Crowdsourcing Practice for Efficient Data Labeling: Aggregation, Incremental Relabeling, and Pricing

Alexey Drutsa, Valentina Fedorova, Dmitry Ustalov, Olga Megorskaya, Evfrosiniya Zerminova, Daria Baidakova
Part V:

Theory on Aggregation

Dmitry Ustalov and Valentina Fedorova,
Data Analysis and Research Group, Yandex

Yandex.Toloka is a service of Swiss company Yandex Services AG
Tutorial schedule

Introduction: 15 min

Part I: 20 min
Main Components

Part II: 10 min
Introduction to Crowd Platform

Part III: 15 min
Brainstorming pipeline

Part IV: 60 min
Set & Run Projects

Break: 30 min

Part VI: 20 min
Set & Run Projects cont.

Part V: 25 min
Theory on Aggregation

Part VII: 10 min
Results & Conclusions
Labeling data with crowdsourcing

Classify images:

- Cat
- Dog
- Other

› How to choose a reliable label?
› How many labels per object?
› How much to pay for labels?
› …
Evaluation of labeling approaches

- Labels with a maximal level of accuracy for a given budget
- Labels of a chosen accuracy level for a minimal budget
Key components of labeling with crowds:

- Aggregation
- Incremental relabeling
- Performance-based pricing
Aggregation
Labeling data with crowds

Classify images:

Upload multiple copies of each object to label

Performers assign noisy labels to objects

Aggregate multiple labels for each object into a more reliable one
Process results

Yandex Toloka

Projects  Users  Skills  Profile  Messages

Projects  Does the image contains traffic lights?  pool

pool — closed

POOL TASKS (File example for task uploading (tsv.UTF-8))

Upload files  Edit  Preview

30 task suites  0 training task
90 tasks  10 control task

100 %
Done 30, accepted 30
View assignments
Multiclass labels
Project 1: Filter images

Are there shoes in the picture?

Yes

No
Notation

› Categories $k \in \{1, ..., K\}$. E.g.: ○ Cat △ Dog □ Other

› Objects $j \in \{1, ..., J\}$. E.g.:

› Performers: $w \in \{1, ..., W\}$. E.g.:
  • $W_j \subseteq \{1, ..., W\}$ - performers labeled object $j$
The simplest aggregation: Majority Vote (MV)

- The problem of aggregation:
  - Observe noisy labels
    \[ y = \{ y_j^w | j = 1, \ldots, J \text{ and } w = 1, \ldots, W \} \]
  - Recover true labels \( z = \{ z_j | j = 1, \ldots, J \} \)

- A straightforward solution:
  \[ \hat{z}_j^{MV} = \arg \max_{y=1,\ldots,k} \sum_{w \in W_j} \delta(y = y_j^w), \text{ where } \delta(A) = 1 \text{ if } A \text{ is true and } 0 \text{ otherwise} \]
Performance of MV vs other methods

Zhou D. et al. Regularized minimax conditional entropy for crowdsourcing. 2015
Properties of MV

› All performers are treated similarly  

› All objects are treated similarly
Advanced aggregation: performers and objects

› Parameterize expertise of performers by $e^w$

› Parameterize difficulty of objects by $d_j$
Advanced aggregation: latent label models
Latent label models: noisy label model

A noisy label model $M_j^w = M(e^w, d_j)$ is a matrix of size $K \times K$ with elements

$$M_j^w[c, k] = Pr(Y_j^w = k | Z_j = c)$$

$$q_{c1} + q_{c2} + q_{c3} = 1 \text{ for each } c$$
Latent label models: generative process

› Noisy labels generation:

1. Sample \( z_j \) from a distribution \( P_Z(p) \)

2. Sample \( y_j^w \) from a distribution \( P_Y(M_j^w[z_j,\cdot]) \)

In multiclassification, a standard choice for \( P_Z(\cdot) \) and \( P_Y(\cdot) \) is a Multinomial distribution \( \text{Mult}(\cdot) \)
Latent label models: parameters optimization

- Assumption: $y_j^w$ is cond. independent of everything else given $z_j$, $d_j$, $e^w$

