Probabilistic Modeling for Crowdsourcing Partially-Subjective Ratings

An T. Nguyen\textsuperscript{1}\textsuperscript{*}  Matthew Halpern\textsuperscript{1}  Byron C. Wallace\textsuperscript{2}
Matthew Lease\textsuperscript{1}

\textsuperscript{1}University of Texas at Austin
\textsuperscript{2} Northeastern University

HCOMP 2016

\textsuperscript{*}Presenter
Probabilistic Modeling

A popular approach to improve labels quality
Probabilistic Modeling

A popular approach to improve labels quality

Dawid & Skene (1979)

- Model true labels as hidden variables.
- Qualities of workers as parameters.
- Estimation: EM algorithm.

Extensions

- Bayesian (Kim & Ghahramani 2012)
- Communities (Venanzi et al. 2014)
- Instance features (Kamar et al. 2015)
Probabilistic Modeling

A popular approach to improve labels quality

Dawid & Skene (1979)
- Model true labels as hidden variables.
- Qualities of workers as parameters.
- Estimation: EM algorithm.

Extensions
- Bayesian (Kim & Ghahramani 2012)
- Communities (Venanzi et. al. 2014)
- Instance features (Kamar et. al. 2015)
Common assumption: Single *true label* for each instance. (i.e. objective task)
Probabilistic Modeling

Common assumption: Single true label for each instance. (i.e. objective task)

Subjective task?

- No single true labels
- Gold standard may not be appropriate (Sen et. al., CSCW 2015)
Video Rating task

Data:
- User interaction in smartphone.
- Varying hardware configurations (CPU freq. , cores, GPU)

Task
- Watch a short video
- Rate user satisfaction from 1 to 5
- 370 videos, \( \approx 50 \) AMT ratings each.
General Setting

For each instance:

- No single true label ...
  (i.e. no instance-level gold standard)
General Setting

For each instance:

- No single true label ...
  (i.e. no instance-level gold standard)
- ... but true distribution over true labels.
  (i.e. gold standard on instance label distribution)

Our data: Instances = Videos, Distribution of ratings.
General Setting

For each instance:

▶ No single true label ...
  (i.e. no instance-level gold standard)
▶ ... but true distribution over true labels.
  (i.e. gold standard on instance label distribution)

Our data: Instances = Videos, Distribution of ratings.

Two tasks:

▶ Predict that distribution.
▶ Detect unreliable workers.
Model

Intuition:

1. Unreliable workers tend to give unreliable ratings.
Model

Intuition:

1. Unreliable workers tend to give unreliable ratings.
2. Unreliable ratings are independent of instances.
   (e.g. rate videos without watching)
Model

Intuition:
1. Unreliable workers tend to give unreliable ratings.
2. Unreliable ratings are independent of instances. (e.g. rate videos without watching)

Assumptions:
1. Worker $j$ has param $\theta_j$: how often his labels unreliable.
Model

Intuition:
1. Unreliable workers tend to give unreliable ratings.
2. Unreliable ratings are independent of instances. (e.g. rate videos without watching)

Assumptions:
1. Worker $j$ has param $\theta_j$: how often his labels unreliable.
2. Rating labels are samples from $\text{Normal}(\mu, \sigma)$
Model

Intuition:

1. Unreliable workers tend to give unreliable ratings.
2. Unreliable ratings are independent of instances. (e.g. rate videos without watching)

Assumptions:

1. Worker $j$ has param $\theta_j$: how often his labels unreliable.
2. Rating labels are samples from $\text{Normal}(\mu, \sigma)$
   - Unreliable: $\mu, \sigma$ fixed.
   - Reliable: $\mu, \sigma$ vary with instances.
Model

(i indexes instances, j indexes workers)

Reliable indicator

\[ Z_{ij} \sim \text{Ber}(\theta_j) \]
Model
(i indexes instances, j indexes workers)

Reliable indicator

\[ Z_{ij} \sim \text{Ber}(\theta_j) \]

Labels

\[ L_{ij} | Z_{ij} = 0 \sim \mathcal{N}(3, s) \]
\[ L_{ij} | Z_{ij} = 1 \sim \mathcal{N}(\mu_i, \sigma_i^2) \]
Model
(i indexes instances, j indexes workers)

Reliable indicator

\[ Z_{ij} \sim \text{Ber}(\theta_j) \]

Labels

\[ L_{ij} | Z_{ij} = 0 \sim \mathcal{N}(3, s) \]
\[ L_{ij} | Z_{ij} = 1 \sim \mathcal{N}(\mu_i, \sigma_i^2) \]

Models: Features $\rightarrow \mu, \sigma$

\[ \mu_i = w^T x_i \]
\[ \sigma_i = \exp(v^T x_i) \]
Model

(i indexes instances, j indexes workers)

Reliable indicator

\[ Z_{ij} \sim \text{Ber}(\theta_j) \]

Labels

\[ L_{ij} | Z_{ij} = 0 \sim \mathcal{N}(3, s) \]

\[ L_{ij} | Z_{ij} = 1 \sim \mathcal{N}(\mu_i, \sigma_i^2) \]

