeling Functions, a Key Abstraction

\[ x \in \mathcal{X} \rightarrow \lambda \rightarrow \tilde{y} \in \mathcal{Y} \cup \{0\} \]

data point  labeling function  weak label
Two Main Interfaces
Two Main Interfaces

```python
def label_function(x, a):
    if x < a:
        return True
    elif x > 2 * a:
        return False
    else:
        return None
```

Hand-defined
Two Main Interfaces

Hand-defined

```python
def label_function(x, a):
    if x < a:
        return True
    elif x > 2 * a:
        return False
    else:
        return None
```

Declarative

```
Ontology( ctd, [A, B, -C])
Pattern("{{a}} causes {{b}}")
CustomFn(x: heuristic(x))
```
Two Main Interfaces

Hand-defined

```python
def label_function(x, a):
    if x < a:
        return True
    elif x > 2 * a:
        return False
    else:
        return None
```

Declarative

Iterate over a year of user studies

Ontology(ctd, [A,B,-C])

Pattern("{{{a}}} causes {{{b}}}")

CustomFn(x:heuristic(x))
Data Model Example: Relation Extraction
We study a patient who became quadriplegic after parenteral magnesium administration for preeclampsia.
We study a patient who became **quadriplegic** after parenteral **magnesium** administration for **preeclampsia**.
We study a patient who became quadriplegic after parenteral magnesium administration for preeclampsia.

Context type: chemical
Data Model Example: Relation Extraction

We study a patient who became quadriplegic after parenteral magnesium administration for preeclampsia.

Context type: chemical  Context type: disease
We study a patient who became quadriplegic after parenteral magnesium administration for preeclampsia.

Candidate 1: $x_1 := \text{Causes (magnesium, quadriplegic)}$
We study a patient who became quadriplegic after parenteral magnesium administration for preeclampsia.

Candidate 1: $x_1 := \text{Causes (magnesium, quadriplegic)}$

Candidate 2: $x_2 := \text{Causes (magnesium, preeclampsia)}$
Building the Label Matrix

Datapoints

Labeling Functions

\[ \lambda_1 \]

\[ x_1 \]

\[ \vdots \]

\[ x_n \]
Building the Label Matrix

Datapoints: $x_1, x_n$

Labeling Functions: $\lambda_1, \lambda_2$

Labels: 1 (true), -1 (false), abstain
Building the Label Matrix

Datapoints $\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n$

Labeling Functions $\lambda_1, \lambda_2, \lambda_3$

1 (true) -1 (false) abstain
Building the Label Matrix

Datapoints

Labeling Functions

$\lambda_1 \lambda_2 \lambda_3 \ldots \lambda_n$

$x_1$

$\ldots$

$x_n$

1 (true)
-1 (false)
abstain
Building the Label Matrix

Datapoints

Labeling Functions

\[ \begin{array}{cccccc}
\lambda_1 & \lambda_2 & \lambda_3 & \cdots & \cdots & \lambda_n \\
\hline
x_1 & & & & & \\
\vdots & & & & & \\
\vdots & & & & & \\
\vdots & & & & & \\
x_n & & & & & \\
\end{array} \]

\[ \tilde{y} \]
Building the Label Matrix

Datapoints

Labeling Functions

$\lambda_1$ $\lambda_2$ $\lambda_3$ $\cdots$ $\lambda_n$

$\mathbf{x}_1$

$\mathbf{x}_2$

$\mathbf{x}_n$

1 (true)

-1 (false)

abstain

High correlation

$\tilde{y}$?
Building the Label Matrix

Datapoints $x_1, \ldots, x_n$

Labeling Functions $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n$

Color coding:
- Blue: 1 (true)
- Green: -1 (false)
- White: abstain

High correlation

$\tilde{y}$?

High conflict
Building the Label Matrix

Datapoints

Labeling Functions

1 (true)
-1 (false)
abstain

low coverage

High correlation

High conflict

$\tilde{y}$?
Building the Label Matrix

Datapoints

Labeling Functions

1 (true)
-1 (false)
abstain

low coverage
accuracy?