- The likelihood of $y$ and $z$ under the latent label model:

$$L\left(\{z_j\}_{j=1}^J, p, \{d_j\}_{j=1}^J, \{e^w\}_{w=1}^W\right) = \prod_{j \in J} \sum_{z_j \in \{1, \ldots, K\}} \Pr(z_j | p) \prod_{w \in W_j} \Pr(y_j^w | z_j, d_j, e^w)$$

- Estimate parameters and true labels by maximizing $L(...)$
Latent label models: EM algorithm

Maximization of the expectation of log-likelihood (LL), a lower bound on LL of \( y \) and \( z \)

\[
\mathbb{E}_z \log \Pr(y, z) = \sum_{j \in I} \sum_{z_j \in \{1, ..., K\}} \Pr(z_j | p) \log \prod_{w \in W_j} \Pr(z_j | p) \Pr(y_j^w | z_j, d_j, e^w)
\]

**E-step**: Use Bayes’ theorem for posterior distribution of \( \hat{z} \) given \( p, d, e \):

\[
\hat{z}_j[c] = \Pr(Z_j = c | y, p, d, e) \propto \Pr(Z_j = c | p) \prod_{w \in W_j} \Pr(y_j^w | z_j = c, d_j, e^w)
\]

**M-step**: Maximize the expectation of LL with respect to the posterior distribution of \( \hat{z} \):

\[
(p, d, e) = \arg \max \mathbb{E}_z \log \Pr(z_j | p) \prod_{w \in W_j} \Pr(y_j^w | z_j, d_j, e^w)
\]

- Analytical solutions
- Gradient descent
Latent label model (LLM): special cases

Dawid and Skene model (DS):
› categories are different
› objects are similar
› performers are different

Generative model of labels, abilities, and difficulties (GLAD):
› categories are similar
› objects are different
› performers are different

Minimax conditional entropy model (MMCE):
› categories are different
› objects are different
› performers are different
Dawid and Skene model (DS)

LLM with parameters:

\( p - \) vector of length \( K \): \( p[i] = \Pr(Z = c) \)

\( e^w - \) matrix of size \( K \times K \):
\[
e^w[c, k] = \Pr(Y^w = k | Z = c)
\]
DS: parameters optimization

\( \hat{z}_j[c] = \frac{p[c] \prod_{w \in W_j} e^w[c, y_j^w]}{\sum_k p[k] \prod_{w \in W_j} e^w[k, y_j^w]}, \quad c = 1, \ldots, K \)

\( e^w[c, k] = \frac{\sum_{j \in J} \hat{z}_j[c] \delta(y_j^w = k)}{\sum_{q=1}^K \sum_{j \in J} \hat{z}_j[c] \delta(y_j^w = q)}, \quad k, c = 1, \ldots, K \)

\( p[c] = \frac{\sum_{j \in J} \hat{z}_j[c]}{J}, \quad c = 1, \ldots, K \)
Generative model of Labels, Abilities, and Difficulties (GLAD)

LLM with parameters:

- scalar \( d_j \in (0, \infty) \)
- scalar \( e^w \in (-\infty, \infty) \)
- Model:

\[
\Pr(Y_j^w = k|Z_j = c) = \begin{cases} 
  a(w, j), & c = k \\
  1 - a(w, j), & c \neq k \\
  \frac{K - 1}{1 + \exp(-e^w d_j)}, & c \neq k
\end{cases}
\]

where \( a(w, j) = \frac{1}{1 + \exp(-e^w d_j)} \)

Whitehill et al., Whose vote should count more: Optimal integration of labels from labelers of unknown expertise, 2009
GLAD: parameters optimization

Let $a(w, j) = \frac{1}{1 + \exp(-e^w d_j)}$ and $P(z_j)$ be a predefined prior (e.g., $P(z_j) = \frac{1}{K}$)

E-step:

$$\hat{z}_j \sim P(Z_j = c) \prod_{w \in W_j} a(w, j)^{\delta(y_j^w = c)} \left(1 - a(w, j)\right)^{\delta(y_j^w \neq c)}$$

M-step: estimate $(d^t, e^t)$ for given $\hat{z}$ using gradient descent

$$(d^t, e^t) = \arg\max \sum_{j \in J} \left[ \mathbb{E}_{\hat{z}_j} \log P(z_j) + \sum_{w \in W_j} \mathbb{E}_{\hat{z}_j} \log \Pr(y_j^w | z_j) \right]$$
MiniMax Conditional Entropy model (MMCE)