Models: Features \( \rightarrow \mu, \sigma \)

\[ \mu_i = \mathbf{w}^T \mathbf{x}_i \]

\[ \sigma_i = \exp(\mathbf{v}^T \mathbf{x}_i) \]

Prior

\[ \theta_j \sim \text{Beta}(A, B) \]
Learning
(For model without prior on $\theta$)

**EM** algorithm, iterate
Learning
(For model without prior on $\theta$)

**EM** algorithm, iterate

**E-step:** Infer posterior over $Z_{ij}$
(analytic solution)

**M-step:** Optimize parameters $w, v$ and $\theta$
(BFGS)
Learning
(For the Bayesian model, with prior on $\theta$)

Closed-form EM not possible
Learning
(For the Bayesian model, with prior on $\theta$)

Closed-form EM not possible

Meanfield: approximate posterior $p(z, \theta)$ by

$$q(z, \theta) = \prod_{ij} q(Z_{ij}) \prod_{j} q(\theta_{j})$$
Learning
(For the Bayesian model, with prior on $\theta$)

Closed-form EM not possible

Meanfield: approximate posterior $p(z, \theta)$ by

$$ q(z, \theta) = \prod_{ij} q(Z_{ij}) \prod_{j} q(\theta_j) $$

Minimize $KL(q\|p)$ using co-ordinate descent.
(similar to LDA topic model, details on paper)
Evaluation

Difficulty: Subjective, don’t know who is reliable.
Evaluation

Difficulty: Subjective, don’t know who is reliable.

Solution:
- Assume all labels in data are reliable.
- Select $p\%$ workers at random.
- Change $q\%$ their labels to ‘unreliable labels’.
Difficulty: Subjective, don’t know who is reliable.

Solution:

- Assume all labels in data are reliable.
- Select $p\%$ workers at random.
- Change $q\%$ of their labels to ‘unreliable labels’.
- $p, q$ are evaluation parameters

$p \in \{0, 5, 10, 15, 20\}$, $q \in \{20, 40, 60, 80, 100\}$
Evaluation

Distribution of ‘unreliable labels’.
Evaluation

Distribution of ‘unreliable labels’.

AMT task
- Pretend to be spammer.
- Give ratings without watching video.

Recall our model:
- unreliable lab. \sim N(3, s)
- i.e. We don’t cheat.
Evaluation

Distribution of ‘unreliable labels’.

AMT task
- Pretend to be spammer.
- Give ratings without watching video.

Recall our model:
- unreliable lab. $\sim \mathcal{N}(3, s)$
- i.e. We don’t cheat.
Baselines

Predict ratings distribution (mean & var)
- Two Linear Regression models ...
- ... for mean and variance.
Baselines

Predict ratings distribution (mean & var)
- Two Linear Regression models ...
- ... for mean and variance.

Detect unreliable workers: Average Deviation
- Each instance: Deviation from the mean rating.
- Each worker: average the deviations.
- High AD → unreliable.
Results (varying unreliable workers)
(Baselines LR2: Linear Regression, AD: Average Deviation
NEW: Our Model, B-NEW: Our Bayesian Model)
Observations

- Bayesian model (B-NEW) better in prediction...
- ... but worse in detecting unreliable workers.
Observations

- Bayesian model (B-NEW) better in prediction...
- ... but worse in detecting unreliable workers.

Prior on worker parameter $\theta$

- Reduce overfitting of $w, v$.
- Create bias on workers.
Observations

- Bayesian model (B-NEW) better in prediction...
- ... but worse in detecting unreliable workers.

Prior on worker parameter $\theta$

- Reduce overfitting of $w, v$.
- Create bias on workers.

Other experiments

- Varying unreliable ratings, training data, number of workers
- Similar results (on paper).
Discussion

- Subjective task: common but little work.
- Our method improves prediction & detection.

Extensions:
- Improve recommendation systems.
- Other subjective tasks.
- More realistic evaluation.
- Better learning for Bayesian model.

Data + Code on GitHub

Acknowledgment: Reviewers, Workers, NSF (and Angry Birds).

Questions?
Discussion

- Subjective task: common but little work.
- Our method improves prediction & detection.

Extensions:
- Improve recommendation systems.
- Other subjective tasks.
- More realistic evaluation.
- Better learning for Bayesian model.
Discussion

- Subjective task: common but little work.
- Our method improves prediction & detection.

Extensions:
- Improve recommendation systems.
- Other subjective tasks.
- More realistic evaluation.
- Better learning for Bayesian model.

Data + Code on GitHub

Acknowledgment: Reviewers, Workers, NSF
Discussion

- Subjective task: common but little work.
- Our method improves prediction & detection.

Extensions:
- Improve recommendation systems.
- Other subjective tasks.
- More realistic evaluation.
- Better learning for Bayesian model.

Data + Code on GitHub

Acknowledgment: Reviewers, Workers, NSF (and Angry Birds).

Questions?