High correlation

High conflict

\( \tilde{y} \)?
Step 2: Generative Label Model
How to aggregate labeling functions?
How to aggregate labeling functions?

$x_1 \lambda_1 \lambda_2 \lambda_3 \cdots \lambda_n$

Majority Vote
How to aggregate labeling functions?

For an example, consider:

\[ \lambda_1 \lambda_2 \lambda_3 \cdots \lambda_n \]

Let \( x_1 \) be the input vector:

\[
\begin{array}{cccc}
\text{3} & \square & 1 & 1 \\
\end{array}
\]

Using the Majority Vote method:

\[ \tilde{y}_1 = \square \]
How to aggregate labeling functions?

Majority Vote

$\lambda_1 \lambda_2 \lambda_3 \cdots \lambda_n$

$x_1$

3 1 1

$\hat{y}_1 = \text{Blue}$
Majority Vote Issue 1: Double Counting

\[
\begin{array}{cccccc}
\lambda_1 & \lambda_2 & \lambda_3 & \cdots & \cdots & \lambda_n \\
x_1 & & & & & \\
\vdots & & & & & \\
\vdots & & & & & \\
\vdots & & & & & \\
x_n & & & & & \\
\end{array}
\]
Majority Vote Issue 1: Double Counting

\[ \lambda_1 \lambda_2 \lambda_3 \cdots \lambda_n \]

\[ x_1 \]

\[ \vdots \]

\[ x_n \]
Majority Vote Issue 1: Double Counting
Majority Vote Issue 2: Variable Coverage, Accuracy
Majority Vote Issue 2: Variable Coverage, Accuracy

All Data
Majority Vote Issue 2: Variable Coverage, Accuracy

10% labels

100% accuracy
Majority Vote Issue 2: Variable Coverage, Accuracy

\[ \lambda_1 \]

\[ \lambda_2 \]

100% accuracy

10% labels

50% accuracy

100% labels

100% labels
Majority Vote Issue 2: Variable Coverage, Accuracy

\[ \lambda_1 (x) \neq \lambda_2 (x) \]
Majority Vote Issue 2: Variable Coverage, Accuracy

\[ \lambda_1(x) \neq \lambda_2(x) \]

100% labels

10% labels

100% accuracy

50% accuracy

50 - 50?
Majority Vote Issue 2: Variable Coverage, Accuracy

\[ \lambda_1(x) \neq \lambda_2(x) \]

100% accuracy

10% labels

50% accuracy

100% labels

50% accuracy

\[ \wedge \] *? \lambda_3
Majority Vote Issue 2: Variable Coverage, Accuracy

\[ \lambda_1(x) \neq \lambda_2(x) \]
\[ \lambda_2(x) = \lambda_3(x) \]

100% accuracy
10% labels

50% accuracy

100% labels

50% accuracy

\[ [\wedge]^* ? \lambda_3 \]
Majority Vote Issue 2: Variable Coverage, Accuracy

\[ \lambda_1(x) \neq \lambda_2(x) \]
\[ \lambda_2(x) = \lambda_3(x) \]

100% accuracy
10% labels
50% accuracy
100% labels
50% accuracy

Defer to less accurate?
Alternative: Modeling Latent Accuracies

Unweighted Majority Vote
Alternative: Modeling Latent Accuracies

Unweighted Majority Vote

Weighted by Accuracies
Alternative: Modeling Latent Accuracies

Unweighted Majority Vote

Weighted by Accuracies
Alternative: Modeling Latent Accuracies

How to model accuracies without ground-truth labels?
Generative Label Model

$$(x, y) \rightarrow y$$ Latent variable
Generative Label Model

\[(x, y) \rightarrow y\]

Latent variable

\[\lambda_1(x)\]
\[\lambda_2(x)\]
\[\lambda_3(x)\]