› Find parameters that minimize the maximum conditional entropy of observed labels:

\[
\min_Q \max_P - \sum_{j \in j} Q(Z_j = c) \sum_{w \in W} P(Y_j^w = k|Z_j = c) \log P(Y_j^w = k|Z_j = c)
\]

LLM with parameters:

› \(d_j\) – matrix of size \(K \times K\)

› \(e^w\) – matrix of size \(K \times K\)

› Noisy label model:

\[
\Pr(Y_j^w = k|Z_j = c) = \exp(d_j[c, k] + e^w[c, k])
\]

Zhou et al., Learning from the Wisdom of Crowds by Minimax Entropy, 2012
### Summary of aggregation methods

<table>
<thead>
<tr>
<th>Categories (K)</th>
<th>MV</th>
<th>DS</th>
<th>GLAD</th>
<th>MMCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>○</td>
<td>△</td>
<td>○</td>
<td>△</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Objects (J)</th>
<th>MV</th>
<th>DS</th>
<th>GLAD</th>
<th>MMCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image" alt="Objects" /></td>
<td><img src="image" alt="Objects" /></td>
<td><img src="image" alt="Objects" /></td>
<td><img src="image" alt="Objects" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Performers (W)</th>
<th>MV</th>
<th>DS</th>
<th>GLAD</th>
<th>MMCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image" alt="Performers" /></td>
<td><img src="image" alt="Performers" /></td>
<td><img src="image" alt="Performers" /></td>
<td><img src="image" alt="Performers" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of parameters</th>
<th>MV</th>
<th>DS</th>
<th>GLAD</th>
<th>MMCE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>(WK^2 + K)</td>
<td>(W + J)</td>
<td>((W + J)K^2)</td>
</tr>
</tbody>
</table>
Key components of labeling with crowds

Aggregation → Incremental relabeling → Performance-based pricing
Incremental relabeling

aka dynamic overlap
Pool settings: dynamic overlap

**Quality control**
Add rules to get more accurate responses.
All rules work independently.

- **Non-automatic acceptance:** No
- **Captcha frequency:** None

**Overlap**
Specify how many performers you want to complete each task in the pool.

- **Overlap:**
- **Dynamic overlap:** Off

**Speed/quality ratio**
Specify additional conditions for selecting performers by their rating in Toloka. This will improve quality, but may reduce the speed of task completion because there will be fewer performers available for completing tasks. Learn more

- **Top % on line time:**
- Specify the percentage of top-rated active users who can access tasks in the pool.
Incremental relabeling problem

Obtain aggregated labels of a desired level of quality using a fewer number of noisy labels
Incremental relabeling scheme (IRL)

Request a label for each object

In real time IRL algorithm receives:
(1) previously accumulated labels
(2) new labels

Decides:
(1) which objects are labeled
(2) which objects to relabel

Repeat until all tasks are labeled
Notations

› Consider one object $z \in \{1, ..., K\}$ - latent true label

› $y_w \in \{1, ..., K\}$ - observed noisy label from performer $w$:  

Classify images:
- Cat
- Dog
- Other
Notations

- Noisy label model for performer $w$:
  \[ M_w \in [0,1]^{K \times K}: \Pr(Y_w = k | Z = c) = M_w[c, k] \]

- Prior distribution: \( \Pr(Z = k) = p_k \)
Posterior distribution

\[ \{y_{w_1}, \ldots, y_{w_n}\} - \text{accumulated noisy labels for the object} \]

Using Bayes rule:

\[
\Pr(Z = k | \{y_{w_1}, \ldots, y_{w_n}\}) = \frac{\Pr(Z = k) \Pr(\{y_{w_1}, \ldots, y_{w_n}\} | Z = k)}{\Pr(\{y_{w_1}, \ldots, y_{w_n}\})} = \frac{p_k \prod_{i=1}^{n} M_{W_i}[k, y_{w_i}]}{\sum_{t=1}^{K} p_t \prod_{i=1}^{n} M_{W_i}[t, y_{w_i}]}.
\]
Expected accuracy of aggregated labels