Accuracies

Observed LF outputs
Generative Label Model

\[(x, y) \rightarrow y\]

Latent variable

\[\lambda_1(x)\]
\[\lambda_2(x)\]
\[\lambda_3(x)\]

Accuracies

Dependencies

Observed LF outputs
Generative Label Model

Try to maximize probability of LF outputs given generative model
Step 3: Discriminative End Model
Training a Discriminative Model

LSTM

CNN

Conv 1-1  Conv 1-2  Pooling  Conv 2-1  Conv 2-2  Pooling  Conv 3-1  Conv 3-2  Pooling  Conv 3-3  Pooling  Conv 4-1  Conv 4-2  Pooling  Conv 4-3  Pooling  Conv 5-1  Conv 5-2  Pooling  Conv 5-3  Pooling  Dense  Dense  Dense
Why Add A Discriminative Model?
Why Add A Discriminative Model?

+ labeling functions
+ generative model
Why Add A Discriminative Model?

+ labeling functions
+ generative model

still potentially sparse
Why Add A Discriminative Model?

+ labeling functions
+ generative model
+ discriminative model

generalize to other data points
Label Model Tradeoffs
Modeling Label Function Accuracies

Unweighted Majority Vote or More Complicated Model?
Modeling Label Function Accuracies

Low Density
Modeling Label Function Accuracies

**Little Disagreement**

![Diagram showing Little Disagreement]

**Low Density**
Modeling Label Function Accuracies

Little Disagreement

Low Density

High Density
Modeling Label Function Accuracies

Little Disagreement

Consensus Reached

Low Density

High Density
Modeling Label Function Accuracies

Little Disagreement

Labeling Functions

Datapoints

Low Density

Medium Density

High Density

Consensus Reached
Modeling Label Function Accuracies

Little Disagreement

Labeling Functions

Low Density

Medium Density

High Density

Consensus Reached
Modeling Label Function Accuracies

What is medium density?

Medium Density
Modeling Label Function Accuracies

What is medium density?

How many labels per datapoint?

or

Could a weighted vote flip an incorrect unweighted majority vote?
What is medium density?

How many labels per datapoint?

or

Could a weighted vote flip an incorrect unweighted majority vote?

Modeling Helps!

Labeling Functions

Medium Density
Modeling Label Function Correlations

\[ \lambda_1 \lambda_2 \lambda_3 \cdots \lambda_n \]

\( \boldsymbol{x}_1 \)
\( \boldsymbol{x}_n \)

Labels functions

Datapoints

\( \epsilon = 1 \)

Correlations: 0

Predictive Performance: 20

\( \tilde{y} \)
Modeling Label Function Correlations

Datapoints: $x_1, \cdots, x_n$

Labeling Functions: $\lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_n$

Correlations: $\varepsilon = 0.85$

Predictive Performance: 40
Modeling Label Function Correlations

\[ \varepsilon = 0.7 \]

Correlations: 2

Predictive Performance: 60
Modeling Label Function Correlations

Datapoints

Labeling Functions

$\epsilon = 0.55$

Correlations: 3

Predictive Performance: 80
Modeling Label Function Correlations

\[ \epsilon = 0.4 \]

Correlations: 15

Predictive Performance: 81
Modeling Label Function Correlations

Simulated Labeling Functions

- Performance
- # of Correlations
- Elbow Point

Number of Correlations vs. Correlation Threshold
Modeling Label Function Correlations
Modeling Label Function Correlations
Results
Relation Extraction from Text

Scientific Articles

Electronic Health Records

Chemical-Disease relations

Spouses
Relation Extraction from Text

<table>
<thead>
<tr>
<th>Task</th>
<th>Distant Supervision</th>
<th>Snorkel (Gen.)</th>
<th>Snorkel (Disc.)</th>
<th>Hand Supervision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>F1</td>
<td>P</td>
</tr>
<tr>
<td>Chem</td>
<td>11.2</td>
<td>41.2</td>
<td>17.6</td>
<td>78.6</td>
</tr>
<tr>
<td>EHR</td>
<td>81.4</td>
<td>64.8</td>
<td>72.2</td>
<td>77.1</td>
</tr>
<tr>
<td>CDR</td>
<td>25.5</td>
<td>34.8</td>
<td>29.4</td>
<td>52.3</td>
</tr>
<tr>
<td>Spouses</td>
<td>9.9</td>
<td>34.8</td>
<td>15.4</td>
<td>53.5</td>
</tr>
</tbody>
</table>

Snorkel exceeds models trained by distant supervision by an average of 132%.
Relation Extraction from Text

<table>
<thead>
<tr>
<th>Task</th>
<th>Distant Supervision</th>
<th>Snorkel (Gen.)</th>
<th>Snorkel (Disc.)</th>
<th>Hand Supervision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>F1</td>
<td>Lift</td>
</tr>
<tr>
<td>Chem</td>
<td>11.2</td>
<td>41.2</td>
<td>17.6</td>
<td></td>
</tr>
<tr>
<td>EHR</td>
<td>81.4</td>
<td>64.8</td>
<td>72.2</td>
<td></td>
</tr>
<tr>
<td>CDR</td>
<td>25.5</td>
<td>34.8</td>
<td>29.4</td>
<td></td>
</tr>
<tr>
<td>Spouses</td>
<td>9.9</td>
<td>34.8</td>
<td>15.4</td>
<td></td>
</tr>
</tbody>
</table>

Snorkel comes within 2.11% of the F1 score of hand supervision.
Cross-Modal Experiments

Abnormality Detection in Lung Radiographs

Crowdsourcing
Cross-Modal Experiments

<table>
<thead>
<tr>
<th>Task</th>
<th>Snorkel (Disc.)</th>
<th>Hand Supervision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radiology (AUC)</td>
<td>72.0</td>
<td>76.2</td>
</tr>
<tr>
<td>Crowd (Acc)</td>
<td>65.6</td>
<td>68.8</td>
</tr>
</tbody>
</table>

Snorkel comes within 5.08% of the accuracy of hand supervision.
Effect of Generative Modelling

<table>
<thead>
<tr>
<th>Task</th>
<th>Disc. Model on Unweighted LF's</th>
<th>Disc. Model</th>
<th>Lift</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chem</td>
<td>48.6</td>
<td>54.1</td>
<td>+5.5</td>
</tr>
<tr>
<td>EHR</td>
<td>80.9</td>
<td>81.4</td>
<td>+0.5</td>
</tr>
<tr>
<td>CDR</td>
<td>42.0</td>
<td>45.3</td>
<td>+3.3</td>
</tr>
<tr>
<td>Spouses</td>
<td>52.8</td>
<td>54.2</td>
<td>+1.4</td>
</tr>
<tr>
<td>Crowd (Acc)</td>
<td>62.5</td>
<td>65.6</td>
<td>+3.1</td>
</tr>
<tr>
<td>Rad. (AUC)</td>
<td>67.0</td>
<td>72.0</td>
<td>+5.0</td>
</tr>
</tbody>
</table>

Improvement of 5.81% on average
## Impact of Labeling Functions

Adding different types of labeling functions improves predictive performance.

<table>
<thead>
<tr>
<th>LF Type</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>Lift</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text Patterns</td>
<td>42.3</td>
<td>42.4</td>
<td>42.3</td>
<td></td>
</tr>
<tr>
<td>+ Distant Supervision</td>
<td>37.5</td>
<td>54.1</td>
<td>44.3</td>
<td>+2.0</td>
</tr>
<tr>
<td>+ Structure-based</td>
<td>38.8</td>
<td>54.3</td>
<td>45.3</td>
<td>+1.0</td>
</tr>
</tbody>
</table>
User Study

How quickly can users learn to write labelling functions?

Is writing labelling functions more time-efficient than hand-labelling data?
Future Directions?