Let $A$ be an aggregation model, e.g. MV, DS, GLAD,…

Denote aggregated label $z^A = A(\{y_{w_1}, ..., y_{w_n}\})$

Expected accuracy of aggregated labels given noisy labels is

$$E(\delta(z = z^A) | \{y_{w_1}, ..., y_{w_n}\}) = Pr(z = z^A | \{y_{w_1}, ..., y_{w_n}\})$$

Stop labeling if $E(\delta(z = z^A) | \{y_{w_1}, ..., y_{w_n}\}) \geq C$

Threshold in IRL: cost – accuracy trade-off

Optimal threshold $c = 0.95$

A higher $c$ does not increase accuracy

Saving $\approx 35\%$ of noisy labels
Key components of labeling with crowds

- Aggregation
- Incremental relabeling
- Performance-based pricing
Performance-based pricing
aka dynamic pricing
Pool settings: dynamic pricing

Price per task suite
You can add one or more tasks to the page. Enter the total price for all tasks on the page.

Price in US dollars: 0.07

Performers
Filter performers who can access the task. Toloka has users from different countries, so don't forget to filter by language and region. Learn more.

Adult content: Yes
Labeling as a game: notation

performer $w$

Task

Effort
$h \in [0,1]$

Accuracy
$a \in [0,1]$

Value
$v = v(a)$

Payment
$p = p(a)$

requester

Classify images:

- Cat
- Dog
- Other
Labeling as a game: formalization

- Each performer $\omega$ chooses a level of effort $h$ for labeling object to maximize earnings per unit of spent effort:

$$\max_{h \geq 0} \frac{p(a_\omega(h))}{h}$$

- The requester chooses a pricing $p(a)$ to minimize payments per unit of obtained value

$$\max_{a \in [0,1]} \frac{v(a)}{p(a)}$$
Labeling as a game: incentive compatible pricing

Assume \( a_w(h) \) is a linear function of \( h \):

\[
a_w(h) = c_1 h + c_0
\]

The requester and performers maximize their utility simultaneously if the pricing \( p(a) \) for each label is proportional to its accuracy \( a \).
Performance-based pricing in practice: settings

- Price $p$ for the level of accuracy $a_0$: $\Pr(\hat{z} = z) \geq a_0$  
  E.g.: $p = 0.3\$$  
  $a_0 = 0.99$

- $\hat{q}_w = \Pr(y^w = z)$ - estimated quality level of performer $w$,  
  e.g. the fraction of correct labels for golden set (GS):

  - 5 correct GS among 10  
    $\hat{q}_w = 0.5$
  - 16 correct GS among 20  
    $\hat{q}_w = 0.8$
  - 100 correct GS among 100  
    $\hat{q}_w = 1$

Wang, Ipeirotis, and Provost, Quality-Based Pricing for Crowdsourced Workers, 2013
Performance-based pricing in practice: settings

\[ \hat{z}_j^{w_{MV}} = \arg \max_{y=1,\ldots,K} \sum_{w \in W_j} \hat{q}_w \delta(y = y_j^w) \]

IRL algorithm is based on the expected accuracy of \( \hat{z}_j^{w_{MV}} \).
Performance-based pricing in practice

**Pricing rules**

1. If \( \hat{q}_w \geq a_0 \), then the price is \( p \)

2. Else find \( n \):

\[
\sum_{k=0}^{n/2} \binom{n}{k} \hat{q}_w^{n-k} (1 - \hat{q}_w)^k \geq a_0
\]

The price is \( \frac{p}{n} \)

- \( a_0 = 0.99 \)
- \( \hat{q}_w = 1 \)
  - \( \Rightarrow n = \infty \)
  - Expected accuracy for MV
  - \( 0.3\$ \)

- \( \hat{q}_w = 0.8 \)
  - \( \Rightarrow n = 15 \)
  - \( 0.02\$ \)

- \( \hat{q}_w = 0.5 \)
  - \( \Rightarrow n = \infty \)
  - \( 0\$ \)
Key components of labeling with crowds

Aggregation → Incremental relabeling → Performance-based pricing
Thank you!
Questions?

Dmitry Ustalov
Analyst at Data Analysis and Research Group

dustalov@yandex-team.ru
https://research.yandex.com/tutorials/crowd/sigmod-2020